Advanced Nonlinear Dimensionality Reduction Methods for Multidimensional Time Series: Application to Human Motion Analysis

by

Michał Lewandowski

Submitted in partial fulfilment of the requirements of
Kingston University for the degree of

Doctor of Philosophy

June, 2011

Kingston University London

Supervision:

Dr Jean-Christophe Nebel (Director of Studies)

Dr Dimitrios Makris

Dr Jarosław Francik

Dr James Orwell

Digital Imaging Research Centre
Faculty of Computing, Information Systems and Mathematics
Kingston University
Penrhyn Road
Kingston upon Thames
Greater London
KT1 2EE
United Kingdom

Abstract

This dissertation contributes to the state of the art in the field of pattern recognition and machine learning by advancing a family of nonlinear dimensionality reduction methods. We start with the automatisation of spectral dimensionality reduction approaches in order to facilitate the usage of these techniques by scientists in various domains wherever there is a need to explore large volumes of multivariate data. Then, we focus on the crucial and open problem of modelling the intrinsic structure of multidimensional time series. Solutions to this outstanding scientific challenge would advance various branches of science from meteorology, biology, engineering to computer vision, wherever time is a key asset of high dimensional data. We introduce two different approaches to this complex problem, which are both derived from the proposed concept of introducing spatio-temporal constraints between time series. The first algorithm allows for an efficient deterministic parameterisation of multidimensional time series spaces, even in the presence of data variations, whereas the second one approximates an underlying distribution of such spaces in a generative manner. We evaluate our original contributions in the area of visual human motion analysis, especially in two major computer vision tasks, i.e. human body pose estimation and human action recognition from video. In particular, we propose two variants of temporally constrained human motion descriptors, which become a foundation of view independent action recognition frameworks, and demonstrate excellent robustness against style, view and speed variability in recognition of different kinds of motions. Performance analysis confirms the strength and potential of our contributions, which may benefit many domains beyond computer vision.

To my amazing wife
Agnieszce.

Acknowledgments

My time at Kingston University London has been an exciting and invaluable experience. It was very satisfying to conduct this research and propose novel solutions to scientific challenges of machine learning and computer vision fields.

First and foremost, I would like to thank my excellent supervisors, for their continuous guidance, wisdom and patience during my PhD. Especially, I would like to express my deepest gratitude to my primary supervisor Dr Jean-Christophe Nebel for conceiving this project and his enthusiastic support throughout it. His desire to maintain the highest standards and professional attitude were key to the development of this research and to my growth as a researcher. I am grateful for his advice and assistance to keep me along the right track, while letting me to implement my ideas and take the research my own direction. What's more, I truly appreciated for his numerous hours spent on reading my papers and this thesis, his insight comments and suggestions contributed greatly to them. Also, I would like to thank my co-supervisor Dr Dimitrios Makris for his expertise, useful discussions and priceless feedback during the preparation of papers and this dissertation. Overall, working with my all supervisors has been a pleasure. They were an exceptional source of knowledge and motivation for me, and definitely I have learnt a lot from them, which without doubt put me in good shape for the future.

Moreover, I would like to thank all my colleagues at the Digital Imaging Research Centre, for their support, valuable hints and discussions. Among all, I want to sincerely acknowledge in name Jesús Martínez del Rincón, Paul Kuo, Alexandros Moutzouris and Norbert Buch for inspiring research atmosphere and fruitful collaborations.

Furthermore, I would like to give my heartfelt thanks to my parents and family for their endless support in all my endeavours, always helpful advice and unconditioned faith in me all the way through. Also a special thanks to all my friends who made my life more enjoyable over this period and helped a lot in clearing my head in times of pressure and intensive work.

Finally, above all, I dedicate this dissertation to my beautiful wife Agnieszce, I cannot find appropriate words to express my genuine gratitude. Without her love, unfailing support and constant encouragement I would have given up long time ago. Thank you for your amazing generosity and understanding which has enabled me to fulfil this dream.

Declarations

I hereby declare that this dissertation describes my solely own research, which was carried out at Kingston University, except where otherwise indicated. Other sources are acknowledged by explicit references. Some of the research presented in this thesis has already been published or is under review for publication. For a complete list of publications, please refer to the next page.

This thesis has not been previously accepted in substance for any degree and is not being concurrently submitted to any other University for examination either in the United Kingdom or overseas.

Michał Lewandowski

.

List of Publications

- Kuo, P., Ammar, T., **Lewandowski, M.**, Makris, D., and Nebel, J.-C. (2009). Exploiting human bipedal motion constraints for 3d pose recovery from a single uncalibrated camera. *Proceedings of the 4th International Conference on Computer Vision Theory and Applications*, 1:557–564. [Kuo et al., 2009].
- **Lewandowski, M.**, Makris, D., and Nebel, J. (2009). Automatic configuration of spectral dimensionality reduction methods for 3d human pose estimation. *Workshop on Visual Surveillance at International Conference on Computer Vision*. [Lewandowski et al., 2009].
- **Lewandowski, M.**, Makris, D., and Nebel, J.-C. (2010). Automatic configuration of spectral dimensionality reduction methods. *Pattern Recognition Letters*, 31. [Lewandowski et al., 2010a].
- **Lewandowski, M.**, Martinez-del Rincon, J., Makris, D., and Nebel, J.-C. (2010). Temporal extension of laplacian eigenmaps for unsupervised dimensionality reduction of time series. *Proceedings of the 20th International Conference on Pattern Recognition* (oral presentation). [Lewandowski et al., 2010c].
- **Lewandowski, M.**, Makris, D., and Nebel, J.-C. (2010). View and style-independent action manifolds for human activity recognition. *Proceedings of the 11th European Conference on Computer Vision*, 6316. [Lewandowski et al., 2010b].
- **Lewandowski, M.**, Makris, D., and Nebel, J.-C. (2011). Probabilistic feature extraction from multivariate time series using spatio-temporal constraints. *Proceedings of the 15th Pacific-Asia Conference on Knowledge Discovery and Data Mining* (oral presentation). [Lewandowski et al., 2011].

- Moutzouris, A., Martinez-del Rincon, J., **Lewandowski, M.**, Nebel, J.-C., and Makris, D. (2011). Human pose tracking in low dimensional spaces enhanced by limb correction. *Proceedings of the 18th International Conference on Image Processing*. [Moutzouris et al., 2011].

Glossary of Terms

PCA	Principle Component Analysis
PPCA	Probabilistic Principle Component Analysis
MDS	Multidimensional Scalling
Isomap	Isometric Feature Mapping
ST-Isomap	Spatio-Temporal Isometric Feature Mapping
LLE	Locally Linear Embedding
LE	Laplacian Eigenmaps
TLE	Temporal Laplacian Eigenmaps
GPLVM	Gaussian Process Latent Variable Model
ST-GPLVM	Spatio-Temporal Gaussian Process Latent Variable Model
BC-GPLVM	Back-Constrained Gaussian Process Latent Variable Model
GPDM	Gaussian Process Dynamical Model
LVM	Latent Variable Model
GTM	Generative Topographic Mapping
MLLM	Mixture of Local Linear Models
RBF	Radial Basis Function
RBFN	Radial Basis Function Network
G-RBFN	Graph-based Radial Basis Function Network
MLP	Multilayer Perceptron
ID	Intrinsic Dimensionality
MoCap	Motion Capture
DTW	Dynamic Time Warping
MTS	Multidimensional/Multivariate Time Series
MVS	Multidimensional/Multivariate View Series
MAE	Mean Angle Error
RMS	Root Mean Square Error

MI	Mutual Information
SR	Spearman Rho
RV	Residual Variance
PA	Procrustes Analysis
KMC	K-mean clustering
RPCL	Rival Penalized Competive Learning
MCL	Markov Cluster Algorithm
EE	Eigenvalue based estimator

Glossary of Notations

Generally, we denote scalars in bold lowercase or uppercase ($\mathbf{d}, \mathbf{N}$), vectors in italics lowercase (x, y), whereas matrices in italics uppercase (X, Y).

d	the dimension of reduced/latent space
D	the dimension of data space
x	the vector in a low **d**-dimensional space
y	the vector in a high **D**-dimensional data space
X	the matrix of low **d**-dimensional vectors
Y	the matrix of high **D**-dimensional vectors
N	the number of vectors in X, Y , i.e. number of data points
i, j	indices of matrices, usually in range $i, j = 1..\mathbf{N}$, if not overridden otherwise
c	the centre of cluster
C	the matrix of centres
Z	the number of clusters in the matrix C
L	the Laplacian matrix of a graph
W	weights of a graph
x_i, y_i, c_i	the (i)*th* vector of a corresponding matrix X, Y, C
$x_{i,j}, y_{i,j}, c_{i,j}, w_{i,j}$	the (i,j)*th* entry of a corresponding matrix X, Y, C, W
λ	the vector of eigenvalues
v	the vector of eigenvectors or the vector of view parameters
s	the vector of style parameters
$\bullet$	the dot product
$\lVert \cdot \rVert$	the Euclidean norm

$\mathbf{A}\times\mathbf{B}$	the size of matrix with **A** rows and **B** columns
$\mathbf{AB}$	a product of matrices **A** and **B**
$\mathbf{A}^T$	a transpose of matrix **A**
F	the dimensionality reduction mapping from data space to to latent or reduced-dimension representation space
f	the reconstruction mapping from latent or reduced-dimension representation space to data space
G	the forward mapping function from a high to low dimensional space
g	the inverse mapping function from a low to high dimensional space
I	the identity matrix
$tr(A)$	the trace of the matrix A
$p(x)$	the probability of x
$p(y \mid x)$	the probability of y given x
μ	the mean
Σ	the covariance matrix
$\mathbb{N}(\mu,\Sigma)$	the Gaussian distribution with mean μ and covariance Σ
$\mathbb{N}(X \mid \mu,\Sigma)$	the Gaussian distribution over X with mean μ and covariance Σ
Φ	hyperparameters
ψ	the empirical kernel map/interpolation matrix
φ	the Gaussian basis function
K	a positive semi definite Mercel kernel
$k_{i,j}$	the element of matrix K
$\kappa(x_i, x_j)$	the kernel function evaluated on data points x_i and x_j

K	the neighbourhood size
$\mathbb{R}$	real numbers
ρ	the value of quantitave measure
$H(X)$	the marginal entropy of X
$H(X,Y)$	the marginal entropy of X and Y

List of Figures

List of Tables

Content

1. Introduction

"How complex or simple a structure is depends critically upon the way we describe it. Most of the complex structures found in the world are enormously redundant, and we can use this redundancy to simplify their description. But to use it, to achieve the simplication, we must find the right representation" [Simon, 1996]

Professor H.A. Simon

Nobel Prize Winner 1978

Understanding and exploration of the intrinsic structure of multidimensional phenomena are of fundamental importance in data mining, pattern recognition, and machine learning. The past decade has witnessed a remarkable explosion of a high dimensional digital content in most disciplines of science due to rapid improvements in data acquisition and storage capabilities as well as falling costs of data warehousing technology. As a consequence, in many areas where observations used to be scarce, we have now access to sufficient amounts of information to explain a phenomenon with a data-driven paradigm, i.e. to induce a model for an event of interest given acquired observations.

Providing a machine which has the ability to learn and study such models has been fascinating scientists for a long time in various branches of science from linguistics, biology, engineering, artificial intelligence to computer vision. However, in order to represent the natural complexity and all inherent aspects of a phenomenon, a tremendous amount of parameters has to be measured. This high dimensionality introduces outstanding challenges in the creation of generalised and meaningful models by a machine, since the number of available training samples is usually not sufficient to cover appropriately all dimensions. In addition, many parameters are redundant or irrelevant in describing a given event of interest, thus making the process of learning extremely difficult. These problems have led to the formation of the machine learning/pattern recognition field referred to as *dimensionality reduction*. Dimensionality reduction is a transformation of high dimensional observations into a faithful low dimensional representation in order to simplify data representation and extract true intrinsic parameterisation of a phenomenon. This is achieved by removing redundant information, while maintaining important relationships between parameters. As a consequence, the number of required parameters is significantly reduced to essential ones, thus facilitating the process of model learning by a machine.

Let's consider the scientific discipline of computer vision, which aims at enabling a machine to interpret the world, which is presented to it by one or more cameras, in a similar way to humans. Recorded human motion is a classic example of a high dimensional and complex phenomenon, which is extremely difficult to model by a machine due to large variations in motion style and dynamics, human body shape and appearance, camera viewpoint and environment settings. However, automatic analysis of human motion is now of fundamental importance in many areas and desired by many potential applications. They include content-based video analysis, security and surveillance systems, human-computer interactions, animation and synthesis in the entertainment industry (e.g. games and movies). Therefore, an appealing solution to tackle this problem is to reduce the dimensionality of human motion in order to assist the generation of robust human motion models.

In this thesis, we explore the realm of dimensionality reduction with a special focus on its application to human motion analysis. We propose several novel approaches which allow for the effective modelling of high dimensional data and prove to be superior to the current state of the art in a range of computer vision tasks such as pose recovery and action recognition.

In this introductory chapter, first, we present the context of our research in sections 1.1 and 1.2. Following this, the principal contributions of this work are summarised in section 1.3, whereas the structure of this dissertation is outlined in section 1.5.

1.1. Dimensionality Reduction

A phenomenon is usually represented by a set of observations, which are measurements of a set of D quantitative values, i.e. features or attributes that are collected by data capture devices. These values can be arranged in the form of a D-dimensional vector, which reflects distinctive aspects and characteristics of the considered observation. Since features can vary independently from each other; they are often referred to as the degrees of freedom of a model [Good, 1973]. However, due to the natural complexity of the modelled phenomena and imperfect capturing devices, a very large number of features is collected with the aim to capture adequately all inherent aspects of observed events. This leads to information overload in most sciences and the crucial paradox: the more features (dimensions) are available, the more challenging the process of model learning and information extraction is. For instance, high dimensional data may contain several features that are measurements of the same underlying cause, thus they are redundant. Moreover, some features may be irrelevant and not very informative in characterising the nature of the phenomenon. Finally, a closely related fundamental challenge in the high-dimensional data analysis is the so-called dimensionality curse (see section 2.2), i.e. observations in a high dimensional space are far less representative than those in a low dimensional space because of an inherent sparsity of the high dimensional space. As a result, the number of observations required to cover ‘satisfactory’ the entire high dimensional space increases exponentially with the number of measured features. This implies that very often the collected data represent the degrees of freedom of capturing devices instead of those of the actual underlying phenomenon.

Dimensionality reduction overcomes these fundamental problems associated with the exploration of large volumes of multidimensional data. This is achieved by discovering a compact, meaningful and intrinsic parameterisation of the phenomenon that governs the observed data. Therefore, dimensionality reduction can be seen as the process which transforms a capturing device representation with many degrees of freedom in a smaller number of relevant degrees of freedom which characterise accurately the event of interest. A schematic representation of this process is shown in Figure 1.1. In addition to computational costs decrease, the key advantage of dimensionality reduction is better data representation and understanding while preserving as much of the original information as possible. Moreover, since the world is essentially three dimensional, 1, 2 and even 3-dimensional data are very intuitive and assimilable representations for human perception. As we will show in this thesis, many complex phenomena, such as human motion, are intrinsically of very few dimensions, therefore dimensionality reduction can be employed to visualise such data and facilitate its analysis and interpretation.

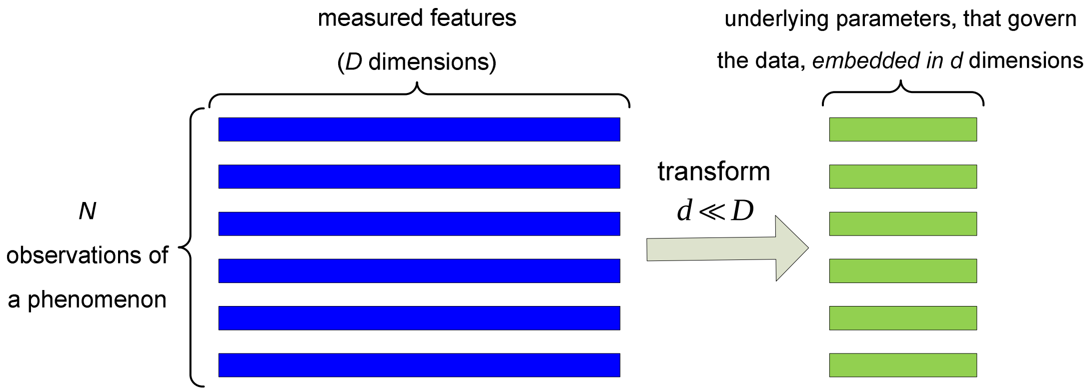


Figure 1.1. The concept of dimensionality reduction.

To illustrate the concept of dimensionality reduction, let's consider an example from visual perception, where a dataset consists of images of an object taken from multiple orientations simultaneously. Images can be thought of as points in some high-dimensional image space where each coordinate represents the intensity value of a single pixel. In this example, images have a size of 76×101 pixels, and thus form points in a 7676-dimensional observation space. However, despite of appearance differences, the perceptually meaningful structure of these images has only one intrinsic degree of freedom (dimension), i.e. the orientation of the depicted object. Therefore, these images are expected to lie on or near a 1-dimensional curve which is embedded in a two dimensional space to model the cyclic nature of the view change (Figure 1.2). This 1-dimensional curve is parameterised only by the viewing angle. The objective of dimensionality reduction techniques is to identify this embedded representation by removing irrelevant and overlapping information from data in order to extract the intrinsic parameterisation that truly governs them.

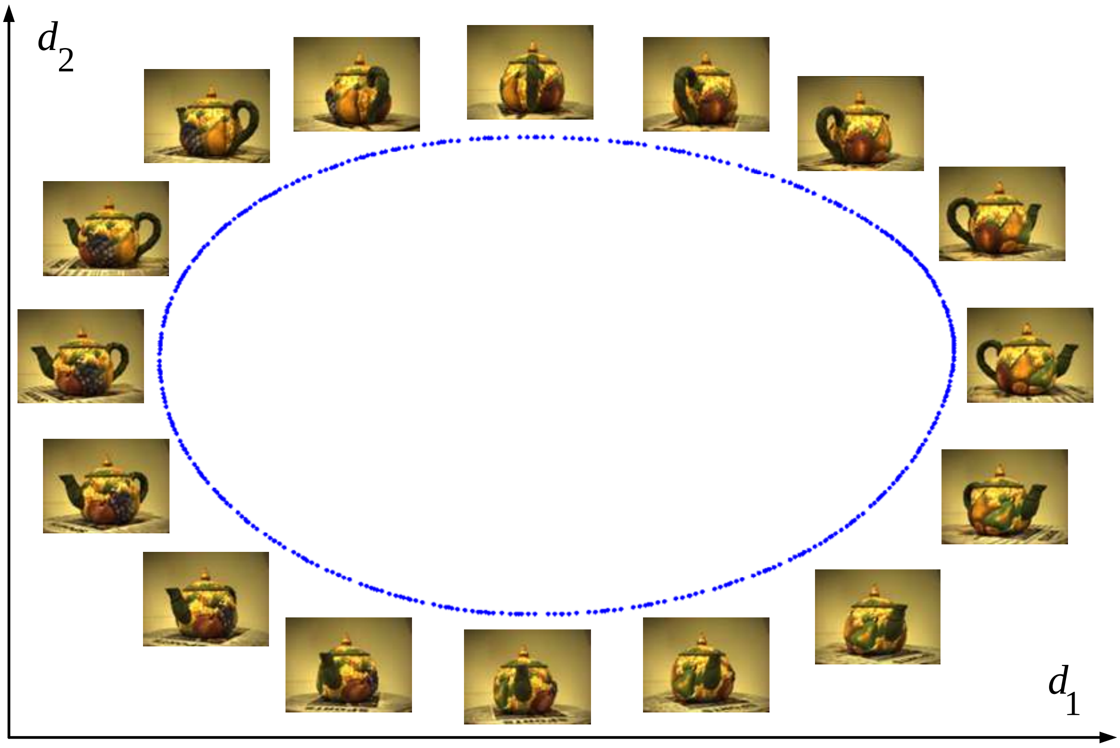


Figure 1.2. The 1-dimensional parameterisation of a highly dimensional image dataset embedded in a 2-dimensional space. The intrinsic dimension corresponds to the viewing angle of the depicted object.

Nowadays, many multivariate statistical methods often rely on a pre-processing step involving some form of dimension reduction to eliminate undesired properties of high dimensional data and consequently improve overall performance. Figure 1.3 illustrates this concept, showing the dimensionality reduction as a pre-processing stage in the whole system. As a result, dimensionality reduction has become an essential process across a wide variety of fields wherever there is a need to explore large volumes of multivariate data. In particular, scientists in the following domains have to deal with this problem:

- computer vision,
- image processing,
- artificial intelligence,
- medicine,

- linguistics,
- signal processing,
- meteorology,
- engineering,
- bioinformatics.

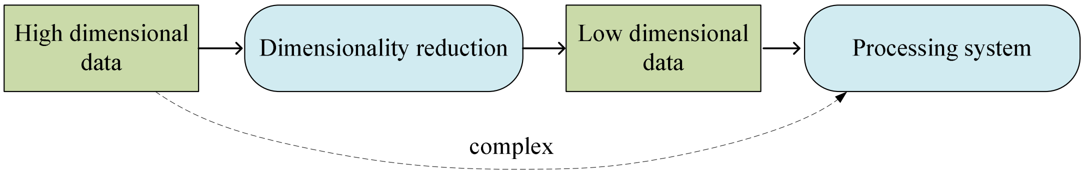


Figure 1.3. Performance of many processing systems can be improved in terms of accuracy and efficiency by reducing dimensionality of the data in a pre-processing step.

1.2. Multidimensional Time Series

In many real world applications, the analysis of behavioural and dynamic characteristics of phenomena is much more informative than the description of their states at a certain point in time. Therefore, another crucial challenge in modelling high dimensional data is the time aspect, which is the intrinsic property of many natural as well as man-made phenomena. As a consequence, the adaptation of time in the dimensionality reduction process seems to be an intuitive and really relevant objective to study, which has only recently been investigated by the research community.

Time series is the standard digital representation of phenomena with the temporal correlation among observations [Hannan, 1970, Chatfield, 1996]. Observations are collected at regular intervals over a period of time and, as a result, successive observations exhibit a certain level of dependency. Note that in principle the two main objectives of time series analysis are to characterise and represent time series and/or to forecast future behaviour. In this thesis, we are only interested in the modelling of time series representation, and we do not address the problem of time series extrapolation.

1.2.1. Human Motion Analysis

A typical example of multidimensional time series data is human motion data. Human motion can be seen abstractly as a continuous state machine, where the body is considered to be in a single high dimensional state at a given instant. The space of human motion is highly dimensional since the human body is a deformable object with no less than 244 degrees of freedom [Zatsiorsky, 2002], anthropometric variability between people [Easterby et al., 1982] and dynamics [Farnell, 1999]. Since, subsequent states of real human motion are temporally correlated and 'short' motion patterns tend to be repeatable over time, a natural digital representation of motion is a time series sequence of high dimensional feature vectors, which correspond to successive states of the motion.

One of the pioneering and systematic investigations into the nature of human motion was carried out by the photographer Eadweard Muybridge in the late 19th century [Muybridge, 1901]. He built a complex system of multiple cameras to capture motion, which was composed of a fixed battery from 12 to 24 cameras along an open shed and an invented shutter with a short exposure time. The cameras were triggered sequentially over time at sufficient speed to generate the earliest 'digital' dataset of human motion and thus allowing the first manual vision-based motion analysis (Figure 1.4).

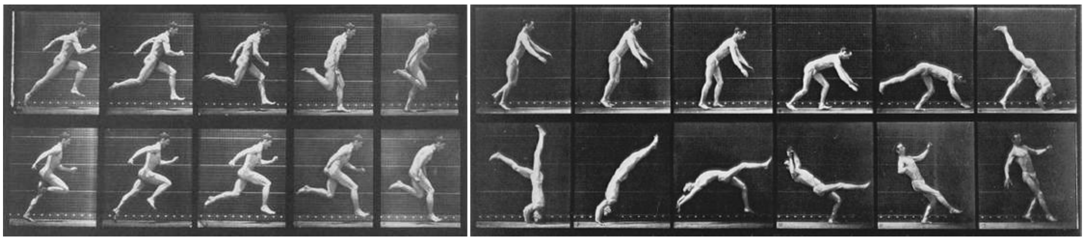


Figure 1.4. The series of photos of the human figure in motion by Eadweard Muybridge taken in the late 19th century [Muybridge, 1901].

In the next century, the classic moving light display experiment of Johansson [Johansson, 1973] has paved the way to the automatic human motion analysis and mathematical modelling of human motion. Johansson demonstrated that a sequence of only a few reflective markers attached to major joints of human body is sufficient to understand and recover motion by a human subject (Figure 1.5).

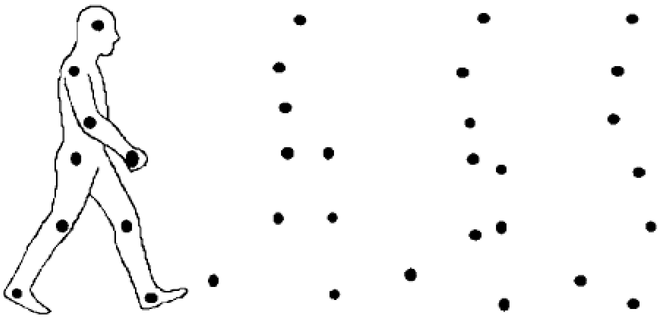


Figure 1.5. Illustration of moving light displays, taken from [Thornton et al., 1998]. When static images are presented in a sequence, an observer can easily organise the complex patterns of lights movement into a coherent perception of human motion.

Over the last few decades this experiment inspired many researchers in human motion analysis and directly led to the invention of marker-based motion capture systems. Modelling human motion by these systems involves strapping sensors (e.g. electromagnetic markers) to the body and then recording transmitted signals in three dimensions at very high frequencies as an individual performs various movements [Menache, 1999]. However, these systems are not only expensive but also very invasive, typically requiring special clothing and a controlled studio-like environment. Moreover, they are not very practical in applications where observed humans are not cooperative. Therefore, in practice, they are primarily used for the training of machine learning algorithms.

In contrast, cameras are low-cost, flexible and non-obtrusive devices which are able to record a massive amount of information about an observed scene. However, in order to perform any human motion analysis, first the individual has to be localised and motion has to be extracted from videos. This is an extremely difficult problem due to image variability which originates from cluttered and dynamic environments, depth ambiguity, occlusions, lighting conditions, camera viewpoint as well as people variability in terms of physical appearance and motion style. Assuming that these issues can be solved satisfactorily, the challenging machine learning problem of inducing human motion models from the extracted information has to be addressed. These models should constrain the space of plausible solutions while maintaining appropriate adaptability to all forms of human movement variations. Despite all these difficulties, markerless and vision-based analysis of human motion is currently one of the most active research domains.

Video based analysis of human motion comprises many aspects. In this thesis, we limit our scope of interest to human pose recovery and human action recognition. The former aims at the determination of locations or angles of key body joints given an image or a video capture of human figure. The latter is a high level description of an image sequence by assigning a meaningful annotation that best describes the observed motion.

1.3. Aim and Objectives

The overall aim of this research is to advance the field of dimensionality reduction with a special attention to human motion analysis.

First, although dimensionality reduction transformation may allow improving overall performance in many processing systems, difficulties in practical usage of these algorithms limit their applicability in various domains. Most powerful dimensionality reduction approaches rely on a set of parameters and extensions in order to be applied effectively. However, manual user input into the process requires specialised knowledge, whereas many scientists would prefer to consider the dimensionality reduction process as a black box, which can be employed directly as a pre-processing step in their systems. To tackle this problem, we propose a methodology for automatic configuration of a group of nonlinear dimensionality reduction methods. Since it facilitates their usage, it makes them more convenient for the research community.

Secondly, despite the huge research effort that has been already dedicated to dimensionality reduction, the majority of the current state of the art approaches ignores or considerably simplify the temporal aspect present in many phenomena. Such approach is clearly inadequate in many real world applications, where usually the analysis of behavioural and dynamic properties of phenomena is much more informative than the description of their states at a certain point in time. As a consequence, the key objective of this thesis is to develop novel dimensionality reduction algorithms which are tailored to time oriented data, i.e. multidimensional time series. Consideration of the time domain during the dimensionality reduction process allows learning more accurate and meaningful models of events which are temporally correlated.

The final objective is to examine practical advantages of the proposed dimensionality reduction approaches by applying them to human motion analysis. Although digital representation of human motion is very complex and high dimensional, we demonstrate that only a few extracted underlying parameters are sufficient to model and discriminate between different human actions regardless of view, speed and motion style.

1.4. Scientific Contribution

This thesis provides significant advances towards a solution of essential problems which are faced by machine learning, pattern recognition and computer vision communities. These contributions originate from our novel and original ideas and are summarised below:

- First, in chapter 2, we give an extensive review of the state of the art in dimensionality reduction with a special attention to computer vision applications. We provide the motivation and the background to the machine learning/pattern recognition task of dimensionality reduction and describe the main directions of research as well as the strengths and weaknesses of different approaches. This chapter can be seen as a knowledge repository about dimensionality reduction and one of the most comprehensive discussions available in the field.
- In chapter 3, we examine thoroughly a family of powerful nonlinear spectral dimensionality reduction methods and review their limitations, i.e. selection of free parameter and lack of generative abilities to unseen examples. Motivated by the personal belief that simplicity of usage is essential to an algorithm popularity, we propose a framework for automatic configuration of spectral dimensionality reduction methods, which overcomes identified weaknesses. As a consequence, this novel framework improves significantly the applicability and performance of spectral methods. The framework has been validated using three main representatives of the spectral family and shows excellent versatility in a range of tasks including human pose recovery.
- Despite of the huge amount of work, which has been devoted to the research in dimensionality reduction (see section for 2.2 overview), the majority of this effort does not take into consideration the dynamic nature of many phenomena. Such static approaches are clearly inappropriate in the context of time

dependent phenomena, where measured features vary continuously over time; thus consecutive observations are expected to be highly correlated. We are convinced that the time domain is a crucial asset of real-world data and thus it is essential to take it into account when modelling such phenomena. To tackle this intellectually and technically challenging problem, in chapter 4, we propose a novel dimensionality reduction method, called Temporal Laplacian Eigenmaps, which takes advantage of spatial and temporal coherency relationships between time series in order to extract the intrinsic parameterisation of a high dimensional time series space regardless of data variations. Our fundamentally different and fresh perspective to the dimensionality reduction problem, which aims at preserving the temporal topology of observed space during dimensionality reduction instead of the traditionally used geometric one, allows us to produce automatically meaningful and generalised low dimensional representations tailored to multivariate time series data. An exhaustive evaluation on a couple of computer vision applications, i.e. pose recovery and action recognition, demonstrates the effectiveness of the proposed methodology for modelling different types of multidimensional time series and its superiority in comparison to the current state of the art approaches.

- To cover adequately the complexity and richness of measured phenomena, tremendous amounts of representative data are often required to learn appropriate data-driven models. Since, in practice, the capture of such amounts of data may be unfeasible, the problem arises about how to generalise known data samples to the entire phenomenon space to obtain a reliable model. Although, several approaches have already been proposed to address this issue (section 2.2.2.3), they either do not consider or radically simplify the temporal aspect of high dimensional data. In chapter 5, we deal with this scientific challenge in the context of multidimensional time series data. Inspired by the spatio-temporal constraints of Temporal Laplacian Eigenmaps, we formulate a generative nonlinear dimensionality reduction algorithm, which is called Spatio-Temporal Gaussian Process Latent Variable Model. Our innovative method is capable of approximating a compact underlying distribution of time series space in the presence of data variations. As a result, a core pattern of multivariate time series is extracted with associated uncertainties of prediction. A comprehensive evaluation, using different types of multidimensional time series, confirms the superiority of this concept in modelling and classification of human motions.

- Finally, in chapter 6, we investigate further a practical aspect of our contributions from chapters 4 and 5 in a challenging real-life computer vision task of view-independent action recognition. Any action recognition system usually involves a combination of methods from the vision and machine learning realms. The vision part is responsible for the extraction of representative and relevant features from action videos, whereas the machine learning one creates actual semantic models of actions. Our contribution falls strictly in the learning domain. We devise two powerful variants of temporally constrained action descriptors, so called action manifolds, which encapsulate style, view and speed variability of any type of motion in a compact and consistent low dimensional representation. The key property of the introduced descriptors is their generalisation potential to previously unobserved motions regardless of view. Despite using basic vision algorithms for video processing, promising experimental results match the performance of the most accurate action recognition methods, while overcoming some of their limitations.

1.5. Thesis Outline

The body of the thesis is divided into seven chapters.

In this chapter, we have put our research into context and summarised our contributions to science.

In chapter 2, first, a formal definition and discussion of the dimensionality reduction problem are presented. They are followed by a detailed review of the state of the art in the dimensionality reduction field.

Then, in chapter 3, we propose a framework for automatic configuration of spectral dimensionality reduction methods. This chapter is based on work that was first published at the *Workshop on Visual Surveillance* during *IEEE International Conference on Computer Vision* (*VS* 2009) [Lewandowski et al., 2009], and then it was extensively extended for a journal version in the *Pattern Recognition Letters* (*PRL 2010*) [Lewandowski et al., 2010a].

In turn, in chapter 4, we introduce a novel nonlinear dimensionality reduction method, called Temporal Laplacian Eigenmaps, which is tailored to modelling multidimensional time series. The chapter is based on work presented in the *IAPR International Conference on Pattern Recognition* (*ICPR 2010*) [Lewandowski et al., 2010c] and additional experiments which were conducted later with a view to an ongoing journal paper preparation.

The next chapter 5 describes a generative nonlinear dimensionality reduction approach for modelling uncertainty of multidimensional time series space, called Spatio-Temporal Gaussian Process Latent Variable Model. The chapter is based on work published in the *Pacific-Asia Conference on Knowledge Discovery and Data Mining* (*PAKDD 2011*) [Lewandowski et al., 2011] and additional experiments which were conduced later with a view to an ongoing journal paper preparation.

In chapter 6, a practical application of our contribution to view-independent action recognition is presented. This chapter is based mainly on work published in the INRIA *European Conference on Computer Vision* (*ECCV 2010*)

[Lewandowski et al., 2010b]. Some elements are also presented in the *PAKDD 2011* conference paper [Lewandowski et al., 2011].

Finally, chapter 7 concludes the undertaken research, summarises limitations and highlights potential directions for future work.

Each contribution chapter, i.e. 3, 4, 5, 6, shares a similar format. They start with a statement and scope of problem in the introductory section (3.1, 4.1, 5.1, 6.1), which will be addressed in that chapter. Then, closely related work, which has already been carried out by research community (3.2, 4.3, 5.2, 6.2), is discussed. Next, a developed solution to the problem is presented (3.3, 4.4, 5.3, 6.4), followed by evaluation of the proposed methodology (3.4, 4.5, 5.4, 6.5). Most of evaluation sections begin with a description of datasets which are used in experiments (3.4.1, 4.5.1, 5.4.1) followed by the experimental setup (3.4.2.1, 4.5.2.1, 5.4.2.1) and an explanation of the performed experiments (3.4.2.2, 4.5.2.2, 5.4.2.2). Then, results of each experiment are presented and discussed. The broader discussion, which summarises all experiments, is provided in the last subsection (3.4.7, 4.5.8, 5.4.5). Finally, all chapters finish with a summary of the contribution (3.5, 4.6, 5.5, 6.6) with respect to the stated objective in the corresponding introduction section (3.1, 4.1, 5.1, 6.1). Note that, it is advised to read chapters in the provided order, since most of them relies on the previous ones, especially the contribution chapters 4, 5, 6.

2. State of the Art Review

2.1. Introduction

This chapter introduces and defines the problem of dimensionality reduction and related topics such as the curse of dimensionality and intrinsic dimensionality (section 2.2). Then, a comprehensive survey of the dimensionality reduction field is provided to show the full evolution of the concept from preliminary relatively simple feature selection approaches (section 2.2.1) to nowadays powerful and popular feature extraction methods (section 2.2.2). Afterwards, two computer vision fields, i.e. human pose recovery and action recognition, are overviewed to establish the general background used for the evaluation of our contributions (section 2.3). In particular, we discuss the current state of the art regarding the usage of dimensionality reduction transformations in human motion analysis (sections 2.3.2.2.3 and 2.3.3.2.4). In addition, this chapter introduces some basic notations and conventions for dimensionality reduction transformations, which are exploited in the rest of the dissertation.

2.2. Dimensionality Reduction

Analysis of multidimensional data often suffers from an effect known as the ‘curse of dimensionality’. The term ‘curse of dimensionality’ was coined by [Bellman, 1961] and refers to the fact that in the absence of simplifying assumptions, the number of data samples required to estimate a function of several variables to a given accuracy (i.e., to get a reasonable low-variance estimate) on a given domain grows exponentially with the number of dimensions [Lee and Verleysen, 2007]. To illustrate the problem, let’s consider the 3-class pattern recognition problem presented by [Gutierrez-Osuna, 2006] (Figure 2.1). First, the 1-dimensional space is divided into B uniform bins with 3 samples each (Figure 2.1, step 1). Each bin is labelled by majority voting using training labels, so that a new sample is classified by assigning the label of the corresponding bin to it. Since there is significant overlap among the classes, a second dimension is incorporated to improve separability (Figure 2.1, step 2). At this point, if we decide to maintain the number of training examples, then a very sparse 2D scatter plot is obtained (Figure 2.1, step 2a). Otherwise, if we choose to keep a constant density of sampling per bin, then the number of training examples increases exponentially to 27 (Figure 2.1, step 2b). As a consequence, a new sample may be unclassified if it is located in an empty bin; this can be solved by adding more training samples to cover evenly the entire space. Adding another dimension makes these problems worse (Figure 2.1, step 3), since now the 3D scatter plot is almost empty (Figure 2.1, step 3a) or at least 81 samples are required (Figure 2.1, step 2b). This phenomenon is known as the curse of dimensionality. Another basic illustration of dimensionality curse problem can be found in [Trunk, 1979].

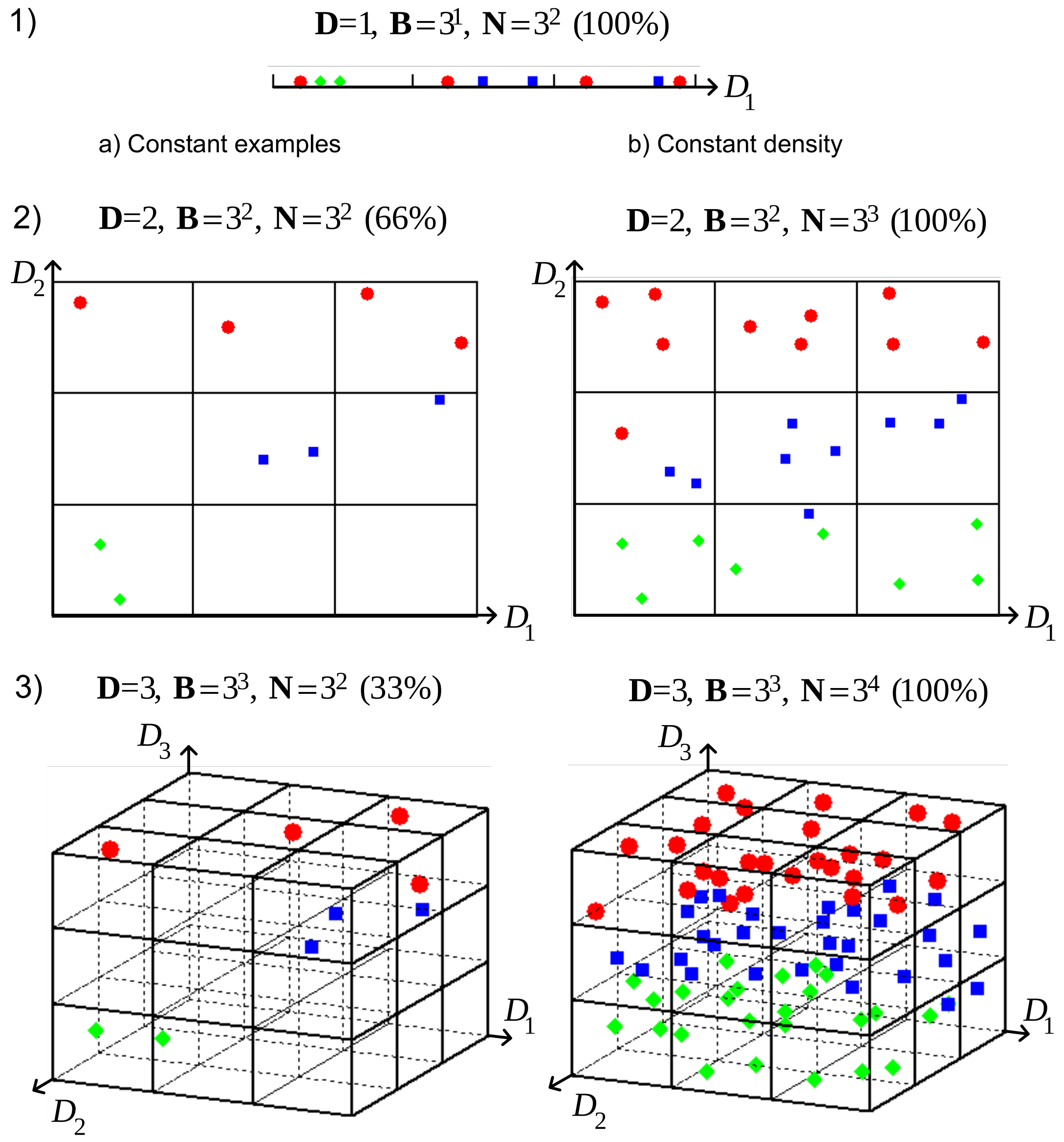


Figure 2.1. Illustration of dimensionality curse in a toy pattern recognition problem where: D – dimensionality, B – number of bins, N – number of samples, (x% = N/B*100) – space denseness.

A few counterintuitive properties of the high dimensional spaces are responsible for the dimensionality curse [Jimenez and Landgrebe, 1998]. First, most of data points of high dimensional spaces reside in unexpected places, such as corners for hypercube or in a thin shell near outer boundary of hypershpere and hyperellipsoid [Scott and Thompson, 1983, Jimenez and Landgrebe, 1998, Weber et al., 1998]. This implies that the centre becomes far less important and the high dimensional space is inherently sparse (Figure 2.1, step 2a, 3a). This property is known as the 'empty space phenomenon' [Scott and Thompson, 1983]. A further undesired property is that the size of data samples required to adequately cover a hyper-volume to perform 'satisfactory' data analysis increases exponentially with dimensionality (Figure 2.1, step 2b, 3b).

In addition to the curse of dimensionality, another complexity induced by analysing high dimensional spaces is the 'nearest neighbour problem', which is defined as [Beyer et al., 1999]:

> *Given a collection of data points and a query point in a* **D**-*dimensional metric space, find the data point that is closest to the query point.*

For a given query point, it has been shown that the distance to the nearest neighbour tends to be similar to distance to the farthest neighbour as dimensionality increases [Beyer et al., 1999]. This is particularly an issue when using the Manhattan norm (L_1), the Euclidean norm (L_2) and the general k-norm L_k [Hinneburg et al., 2000]. This effect is known as the 'concentration phenomenon' [Beyer et al., 1999, Francois et al., 2007]. In addition, this distance grows steadily with dimensionality and decreases only marginally as the number of points increases [Weber et al., 1998]. [Francois et al., 2007] proved formally that the concentration phenomenum is an intrinsic property of the norm when measuring high-dimensional data similarity even when an infinite number of data points are considered.

In general, all these properties manifest themselves by a decrease of overall accuracy of system according to the statistical learning theory approach [Vapnik, 1998]. As a consequence, for a given dataset, there is a maximum number of dimensions above which the quality of data analysis degrades when the number of training samples is small relative to dimensionality (Figure 2.2) [Devijver and Kittler, 1982, Bishop, 1995, Jain and Zongker, 1997, Weber et al., 1998, Jain et al., 2000, Korn et al., 2001, Hua et al., 2009]. This paradoxical behaviour is referred to as the ‘peaking phenomenon’ [Devijver and Kittler, 1982]. Dimensionality reduction attempts to overcome effectively these issues without losing significant information in terms of data intrinsic structure and properties. Moreover, it facilitates classification, visualisation, clustering and compression of high dimensional data.

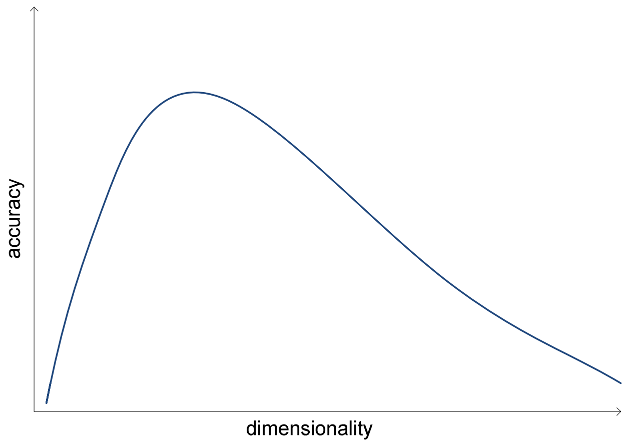


Figure 2.2. Dimensionality versus accuracy of a multidimensional data analysis.

Given a set of data points in a high-dimensional space, dimensionality reduction is defined as the process of discovery of a meaningful and compact representation of reduced dimensionality to obtain more informative, descriptive and practical data representation for further analysis. This process is achieved by eliminating redundancies and irrelevant information present in data while ensuring the maximum possible preservation of information [Jain et al., 2000, van der Maaten et al., 2009].

Ideally, the reduced dimensionality should correspond to the intrinsic dimensionality of the data. This can be understood as the minimum number of independent variables needed to explain satisfactory the observed properties of the data. More formally, from a geometrical point of view, a dataset $Y \subset \mathbb{R}^{\mathbf{D}}$ is said to have intrinsic dimensionality (ID) equal to $\mathbf{d}$ if its elements lie entirely within a d-dimensional subspace of $\mathbb{R}^{\mathbf{D}}$ [Fukunaga, 1982, Fukunaga, 1990].

In general, dimensionality reduction can be performed by either feature selection or feature extraction. Feature selection methods select the most discriminative key features among those given; therefore low-dimensional data representations possess a physical meaning. Alternatively, feature extraction approaches create new informative features by applying certain operations to the original features. In other words, the new projection of data is created based on transformation or combination of the original feature set.

2.2.1. Feature selection

Feature selection is based on the ‘principle of parsimony’ [Bell and Wang, 2000]. This says that, we prefer the model with the smallest possible number of parameters that adequately represents the data. For this reason feature selection methods aim at selecting an optimal subset of relevant features from a given set of original candidate features [Devijver and Kittler, 1982]. Here, a feature vector is defined as a one dimension of a data samples set (see figure 2.3). A definition of the optimal subset and various notions of relevance in a context of feature selection framework are given in [Kohavi and John, 1997, Blum and Langley, 1997]. Using [Jain and Zongker, 1997] notation, given a set of features $Y = \{y_j \mid y_j \in \mathbb{R}^{\mathbf{n}}, j = 1..\mathbf{D}\}$ the goal of a feature selector is to find a subset $X \subseteq Y$ ($x_j \in Y$) which optimises a particular evaluation criterion J and cardinality of set X is $\mathbf{d}$:

$$J(X) = \max_{Z \subseteq Y, \overline{\overline{Z}} = d} J(Z) \tag{2.1}$$

where a higher value of J indicates a better feature subset (Figure 2.3) and $\mathbf{d} < \mathbf{D}$ (often $\mathbf{d} << \mathbf{D}$).

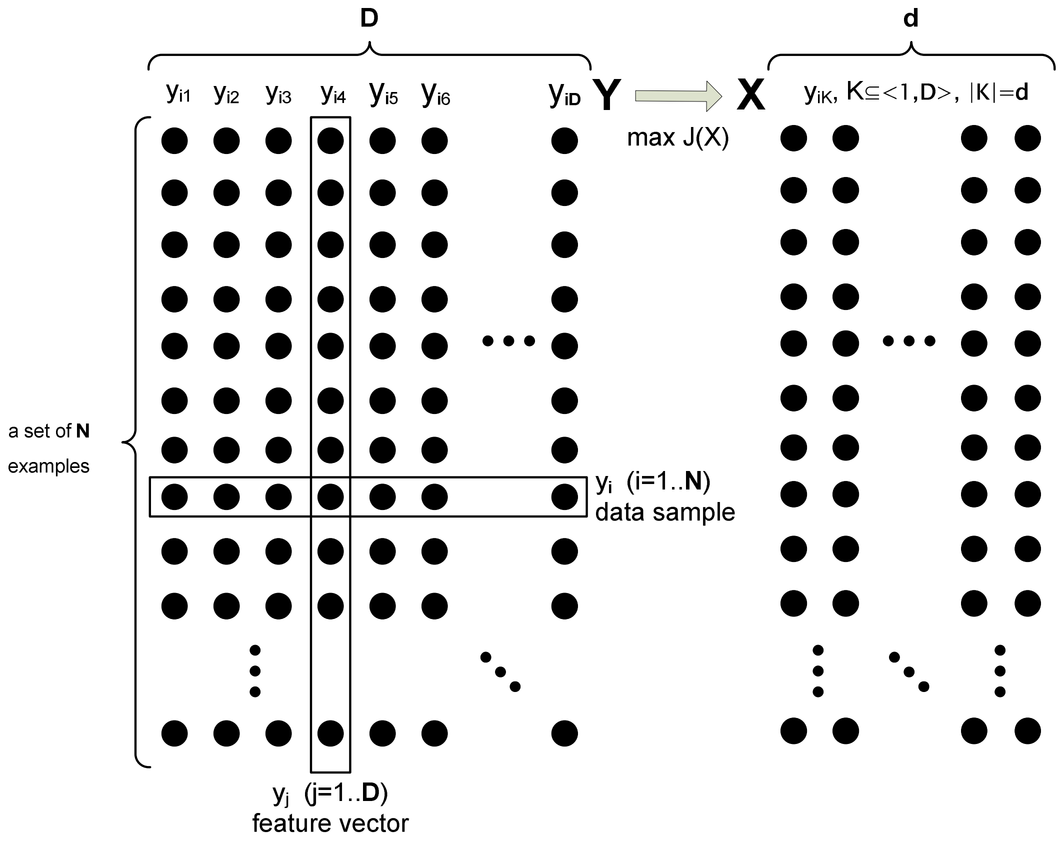

D
d
y_{i1} y_{i2} y_{i3} y_{i4} y_{i5} y_{i6} y_{iD}
Y
X
y_{iK}, K⊆<1,D>, |K|=d
max J(X)
a set of N examples
y_i (i=1..N) data sample
y_j (j=1..D) feature vector

Figure 2.3. Principle of feature selection and notations.

There are four basic steps in a typical feature selection method (Figure 2.4) [Dash and Liu, 1997, Dash and Liu, 2003, Liu and Yu, 2005]:

- a generation procedure to generate the next candidate subset for evaluation,
- an evaluation function to evaluate the candidate subset,
- a stopping criterion to decide when to stop, and
- a validation procedure to check whether the subset is valid.

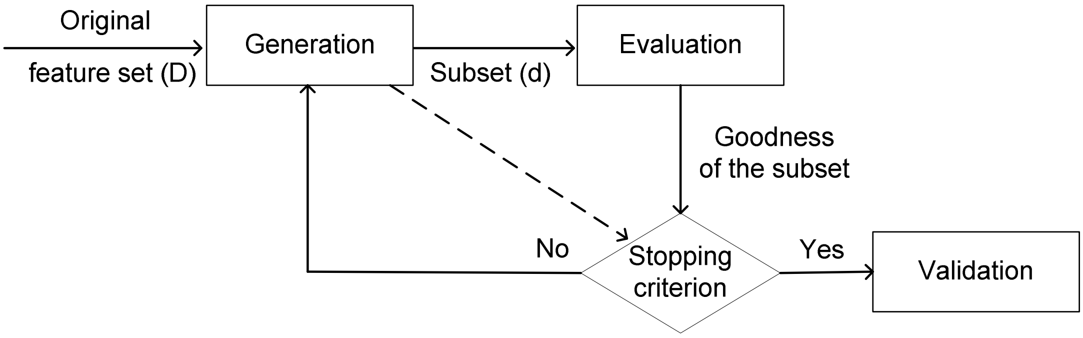


Figure 2.4. Four basic steps of the feature selection process.

A comprehensive review of feature selection algorithms in different fields can be found in [Jain and Zongker, 1997, Jain et al., 2000, Guyon and Elisseeff, 2003, Liu and Yu, 2005, Saeys et al., 2007, Hua et al., 2009].

2.2.1.1.Subset Generation

The generation procedure is essentially a heuristic search, with each state in the search space specifying a candidate subset of features for evaluation. The search process starts with no features, with all features, or with a random subset of features. Since for a data set with **N** features, there exist $2^{\mathbf{N}}$ candidate subsets, different search strategies have been explored.

2.2.1.1.1. Complete Search

In the case of exhaustive search all possible combination of subsets $\mathbf{d} \leftrightarrow \mathbf{D}$ are evaluated like in Focus method [Almuallim and Dietterich, 1994]. However, the exhaustive search is impractical even for moderate sizes of **d** and **D** because of exponentially increases of the size of the search space [Jain and Zongker, 1997, Liu and Yu, 2005]. In order to avoid the enormous calculations of the exhaustive method, different heuristic functions are used to perform non-exhaustive search on a smaller number of subsets by using, for example, Branch & Bound algorithms [Narendra and Fukunaga, 1977, Yu and Yuan, 1993, Chen, 2003, Somol et al., 2004, Cao and Saha, 2005] (B&B). Complete search guarantees to find the optimal subset according to the evaluation criterion which is used [Guyon and Elisseeff, 2003].

2.2.1.1.2. Sequential Search

The simplest sequential search technique is hill climbing in a search tree (also called greedy search). Here a feature’s subset iteratively grows (forward selection) or shrinks (backward elimination) by adding/removing the best descendant features.

It can also start from both ends and iteratively add and remove features simultaneously (bidirectional selection) [Huan and Hiroshi, 1998]. The process terminates when there is no improvement over a current subset. Best-first search [Russell and Norvig, 2003] is a more general and robust method than hill climbing. Instead of using only current descendant features, the most promising feature is selected from all unexpanded nodes which have been generated.

In both search engines, quality of a feature is determined according to a specified rule. These search strategies are computationally advantageous and robust against over fitting in producing deterministic results; however they may miss optimal subsets.

2.2.1.1.3. Random Search

Following the sequential search, a random process is injected into the above classical sequential approaches; this process is similar to simulated annealing [Doak, 1992, Meiri and Zahavi, 2006] or genetic algorithms methods [Siedlecki and Sklansky, 1989, Vafaie and De Jong, 1993, Raymer et al., 2000, Oh et al., 2004]. The random subsets are derived either from Monte Carlo sampling or Random mutation hill climbing [Skalak, 1994]. Alternatively, each subset is produced in a completely random manner like in the Las Vegas algorithm [Brassard and Bratley, 1996]. For all these approaches, the incorporation of randomness helps to escape local optima in the search space; however this can still result in a stochastic suboptimal solution.

2.2.1.2. Subset Evaluation

Another dominating factor in designing a feature selection algorithm is the evaluation function which is used to determine the quality of a candidate subset. A new subset replaces a previous one, only if its evaluation score is better. According to [Blum and Langley, 1997], the evaluation criteria are broadly grouped based on their dependency on mining algorithms (also referred to as inductive or machine learning algorithms) that will finally be applied on the selected feature subset (e.g. classification or clustering).

Popular induction algorithms include decision trees [Quinlan, 1993, Forman, 2003, Draminski et al., 2008], naive Bayes classifiers [Kohavi and John, 1997, Forman, 2003, Ding and Peng, 2003, Peng et al., 2005, Draminski et al., 2008], nearest neighbour classifiers [Draminski et al., 2008, Hua et al., 2009], discriminant analysis [Peng et al., 2005, Hua et al., 2009], least-square linear predictors [Guyon and Elisseeff, 2003], and support vector machines [Forman, 2003, Ding and Peng, 2003, Peng et al., 2005, Saeys et al., 2007, Draminski et al., 2008, Forman, 2008, Lin et al., 2008, Rodriguez-Lujan et al., 2010, Gheyas and Smith, 2010].

2.2.1.2.1. Filter

Filter techniques assess the relevance of features by looking only at the intrinsic characteristics of the training data without involving any inductive algorithm. In most cases a feature relevance score is calculated, and low-scoring irrelevant features are filtered out (Figure 2.5). Afterwards, the best subset of features is passed as input to the mining algorithm. By definition, filter methods are independent of the chosen inductive algorithm; therefore typically they are based on certain statistical criteria, so called measures, to rank subsets.

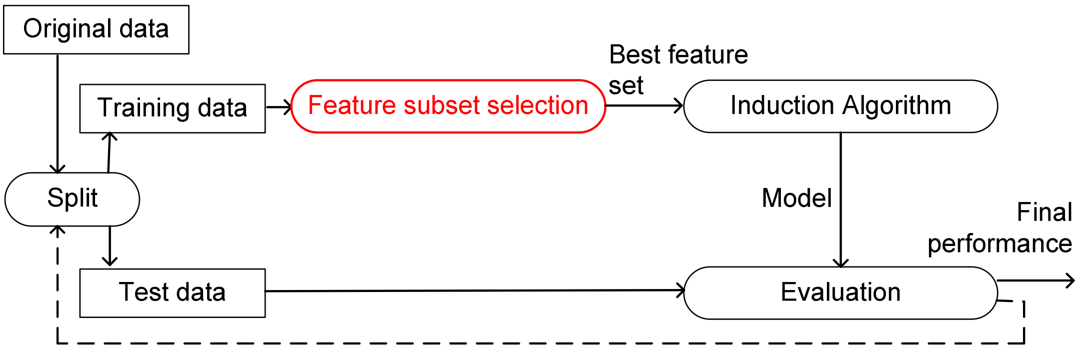


Figure 2.5. The filter approach for feature subset selection.

2.2.1.2.1.1. Separability Measures (distance measures)

The most straightforward method for the feature selection is an exhaustive ranking of each individual feature in a dataset, independently of the context of others. Commonly known ranking metrics for this purpose include Chi-Squared test [Yang and Pedersen, 1997] (Chi), Information Gain [Yang and Pedersen, 1997] (IG) or Bi-Normal Separation [Forman, 2003, Forman, 2008] (BNS). More criteria can be found in [Guyon and Elisseeff, 2003].

In contrast, to improve computationally performance, Branch & Bound algorithm [Narendra and Fukunaga, 1977] (B&B) performs non-exhaustive search and uses intermediate results to obtain bounds on the final criterion value. The key assumption of the algorithm is an adaptation of the monotonicity principle for the criterion function J , i.e.:

$$J(A \cup B) \geq J(A), \ \forall A, B \subseteq Y \tag{2.2}$$

This means that the addition of new features to a current subset must result in an increase of performance according to the evaluation criterion. B&B starts from the full set and removes features using a depth-first strategy. The subsets are coded as bit-strings, i.e. as sequences of zeros and ones which correspond to the absence or presence of a feature in the subset. Computational complexity of search process is improved further by exploiting minimum solution tree [Yu and Yuan, 1993], asymmetrical solution tree [Chen, 2003], approximating values of evaluation function by predictions [Somol et al., 2004] or eventually best-first search approach [Cao and Saha, 2005]. Typical choices of monotonicity criterion include: Bhattacharyya distance [Chen, 2003, Somol et al., 2004], discriminant functions [Chen, 2003], Divergence distance [Somol et al., 2004], Patrick-Fischer distance [Somol et al., 2004] and the minimum Hankel singular values [Cao and Saha, 2005].

On the other hand, the sequential Relief algorithm [Kira and Rendell, 1992] was inspired by instance-based learning [Aha et al., 1991]. The key idea behind Relief method is to assign a relevance weight to each feature, which is meant to denote the relevance of the feature to the target concept. It samples instances randomly from the training set to update relevance values. In context of classification, the relevance estimation of each feature is based on the difference between the selected instance and the two nearest instances of the same and opposite classes. Since the original formulation of Relief can only be applied on binary problems [Kira and Rendell, 1992], [Kononenko, 1994] proposes a generalisation to univariate case. Relief evaluates usefulness of features according to the relevance level [Kira and Rendell, 1992].

2.2.1.2.1.2. Information-theoretic Measures

Information measures typically determine the information gain from features.

In Sequential Forward Generation [Huan and Hiroshi, 1998] (SFG), the algorithm starts with an empty set and adds one feature from the original set at a time. At each round of selection, the best feature is chosen according to fitness function.

The Decision Tree Method [Cardie, 1993] (DTM) employs a similar idea to generate feature subsets, however the search is performed with the decision tree algorithm [Quinlan, 1993] and candidate subsets are evaluated according to entropy criterion.

Monte Carlo feature selection [Draminski et al., 2008] (MCFS) is an example of random information algorithm. It provides an objective measure of relative importance of each feature for a particular classification task regardless of the classifier that will be used. This is achieved by taking into account interdependencies between the features; a feature may prove to be informative only in conjunction with some other features. Importance of a feature is measured via intensive use of classification trees.

2.2.1.2.1.3. Dependency Measures

Dependency measures are also known as correlation or similarity measures. They quantify the ability to predict the value of one variable from the value of another variable.

For instance, in Correlation-based Feature Selection [Hall, 2000] (CFS), a linear Pearson's correlation heuristic is exploited to evaluate the merit of a subset of features rather than individual features like in Relief. This heuristic takes into account the usefulness of individual features for predicting the target concept along with the level of intercorrelation among them. A feature is considered to be a good one if it is relevant to the target concept but is not redundant to any of the other relevant features. A goodness of measure is expressed by a correlation between features. [Yu and Liu, 2003] introduces a concept of predominant correlation and predominant feature to formulate Fast Correlation-Based Filter (FCBF) which allows to reduce time complexity of CFS. In addition, different correlation measures are incorporated into CFS, for instance the Kolmogorov-Smirnov correlation coefficient [Biesiada and Duch, 2005] or Pearson's chi-squared test [Biesiada and Duch, 2007].

In contrast to standard CFS, a maximal relevance [Peng et al., 2005] (MaxRel) and minimal-redundancy-maximal relevance [Ding and Peng, 2003, Peng et al., 2005] (mRMR) frameworks use nonlinear correlation between features in a heuristic search. MaxRel maximises relevance condition to obtain an optimal subset of original features, whereas mRMR also minimises redundancy condition simultaneously. The idea of maximum relevance is to select the features such that they are mutually maximally similar, while the minimum redundancy ensures selection of mutually exclusive features. As a result, minimal redundancy will make the feature set more representative of the entire dataset. Both conditions are defined in terms of mutual information measure [Cover and Thomas, 1991, Ding and Peng, 2003, Peng et al., 2005]. Recently, [Zhang et al., 2008] proposed a two-stage selection algorithm combining the best properties of mRMR and ReliefF to select a compact yet effective gene subset from the candidate set.

Given the prohibitive cost of considering all possible subsets of features, the MaxRel and mRMR algorithms must select features greedily and optimise evaluation criterion with features chosen in previous steps. The smaller time complexity with comparable accuracy is achieved by Quadratic Programming Feature Selection [Rodriguez-Lujan et al., 2010] (QPFS) even though it ranks all training features according to mutual information or Pearson's correlation coefficient as a similarity measure. The feature selection is formulated as the quadratic programming optimisation problem which takes advantage of Nyström approximation to reduce the computational complexity.

2.2.1.2.1.4. Consistency Measures

Consistency measures are defined by inconsistency rate for a given feature set: $I_R(A) \leq \delta$ where δ is a user given inconsistency rate threshold [Dash and Liu, 2003]. An inconsistency is defined as a sum of all the inconsistency counts over all patterns of the feature subset; see [Dash and Liu, 2003] for more details. The best subset satisfies the consistency criterion. Consistency measures are usually used for classification. For instance, Focus method [Almuallim and Dietterich, 1994] exhaustively examines all subsets of features and selects minimal subset of features that is sufficient to determine a value of class label for all instances in a training set. This preference for a small set of features is referred to as Min-Features bias [Almuallim and Dietterich, 1994]. [Liul et al., 1998] proposes an automated interpretation of standard Branch & Bound algorithm (AB&B) where the bound is set to the inconsistency rate of the original feature set. In contrast, the Las Vegas Filter [Liu and Setiono, 1996] (LVF) randomly searches the space of subsets and makes probabilistic choices to guide the search more quickly towards an optimal solution. Method adopts the inconsistency rate to find a minimum number of features that separate classes as consistently as the full set of features can. An inconsistency is defined as two instances having the same feature values but different classes. To improve processing performance of AB&B and LVF, a hybrid Quick Branch & Bound (QBB) was proposed [Huan and Hiroshi, 1998]. It runs first LVF and afterwards AB&B on pre-processed smaller subsets of features [Dash and Liu, 2003].

2.2.1.2.1.5. Summary of Filtering Selection Methods

Advantages of all discussed filter techniques are that they easily scale to very high-dimensional datasets. Moreover, they are computationally simple and efficient, and the most importantly they are independent of the inductive algorithm. As a result, feature selection needs to be performed only once, and then various classifiers can be evaluated.

A common disadvantage is that they ignore the interaction with the mining algorithm (the search in the feature subset space is separated from the search in the hypothesis space) which may lead to worse performance [Guyon and Elisseeff, 2003, Liu and Yu, 2005, Saeys et al., 2007, Hua et al., 2009] in comparison to wrapper methods (see next section). In addition, most of the proposed techniques are univariate (Chi, IG, BNS, Relief, B&B, SFG, DTM, Focus, AB&B, LVF, and QBB). This means that each feature is considered separately, thereby they lack in robustness against interactions among features and feature redundancy. In order to address the problem of ignoring feature dependencies, some multivariate filter techniques were introduced (MCFS, CFS, FCBF, MaxRel, mRMR, QPFS) which aim at the incorporation of feature dependencies to some degree. Finally, filters tend to select the full feature set as the optimal solution, as a result the threshold for rankings must be chosen arbitrary by a user to select only truly important features and exclude noise. Unfortunately, there is no general rule how to set this crucial parameter.

2.2.1.2.2. Wrapper

The wrapper methodology, popularised by [Kohavi and John, 1997], offers a conceptually simple, powerful and universal alternative to the problem of feature selection. Namely, it requires one predetermined machine learning algorithm and uses its performance as the evaluation criterion to determine which features are selected (Figure 2.6). In fact, the inductive algorithm is considered to be a perfect “black box”: no knowledge about it is required. In this setup, a search procedure in the space of all possible feature subsets is defined as ‘wrapper’ around the data mining model, which repeatedly calls the induction algorithm as a subroutine to evaluate quality of various subsets of features. The evaluation of subsets is performed with an internal validation set obtained by a hold-out or k-fold cross validation schema. The feature subset with the highest evaluation is chosen as the final set on which the induction algorithm is run and final performance of the system is calculated.

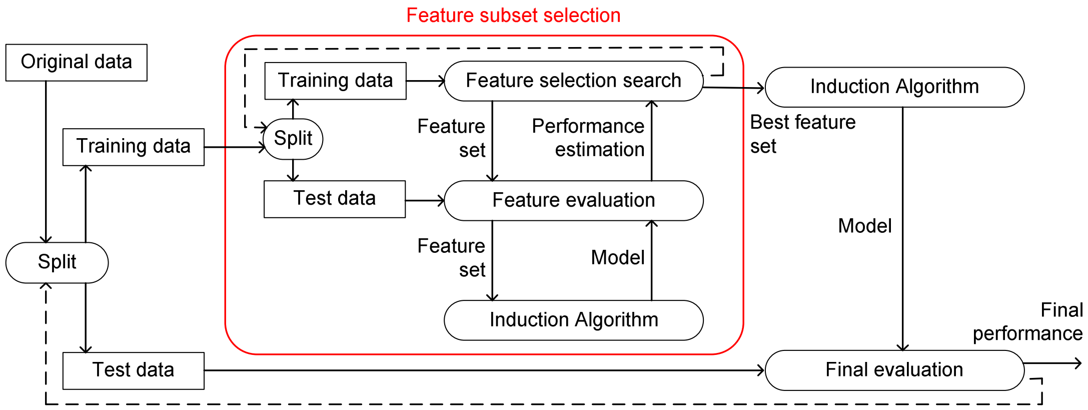


Figure 2.6. The wrapper approach for feature subset selection. The induction algorithm is used as a "black box" by the subset selection algorithm.

2.2.1.2.2.1. Sequential Search Strategy

Sequential Backward Elimination [Green, 1963, Kittler, 1978, Cotter et al., 2001] (SBE) and Sequential Forward Selection [Whitney, 1971, Kittler, 1978, Colak and Isik, 2003] (SFS) are the two most commonly used wrapper methods that exploit a greedy hill-climbing search strategy. SBE starts with the set of all features and progressively eliminates the least promising ones, whereas SFS does the opposite. Similarly, the termination criteria are contrary: while SBE stops if the evaluated performance drops below a given threshold, SFS adds features until performance stops improving. The main drawback of these methods is that they cannot alter already chosen subsets. It means that if a feature is retained (resp. deleted), it cannot be discarded from (resp. reselected to) the resulting subset. As a result both SFS and SBE can easily be trapped into local minima. Moreover, they produce 'nested subsets', i.e. the subset of the four best features chosen must contain the subset of the three best features, and so on. It has been shown in practice that the actual best four features may not contain any of the actual best three features [Jain and Zongker, 1997].

To overcome these problems [Pudil et al., 1994] proposes Sequential Forward/Backward Floating Search (SFFS, SFBS) that performs a greedy search with the ability to backtrack after each sequential step, so that it can locate a better subset. Adaptive versions of the floating search methods were proposed by [Somol et al., 1999]. The adaptive methods (ASFFS, ASFBS) consider adding or removing variable number of features in each sequential step to search for a better subset depending on closeness to the desired number of features d . Recently [Nakariyakul and Casasent, 2009] has extended Sequential Forward Floating Search method by adding a checking procedure whether removing any feature in the currently selected feature subset and adding a new one at each sequential step can improve the current feature subset.

2.2.1.2.2.2. Random Search Strategy

Genetic algorithms [Siedlecki and Sklansky, 1989, Vafaie and De Jong, 1993, Yang and Honavar, 1998, Raymer et al., 2000, Oh et al., 2004] and simulated annealing [Metropolis et al., 1953, Kirkpatrick et al., 1983, Doak, 1992, Meiri and Zahavi, 2006] are stochastic methods for feature subset selection which belong to the class of Monte Carlo algorithms [Fishman, 1996, Rubinstein and Kroese, 2008]. These two classes of techniques are based on the assumption that large domains of data are organised and can evolve to simulate specific processes occurring in nature. Genetic algorithms (GA) are inspired by Darwinian biological principles of evolution and natural selection, where simulated annealing (SA) has the rough physical analogous to the annealing process in metallurgy.

Application of GA for feature selection was inspired by [Siedlecki and Sklansky, 1989] who represents a feature subset as a fixed length binary string (a so-called chromosome), where the value of each position in the string represents the presence or absence of a particular feature. The length of the binary string corresponds to the total number of available features. The algorithm starts with an initial random population of subsets. Afterwards iteratively each chromosome is evaluated on the basis of its overall fitness with respect to the given application domain. The high performing chromosomes will survive and breed into the next generation. The next generation of subsets is formed by using two main genetic operators, i.e. crossover and mutation [Holland, 1975]. The crossover operation is responsible for mixing random parts of two different parent chromosomes to create two new offspring, whereas mutation randomly changes components of a single parent to insert new information into the population. This population of competing solutions evolves in parallel over time. Eventually, it converges to an optimal chromosome since the best features are inherited during the evolutionary process to the next generations with respect to the given goal.

The major advantage of GA are their rapid convergence [Gheyas and Smith, 2010] however the combination of crossover and a low fixed mutation rate may still trap the search in a local minimum [Gheyas and Smith, 2010]. GA can deal with large search spaces efficiently and proves to obtain closer suboptimal solution to global optimum in comparison to greedy sequential approaches, especially with the increase of interactions among features [Vafaie and De Jong, 1993, Yang and Honavar, 1998]. Moreover, they obtain better performance and parsimonious in the number of features required to achieve that accuracy [Raymer et al., 2000]. To reduce time complexity, [Oh et al., 2004] introduces hybrid GA where crossover and mutation operations are followed by local search operations.

Simulated annealing (SA) is an iterative, adaptive and probabilistic method initially introduced by [Metropolis et al., 1953] and later popularised by [Kirkpatrick et al., 1983]. It takes random walks through the problem space, where the probability of taking a step is determined by the Metropolis criteria [Metropolis et al., 1953]. Application of simulated annealing for feature selection is motivated by the following simple idea [Haykin, 1998]:

> *When optimising a very large and complex system (i.e. a system with many degrees of freedom), instead of always going downhill, try to go downhill most of the time.*

The simulated annealing algorithm starts with a randomly generated initial subset and attempts to iteratively improve it. At each iteration, the algorithm selects a random ‘neighbouring’ feature and computes the difference in evaluation quality (i.e. energy difference) between the current and candidate subsets. If the new subset is better for a given application, then it is automatically retained. Otherwise, the new subset is accepted with a probability determined by the Metropolis criteria [Metropolis et al., 1953]. Based on the laws of thermodynamics, in most applications a variant of the Boltzmann distribution is used for calculating this probability. It depends on the energy difference and current temperature, i.e. parameter which gradually decreases with the algorithm progress. This changing temperature causes a progressive decline of the probability for accepting the bad new subset By occasionally accepting inferior subsets, the SA algorithm is able to escape local optima. This acceptance is directly related to the temperature, so it is more likely to happen in the beginning of the process and less probable later. As the algorithm progresses, the temperature is gradually reduced until it vanishes and a final solution is obtained.

The strength of SA is good global search ability, whereas its main weakness a slow convergence speed [Gheyas and Smith, 2010]. SA proves its robustness against local minima in [Meiri and Zahavi, 2006] and demonstrates a high evaluation performance in [Lin et al., 2008]. Hide-and-Seek SA [Romeijn and Smith, 1994] extends standard SA by picking a random feature from all feasible regions following a random vector instead of using only neighbouring regions. As a result, Hide-and-Seek SA converges faster and closer to global optimum regardless of how quickly the ‘temperature’ falls to 0.

[Gheyas and Smith, 2010] introduces a hybrid algorithm SAGA which combines the ability to avoid being trapped in a local minimum of SA with a very high rate of convergence of the crossover operator of GA. The SA algorithm here is a mutation-based search approach which corresponds to a long jump in the search space. SAGA shows the best performance over many subset feature selection methods including GA and SA [Gheyas and Smith, 2010].

2.2.1.2.2.3. Summary of Wrapper Selection Methods

The main advantage of wrapper approaches is that they aim at finding features better suited to the predetermined machine learning algorithm resulting in superior performance of the underlying induction algorithm. Moreover, they include the interaction between feature subset search and model selection, and the ability to take into account feature dependencies. In addition, random wrapper methods are less sensitive to local minima.

However, a common drawback of these methods is that they have a higher risk of over fitting than filter techniques. Besides they also tend to be more computationally intensive especially if running induction algorithm has a high computational cost [Kohavi and John, 1997, Blum and Langley, 1997]. Moreover, the solution suffers from the lack of generality, since it is tuned for a specific induction algorithm. Finally, both GA and SA rely on several user determined parameters which may significantly impact the solution. Currently, established rules do not exist for selecting these parameters.

2.2.1.3. Stopping Criterion

The feature selection process should stop when a specified stopping criterion is reached. Some frequently applied stopping criteria include:

- The search completes.

- Some given bound is reached, where a bound can be a minimum number of features or maximum number of iterations.
- The 'probe' method, i.e. whether further addition (or deletion) of any feature deos not produce a better subset.
- A sufficiently good subset is obtained according to some evaluation function.
- Predefined time limit.

2.2.1.4. Validation

The validation procedure is not a part of the feature selection process itself. It tries to test the validity of the selected subset of features. A straightforward way for result validation is to directly measure the result using prior knowledge about the data. In real-world applications, however, such knowledge is usually unavailable. Hence, the validation process relies on some indirect methods which monitor the change of mining performance with the alteration of features (e.g. classification error). This is achieved by simply conducting the "before-and-after" experiment to compare the error rate of the learning algorithm on the full set of features and that learned on the selected subset.

2.2.1.5. Summary of Feature Selection Methods

The selected optimal set of features can be suitable to understand the physical process that generates the patterns, therefore feature selection has proved to be very popular approach in some applications especially in pattern recognition [Jain et al., 2000, Raymer et al., 2000, Oh et al., 2004, Draminski et al., 2008, Lin et al., 2008, Forman, 2008, Hua et al., 2009, Rodriguez-Lujan et al., 2010, Gheyas and Smith, 2010] and bioinformatics [Saeys et al., 2007]. However, the huge disadvantage is that the obtained solutions are always relative to a certain evaluation criterion. Moreover, it is difficult to propose meaningful evaluation criterion in many domains including computer vision, graphics, speech recognition, image processing etc. A brief summary of discussed feature selection methods according to the framework depicted in Figure 2.4 is presented in Table 2.1.

Table 2.1. Categorisation of feature selection algorithms for dimensionality reduction.

			Search strategies		
			Complete	Sequential	Random
Evaluation criteria	Filter	Separability	B&B Chi, IG, BNS	Relief	
		Information		SFG DTM	MCFS
		Dependency	QPFS	CFS, FCBF MaxRel, mRMR	
		Consistency	Focus AB&B		LFV, QBB
	Wrapper	System performance		SFS, SBE SFFS, SBFS	GA SA SAGA

2.2.2. Feature extraction

Feature extraction is defined as the transformation or/and combination of the original multidimensional features in order to generate a completely new set of informative features in a space of fewer dimensions [Backer et al., 1998]. In contrast to feature selection methods, here a feature vector is defined as a data sample. Feature extraction is a powerful alternative to feature selection since it aims at preserving most of the original information in more appropriate low dimensional representation [Jain et al., 2000, van der Maaten et al., 2009].

Given a space of features $Y = \left\{ y_i \mid y_i \in \mathbb{R}^{\mathbf{D}}, i = 1..\mathbf{N} \right\}$, the linear or nonlinear transformation function F is defined to map the original feature space into a subspace of reduced dimensionality $X = \left\{ x_i \mid x_i \in \mathbb{R}^{\mathbf{d}}, i = 1..\mathbf{N} \right\}$ (Figure 2.7):

$$\begin{aligned} & F : Y \to X,\ Y \subseteq \mathbb{R}^{\mathbf{D}},\ X \subseteq \mathbb{R}^{\mathbf{d}} \\ & x \leftrightarrow x = F(y) : \mathbb{R}^{\mathbf{D}} \to \mathbb{R}^{\mathbf{d}} \end{aligned} \tag{2.3}$$

where $\mathbf{d} < \mathbf{D}$ (often $\mathbf{d} << \mathbf{D}$). A corresponding reconstruction mapping function f is given by:

$$\begin{aligned} & f : X \to Y,\ X \subseteq \mathbb{R}^{\mathbf{d}},\ Y \subseteq \mathbb{R}^{\mathbf{D}} \\ & y \leftrightarrow y = f(x) : \mathbb{R}^{\mathbf{d}} \to \mathbb{R}^{\mathbf{D}} \end{aligned} \tag{2.4}$$

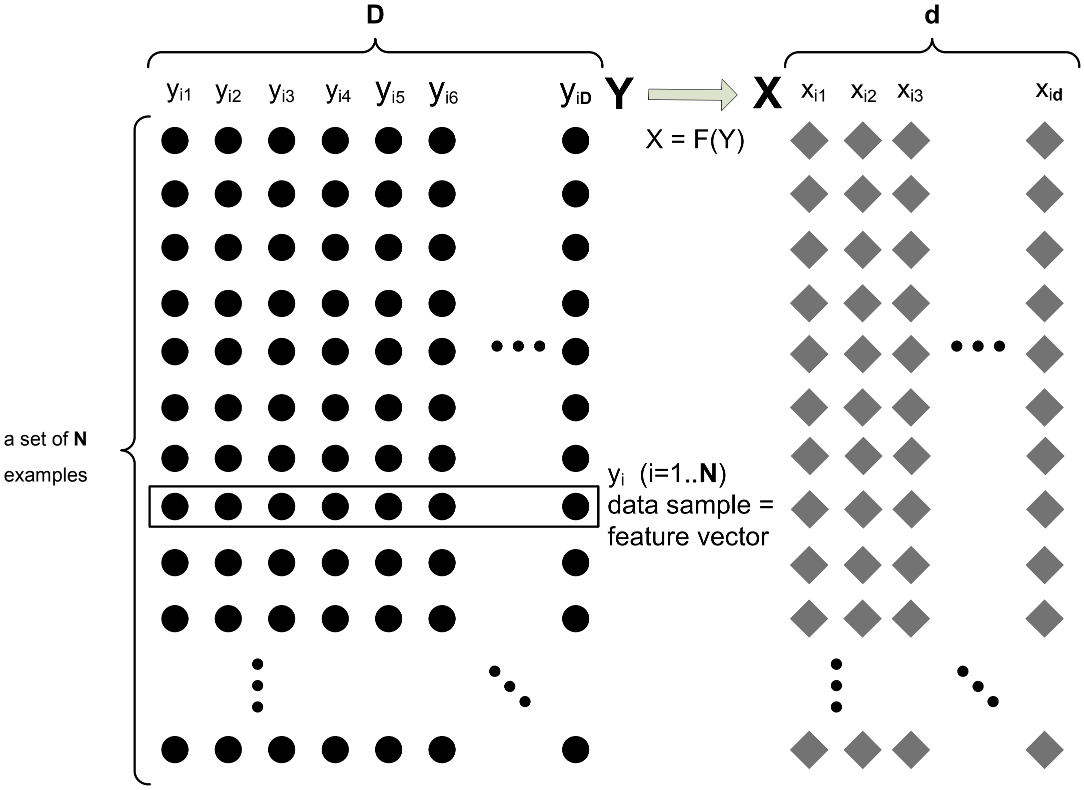


Figure 2.7. Principle of the feature extraction and notations.

The fundamental assumption that justifies feature extraction is that the high dimensional data is actually distributed, at least approximately, on a manifold of smaller dimension than the data space (for the basic terminology and explanation of manifold concept please see section 2.2.2.1). As a consequence, the objective of dimensionality reduction is to uncover this embedded manifold structure from the high dimensional data space. Solving this problem is referred to as manifold learning, since the task is to “learn” unknown geometry of a manifold from a set of points [Camastra and Vinciarelli, 2008].

An overview of feature extraction methods is depicted in Figure 2.8. A much broader review and comparison of deterministic frameworks can be found in [van der Maaten et al., 2009], where probabilistic frameworks are the main concern of work [Quirion et al., 2008]. Generally, feature extraction techniques are divided broadly into two categories, i.e. deterministic and probabilistic frameworks. Both categories are further classified into two main classes: linear and non linear methods. Linear methods assume that the data lie approximately on a linear subspace of the high-dimensional data. Since most real datasets are highly nonlinear, linear methods cannot model the curvature and nonlinear structures embedded in most observed spaces. As a consequence, nonlinear methods were proposed to address this issue.

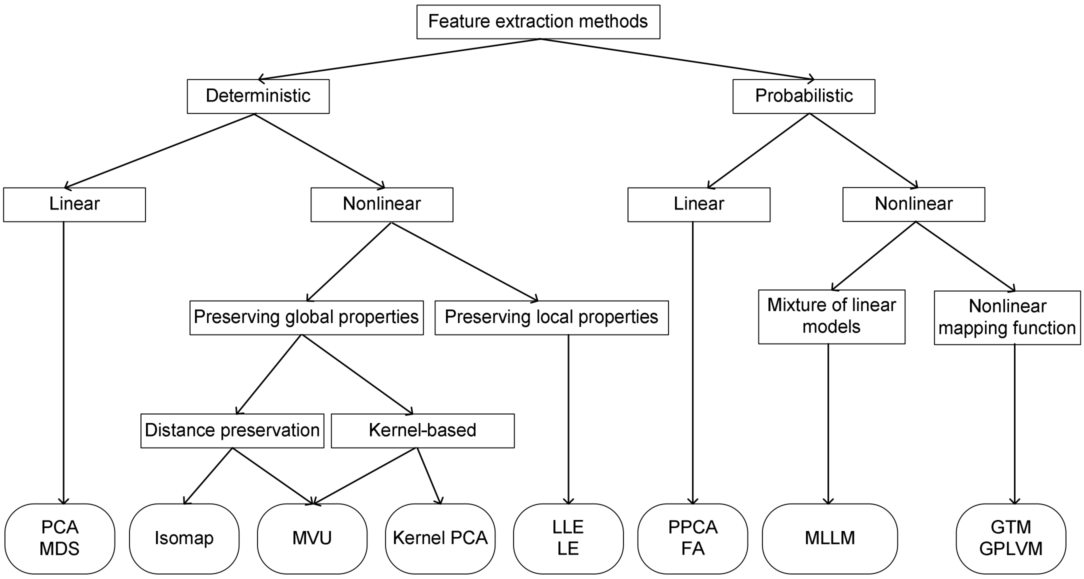


Figure 2.8. Taxonomy of feature extraction algorithms for dimensionality reduction.

2.2.2.1. Manifold Theory

A manifold is a topological space of dimensionality **d** that is locally Euclidean, i.e., around every point, there is a neighbourhood that is topologically the same as the open unit ball in $\mathbb{R}^{\mathbf{D}}$ [Hirsch, 1976]. For instance, let's consider the high dimensional spaces shown in Figure 2.9 which are represented in either $\mathbb{R}^2$ or $\mathbb{R}^3$. Since these spaces are parameterised by only one or two variables, they are intrinsically one or two dimensional manifolds embedded in two or three dimensions.

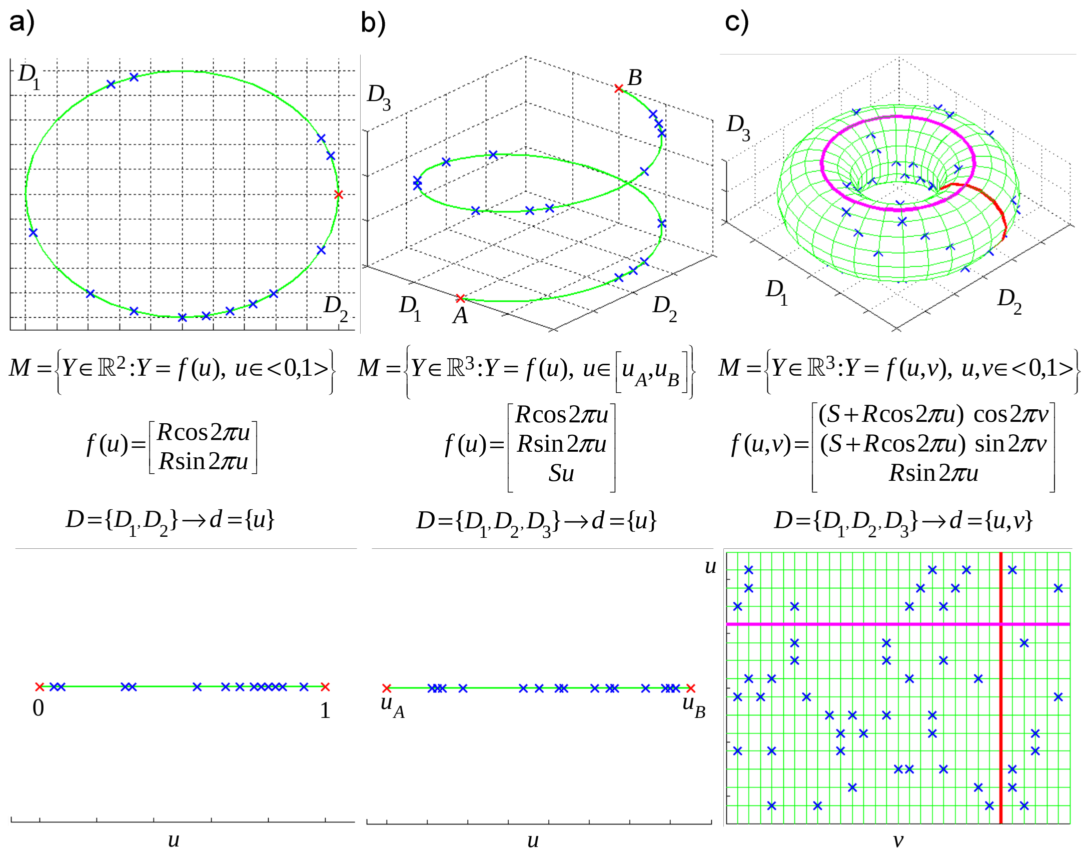

a)
b)
c)
D_1
D_2
D_3
A
B
$M=\{Y\in\mathbb{R}^2 : Y=f(u),\ u\in<0,1>\}$
$f(u)=\begin{bmatrix}R\cos 2\pi u\\ R\sin 2\pi u\end{bmatrix}$
$D=\{D_1,D_2\}\rightarrow d=\{u\}$
$M=\{Y\in\mathbb{R}^3 : Y=f(u),\ u\in[u_A,u_B]\}$
$f(u)=\begin{bmatrix}R\cos 2\pi u\\ R\sin 2\pi u\\ Su\end{bmatrix}$
$D=\{D_1,D_2,D_3\}\rightarrow d=\{u\}$
$M=\{Y\in\mathbb{R}^3 : Y=f(u,v),\ u,v\in<0,1>\}$
$f(u,v)=\begin{bmatrix}(S+R\cos 2\pi u)\cos 2\pi v\\ (S+R\cos 2\pi u)\sin 2\pi v\\ R\sin 2\pi u\end{bmatrix}$
$D=\{D_1,D_2,D_3\}\rightarrow d=\{u,v\}$
0
1
u_A
u_B
u
u
u
v

Figure 2.9. Examples of one (a,b) and two dimensional manifolds (c) embedded in either two (a) or three dimensions (b,c).

From a mathematical point of view, the concept of manifold is defined by recalling the following definitions from differential geometry and topology [Camastra and Vinciarelli, 2008]:

Definition 1. *A* homeomorphism *is a continuous function whose inverse is also a continuous function.*

Definition 2. *A d-dimensional* manifold *M is set that is locally homeomorphic with $\mathbb{R}^{\mathbf{d}}$. That is, for each $x \in M$, there is an open neighbourhood around x, N_x and a homeomorphism $f : N_x \to \mathbb{R}^{\mathbf{d}}$. These neighbourhoods are overlapping and referred to as* coordinate patches*, and the map is referred to a* coordinate chart. *The image of the coordinate charts is referred to as the* parameter space.

Definition 3. *A* smooth *(or differentiable) manifold is a manifold such that each coordinate chart is differentiable with a differentiable inverse (i.e., each coordinate chart is a diffeomorphism).*

In the context of feature extraction, a smooth manifold M is considered: it lies in a high dimensional space ($M \subset \mathbb{R}^{\mathbf{D}}$) and is homeomorphic with a low-dimensional space ($\mathbb{R}^{\mathbf{d}}$, with $\mathbf{d} << \mathbf{D}$).

2.2.2.2. Deterministic Frameworks

Deterministic dimensionality reduction methods optimise an objective function that does not contain any local optima, in other words the solution space is convex [Boyd and Vandenberghe, 2004]. The objective function has usually the form of a (generalised) Rayleigh-Ritz theorem [Horn and Johnson, 1985] (see example in section 4.4.1.2, equation (4.20)) and therefore is optimised by solving the (generalised) eigenvalue problem [Arnoldi, 1951, Fokkema et al., 1999, Knyazev, 2002]. The final embedded space is formed by eigenvectors which correspond to smallest or largest eigenvalues. Any deterministic method is classified as global or local one. The global methods perform the eigendecomposition of a dense cost matrix (Principal Component Analysis, Multidimensional Scaling, Kernel Principal Component Analysis, Isomap, Maximum Variance Unfolding), whereas local methods perform the eigendecomposition of a sparse cost matrix (Locally Linear

Embedding, Laplacian Eigenmaps). All these approaches are discussed in the subsequent sections.

2.2.2.2.1. Linear Methods

2.2.2.2.1.1. Principal Component Analysis

A typical and well-established representative of linear methods is Principal Component Analysis (PCA) which is also known as the Hotelling or the Karhunen-Loeve transform [Hotelling, 1933, Jolliffe, 1989, Jackson, 1991]. Its popularity is due to its conceptual simplicity, its analytical properties and the existence of efficient implementations which have polynomial complexity. The objective is to obtain a low-dimensional representation of the data that preserves maximum amount of variance. In fact, this defines an orthonormal coordinate system where the correlation between different axes is minimised.

In mathematical terms, PCA reduces dimensionality with an orthogonal linear transformation:

$$x = A^T y \tag{2.5}$$

which projects a number of (possibly) correlated variables into a smaller number of uncorrelated variables called *principal components*. It can be shown that this linear mapping A is formed by the top **d** eigenvectors of the **D***x***D** covariance matrix $C = \mathbf{N}^{-1} \sum_{i=1}^{\mathbf{N}} y_i y_i^T$ assuming that the input patterns y_i are centred on the origin. The first principal component accounts for the largest variability in the data, and each successive component accounts for the largest remaining variability. PCA is an optimal linear dimension reduction technique in the mean-square sense, i.e., it minimises the errors in reconstruction of the original data from its low-dimensional representation [Jolliffe, 1989, Jackson, 1991]. PCA is an example of a non-parametric feature extraction which produces a unique solution regardless of the distribution of the data, as long as the data have finite variance along the principal axes.

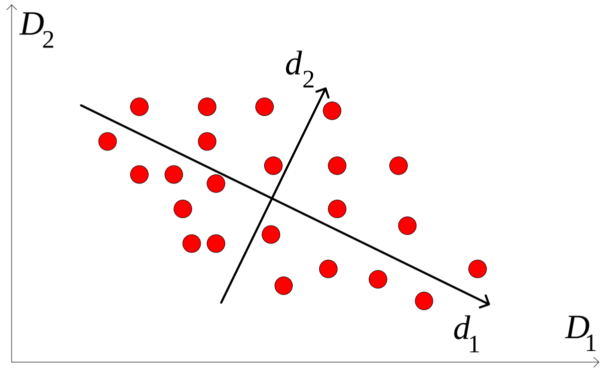


Figure 2.10. Geometrical interpretation of PCA. The PCA projects the data along the directions where the data vary the most.

2.2.2.2.1.2. Multidimensional Scaling

Another classical example of linear methods is Multidimensional Scaling [Torgerson, 1952, Kruskal, 1964, Cox and Cox, 1994] (MDS). In practice, MDS covers a collection of techniques sharing the common goal of faithfully preserving the inner products between different feature vectors in high and low dimensional spaces. This is achieved by, first, constructing the proximity matrix which measures the pairwise similarity among all patterns Y and, then, optimising a stress function. The stress function measures the error between the pairwise inner products in the low-dimensional and high-dimensional representations of the data:

$$\varepsilon = \sum_{i,j=1}^{\mathbf{N}} (y_i \bullet y_j - x_i \bullet x_j) \tag{2.6}$$

The minimisation of the above cost function depends on the specific properties of the chosen inner product. In most approaches MDS is motivated by the idea of preserving pairwise distances which are converted into equivalent dot products with a formula:

$$\tau(S) = -\frac{1}{2} CSC^T \tag{2.7}$$

where $C = (I - \frac{11^T}{\mathbf{N}})$ denotes the geometric centring matrix and $S \in \{X, Y\}$, 1 is a matrix of ones with the size $\mathbf{N} \times \mathbf{N}$. This leads to a technique called the metric MDS which exploits the raw stress function of any distance metric, such as the Euclidean and Manhattan distances [Cox and Cox, 1994]:

$$\varepsilon = \sum_{i,j=1}^{\mathbf{N}} (\tau(dist(y_i, y_j)) - \tau(dist(x_i, x_j))) \tag{2.8}$$

A solution is obtained from the spectral decomposition of the Gram matrix of inner products $g_{ij} = y_i \cdot y_j = -0.5C\ dist(y_i, y_j)C$ [Cox and Cox, 1994] and selecting the **d** dominant eigenvectors. The classical MDS is a special case of metric MDS where Euclidean distances are employed [Torgerson, 1952]. Though based on a somewhat different geometric intuition, classical MDS is closely related to PCA and yields identical output patterns. The connection between PCA and classical scaling is described in more detail in [Williams, 2002].

Alternatively, Sammon's cost function is used in the metric MDS to put more emphasis on retaining distances that were originally small [Sammon, 1969, Cox and Cox, 1994]. This is achieved by weighting the contribution of each pair (i, j) in the stress function using the inverse of their pairwise distance in the high dimensional space:

$$\varepsilon = \frac{1}{\sum_{i,j=1}^{\mathbf{N}} \tau(dist(y_i, y_j))} \sum_{i \neq j} \frac{(\tau(dist(y_i, y_j)) - \tau(dist(x_i, x_j)))^2}{\tau(dist(y_i, y_j))} \tag{2.9}$$

Finally, the non-metric MDS [Kruskal, 1964] is considered as a nonlinear approach since it discovers the underlying structure of monotonic data by maintaining the rank ordering of the interpoint distance based on the ranking of the value of dissimilarities derived from the original input space:

$$\varepsilon = \frac{\sum_{i=1}^{\mathbf{N}} \sum_{j=i+1}^{\mathbf{N}} (\tau(dist(x_i, x_j)) - \tau(\hat{dist}(x_i, x_j)))^2}{\sum_{i<j} \tau(\hat{dist}(y_i, y_j))} \tag{2.10}$$

where $\hat{dist}(x_i, x_j)$ are pseudo-distances derived from the $dist(x_i, x_j)$ with Kruskal's monotone regression procedure [Kruskal, 1964]. They are calculated in such a way that their rank order matches perfectly the rank order of the $dist(y_i, y_j)$ and they are as close as possible to the $dist(x_i, x_j)$ [Kruskal, 1964].

The minimisation of Sammon's metric MDS and non metric MDS is generally performed using either the conjugate gradient back propagation [Johansson et al., 1992] or a pseudo-Newton method [Battiti and Masulli, 1990, Cox and Cox, 1994].

2.2.2.2.1.3. Summary of Linear Methods

PCA and MDS are well-established linear methods used by the research community. However, the fact that they rely on the assumption that the data must lie approximately on a linear subspace of the high-dimensional data limits significantly the scope of potential real life applications.

2.2.2.2.2. Nonlinear Methods

2.2.2.2.2.1. Kernel-based Approaches

Kernel PCA (KPCA) is the nonlinear generalisation of traditional PCA in a high-dimensional space that is constructed using a kernel function [Schölkopf et al., 1997]. If the data is distributed in a nonlinear way then it should be projected on a curve rather than a line (Figure 2.11). Such distribution may be linearised using nonlinear mapping from the input space Y to a higher dimensional feature space, i.e. a Hilbert space H of possibly infinite dimension, using empirical kernel map $\Phi : \mathbb{R}^{\mathbf{D}} \to H, \Phi(y_i) \in H$ [Schölkopf and Smola, 2002]. Here the mapping Φ is approximated implicitly by the form of a dot product $\Phi(\cdot) \bullet \Phi(\cdot)$ in the feature space H . These inner products are computed using kernel functions without actually performing the mapping Φ [Schölkopf et al., 1997]. Formally, for an arbitrary pair of data points y_i and y_j , the dot product between them $\Phi(y_i) \bullet \Phi(y_j)$ is parameterised by the kernel function κ:

$$k_{ij} = \kappa(y_i, y_j) = \Phi(y_i) \bullet \Phi(y_j) \tag{2.11}$$

This kernel function can be any function which satisfies Mercer's condition [Mercer, 1909, Courant and Hilbert, 1953], i.e. gives rise to a positive semi definite Mercel kernel $K = \left\{k_{ij} \mid i, j = 1..\mathbf{N}\right\}$. Popular choices for the kernel function include:

- the linear kernel: $\kappa(y_i, y_j) = y_i \bullet y_j$ (makes KPCA equivalent to standard PCA),
- the polynomial kernel: $\kappa(y_i, y_j) = (b + y_i \bullet y_j)^a$,
- the Gaussian kernel:

$$\kappa(y_i, y_j) = \exp(-0.5 \frac{1}{\sigma^2} (y_i - y_j)^T (y_i - y_j)) \tag{2.12}$$

It is assumed that the data have a zero mean in the feature space H , thus in practice, the symmetric kernel matrix K is double centred by subtracting out the mean from each feature vector. Finally, KPCA computes the **d** dominant eigenvectors $\{v_j \mid j = 1..\mathbf{d}\}$ and eigenvalues $\{\lambda_j \mid j = 1..\mathbf{d}\}$ of the kernel matrix K to produce a low dimensional representation which is linearly related to the feature space, however nonlinearly related to the input space (Figure 2.11). In order to obtain the low-dimensional data representation, data are projected onto scaled versions of the eigenvectors v_j . The result of the projection is given by [Schölkopf et al., 1997]:

$$x = \{\sum_{i,k=1}^{\mathbf{N}} \frac{v_j}{\sqrt{\lambda_j}} \kappa(y_i, y_k) \mid j = 1..\mathbf{d}\} \tag{2.13}$$

Although KPCA is robust against local minima, a substantial disadvantage is that it is sensitive to the choice of kernel used: different kernels produce different low dimensional structures which display different performances. Since no a priori knowledge is available, a whole space of kernel functions has to be explored in order to find the most suited to a particular task. Moreover, KPCA is computationally very expensive, especially for large training set, since it requires evaluation of the kernel function in respect of all pairs of training points. To address this problem [Tipping, 2001] proposed to approximate the covariance matrix K in a feature space by a subset of outer products of feature vectors using a maximum likelihood criterion.

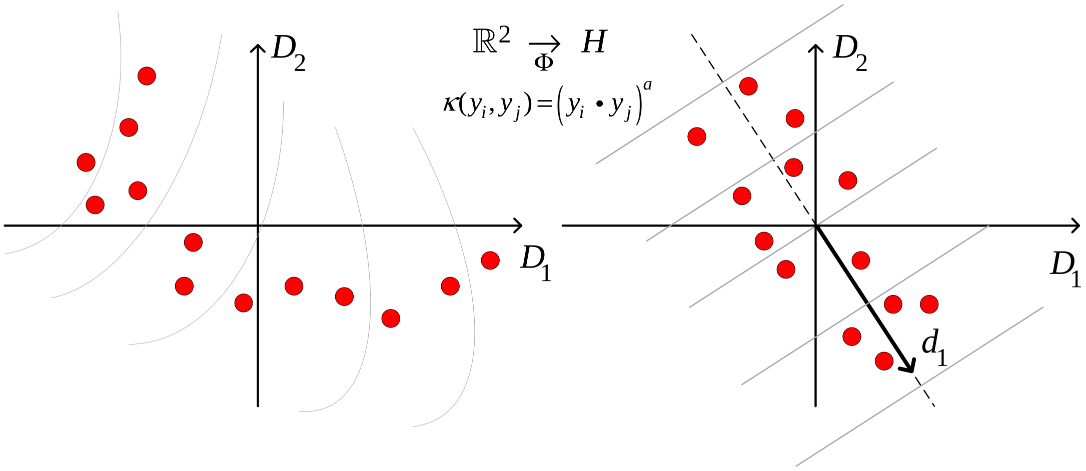

D_2
$\mathbb{R}^2 \xrightarrow[\Phi]{} H$
$\kappa(y_i, y_j) = (y_i \bullet y_j)^a$
D_2
D_1
D_1
d_1

Figure 2.11. The principle of kernel PCA: using a non-linear kernel function κ instead of the standard dot product, PCA is performed implicitly in a possible high-dimensional space H which is nonlinearly related to the input space.

2.2.2.2.2.2. Embedded-based Approaches

Embedded-based approaches, also called spectral methods, have emerged as a powerful tool for unsupervised nonlinear dimensionality reduction and manifold learning. They aim at preserving some geometrical property of the underlying manifold by constructing neighbourhood graphs which express nonlinear dependencies between high dimensional points. Spectral methods can broadly be divided into three families according to the way feature vectors are expressed in function of their neighbours:

- Locally Linear Embedding [Roweis and Saul, 2000] (LLE),

- Laplacian Eigenmaps [Belkin and Niyogi, 2002] (LE),
- Isometric Feature Mapping [Tenenbaum et al., 2000] (Isomap),

All these methods seek to produce an embedded space where proximity relations are preserved, so nearby points in data space remain close in low dimensional space. However, while Isomap attempts to maintain global geometric properties, LLE and LE focus on preserving local geometry in each neighbourhood, which implicitly tends to keep the global layout of the data manifold. These methods share the same structure of algorithm which is illustrated in Figure 2.12.

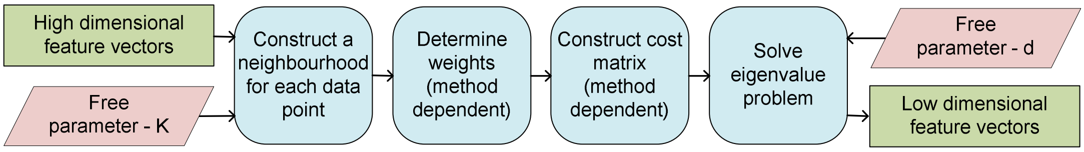

High dimensional feature vectors
Free parameter - K
Construct a neighbourhood for each data point
Determine weights (method dependent)
Construct cost matrix (method dependent)
Solve eigenvalue problem
Free parameter - d
Low dimensional feature vectors

Figure 2.12. Dimensionality reduction using spectral methods.

Briefly, the algorithm structure consists of the following steps. First, the neighbourhood for each data point is constructed and weights, which express the geometrical relationship between each data point and its neighbours, are determined according to the property to be preserved. Then, each method derives a cost matrix from this weighted graph and optimises it subject to constraints that make the problem well-posed. Finally, low dimensional embedding is obtained from the Eigen-decomposition of the cost matrix.

2.2.2.2.2.2.1. Neighbourhood Construction

All algorithms start by finding neighbours for each data point of the dataset. Since the neighbourhoods overlap with one another, a global topology of manifold is efficiently described by a combination of these neighbourhoods in a coherent structure. In addition, for local methods, it is assumed that within each neighbourhood the manifold is approximately linear. This is justified by Taylor's theorem that any differentiable function becomes approximately linear in a sufficiently small region around a point [Kline, 1998].

The neighbourhood for each point is formed by selecting either the K-nearest neighbours or neighbours whose distances are lower than a constant threshold ε, i.e. points belonging to a hyper sphere of radius ε. The Euclidean norm was used as distance metric in the original papers [Tenenbaum et al., 2000, Roweis and Saul, 2000, Belkin and Niyogi, 2002]. Because of its conceptual simplicity and its robustness against data density, the K-nearest neighbour approach is far more popular in the research community. Several methods to automate the process of the parameter selection have been proposed [Kouropteva et al., 2002, Samko et al., 2006, Karbauskait et al., 2007, Goldberg and Ritov, 2009]. Detailed discussion about selection of the neighbourhood size can be found in section 3.2.1. For a large dataset, the identification of neighbours is realised very efficiently with the usage of kd-tree [Bentley, 1975].

2.2.2.2.2.2.2. Determination of Weights

Weights W express the magnitude of geometrical relationship between each data point and its neighbours.

In LLE, they summarise the neighbours' contribution to the linear reconstruction of a data point and are obtained by solving a least square error problem in the original space [Roweis and Saul, 2000] By design, these reconstruction weights reflect intrinsic geometric properties of the data and are invariant to translations, rotations and scaling. Therefore, they are expected to be equally valid for characterisation of the local geometry in the low dimensional patches of a manifold.

In the case of the LE and Isomap algorithms, the weights are related to the distance between a high dimensional point and its neighbours using respectively heat kernel [Belkin and Niyogi, 2002] and Euclidean distance [Tenenbaum et al., 2000].

The manifold is then approximated, by an adjacency undirected graph. Nodes in these graphs correspond to the data points and edges represent the weights between points, i.e. neighbour relations. Graphs are only locally connected (through the neighbourhood of each point), because an edge connection exists only if its weight is not equal to 0. However, one of key assumptions behind the neighbourhood size selection procedure is to create overlapping neighbourhoods on the manifold, thus a fully connected graph can be assembled (Figure 2.13c,d).

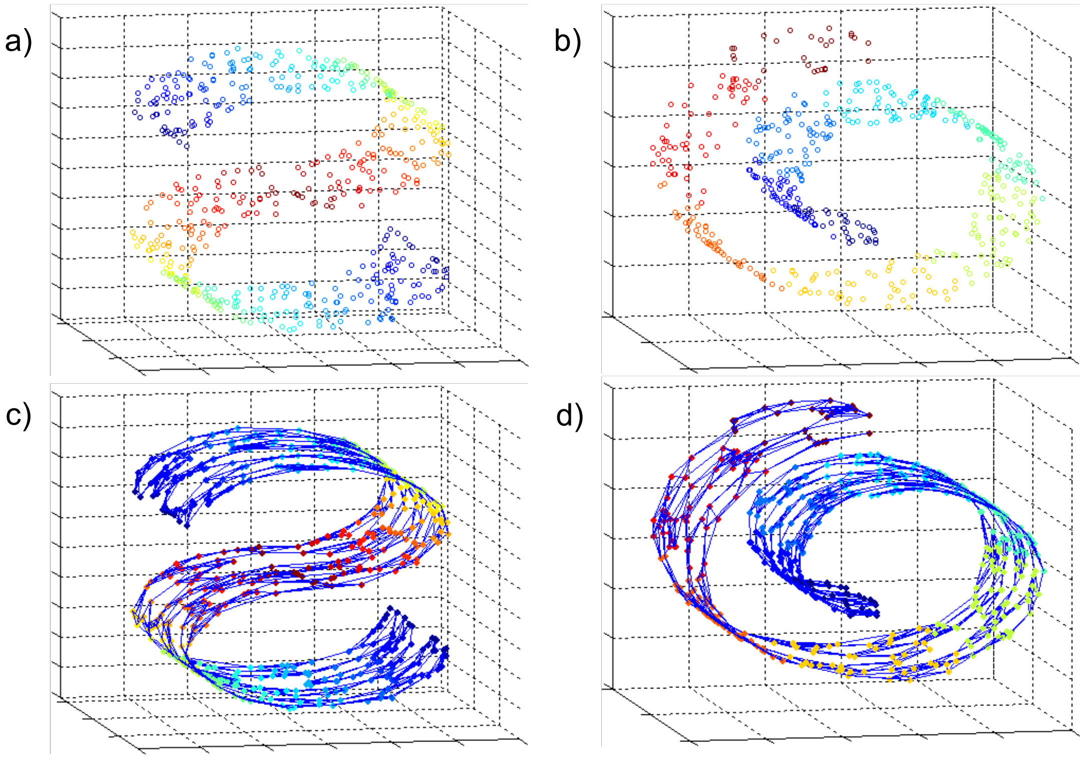


Figure 2.13. Examples of fully connected neighbourhood graphs for: a,c) s-curve and b,d) swissroll.

In the case of Isomap which aims at preserving the global topology of the manifold, the geodetic distances $dist_G(y_i, y_j)$ between all pairs of data points on the manifold are estimated. The geodetic distance between two points is defined as the minimum length of all possible paths encapsulated within the manifold joining both points [Do Carmo, 1976] (Figure 2.14). Locally, the geodesic distance is equal to the Euclidean distance between two points in the high dimensional space. However this approximation breaks down for distant points. Therefore, globally, the geodesic distances are estimated by computing shortest path distances in the graph using a technique such as the Floyd-Warshall's algorithm [Floyd, 1962, Warshall, 1962] or the computationally more efficient Dijkstra's algorithm [Dijkstra, 1959].

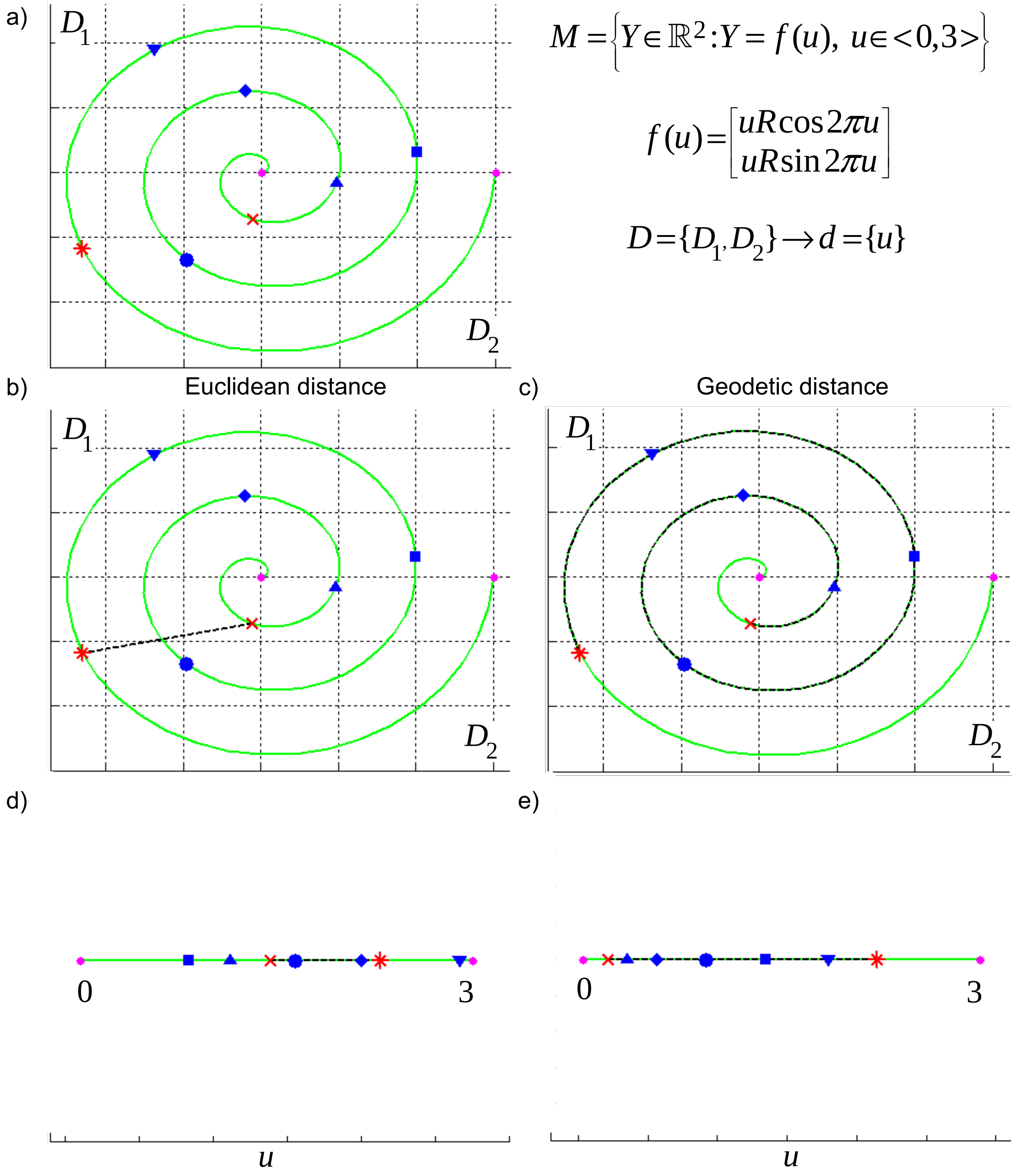

a)
D_1
D_2
$M = \{ Y \in \mathbb{R}^2 : Y = f(u),\ u \in <0,3> \}$
$f(u) = \begin{bmatrix} uR\cos 2\pi u \\ uR\sin 2\pi u \end{bmatrix}$
$D = \{D_1, D_2\} \rightarrow d = \{u\}$
b)
Euclidean distance
D_1
D_2
c)
Geodetic distance
D_1
D_2
d)
0
3
u
e)
0
3
u

Figure 2.14. One dimensional manifold embedded in a two dimensional space illustrating a difference between Euclidean and geodesic distance. Let's consider two red points in the spiral (a), the preservation of Euclidean distance (b) does not reflect the intrinsic dissimilarity between these two points in the one dimensional manifold (d). In contrast, the geodetic distance (c) is able to encapsulate adequately the relationship between these two points along the one-dimensional manifold (e).

2.2.2.2.2.2.3. Cost Function and Cost Matrix

In the next step, an appropriate cost function and corresponding cost matrix are constructed. Since the calculated weights reflect the intrinsic geometric structure of the manifold, an embedded manifold in a low dimensional space is constructed using the same weights. This is achieved by optimizing different quadratic cost functions with respect to the unknown coordinates X and the fixed cost matrix W (the original inputs Y are not involved).

In the LLE method, each low dimensional data point is reconstructed entirely from a weighted linear combination of its respective nearest neighbours, so the d-dimensional coordinates X are chosen to minimise the embedding cost function :

$$\varepsilon = \sum_{i=1}^{\mathbf{N}} \left\| x_i - \sum_{j=1}^{\mathbf{N}} w_{ij} x_j \right\|^2 = tr\left(X^T M X \right) \tag{2.14}$$

where $M = \left(I - W\right)^T \left(I - W\right)$ is the cost matrix.

Similarly, LE minimises the relative distance between nodes in a graph in order to preserve proximity relations between points. As a result, the following cost function was designated using similar components:

$$\varepsilon = \frac{1}{2} \sum_{i,j=1}^{\mathbf{N}} \left\| x_i - x_j \right\|^2 w_{ij} = tr\left(X^T L X \right) \tag{2.15}$$

Here, the cost matrix L is called the Laplacian matrix and is defined by: $L = M - W$, where $M = diag\{m_{11}, m_{22}, ..., m_{\mathbf{nn}}\}$ is a diagonal matrix with entries $m_{ii} = \sum_{j=1}^{\mathbf{N}} w_{ij}$.

The optimisation of the above objective functions is performed subject to the following constraints [Roweis and Saul, 2000, Belkin and Niyogi, 2002]:

- a square cost matrix is real, symmetric and positive semidefinite (i.e. Hermitian).

- the outputs X are centred on the origin,
- embedding vectors have unit covariance.

Isomap tries to preserve the distances and angles between nodes in the graph; however its cost function has a different formulation. Isomap can be understood as a non-linear extension of the classical metric MDS (section 2.2.2.2.1.2), in which, estimates of geodesic distances along the sub manifold are preserved instead of standard Euclidean distances. In other words, Isomap tries to discover points whose pairwise Euclidean distances $dist_E(x_i, x_j)$ in the embedded space match geodesic distances $dist_G(y_i, y_j)$ in the high dimensional data space. As a result, derived from the equation (2.8), the following objective function was proposed:

$$\varepsilon = \sum_{i,j} (\tau(dist_G(y_i, y_j)) - \tau(dist_E(x_i, x_j))) \tag{2.16}$$

2.2.2.2.2.2.4. Optimisation

In the last step, the actual low dimensional representation of data points is revealed through optimisation of an objective function ($\operatorname{argmin}_X \varepsilon$). The optimisation of this constrained quadratic programming problem is performed by introducing Lagrange multipliers [Mizrahi and Sullivan, 1990] to enforce the constraints to an objective function.

The embedded space X is spanned by the eigenvectors given by either the d smallest nonzero eigenvalues in the case of LLE and LE or the d largest eigenvalues for Isomap. Eigenvectors and eigenvalues are calculated by spectral decomposition of cost matrices [Arnoldi, 1951, Fokkema et al., 1999, Knyazev, 2002] according to the generalisation of the Rayleigh-Ritz theorem [Horn and Johnson, 1985]:

- LLE: eigenvalue problem is solved on the sparse cost matrix M .
- LE: generalised eigenvalue problem is solved on the sparse cost matrix L .

- Isomap: eigenvalue problem is solved on the dense cost matrix $\tau(dist_G(y_i, y_j))$ where τ is defined according to equation (2.7).

2.2.2.2.2.2.5. Extensions

Since research into embedding based approaches has been very active, many extensions and improvements have been suggested. Some of them are summarised in this section.

Instead of preservation of the specific local geometric relationships (LLE, LE) or isometric structure of data (i.e. distances and angles) (Isomap), in some scenarios a more faithful embedding of high dimensional data can be obtained by maximally preserving only angles in each neighbourhood [De Silva and Tenenbaum, 2003, Sha and Saul, 2005]. Based on this concept, conformal Isomap [De Silva and Tenenbaum, 2003] and conformal eigenmaps [Sha and Saul, 2005] (extension of LLE and LE) were proposed which attempt to maintain explicitly these local angles. Note that the class of conformal embeddings includes all isometric embeddings, but not vice versa.

Alternatively, Hessian LLE [Donoho and Grimes, 2003], Hessian Eigenmaps [Donoho and Grimes, 2003] and Local Tangent Space Alignment [Zhang and Zha, 2005] (LTSA) explore the geometric relations between neighbouring data points in a local tangent space which is constructed at every point. Hessian LLE minimises the ‘curviness’ of the high-dimensional manifold under the constraint that locally the low-dimensional data representation is isometric. The local tangent space at every data point is described by the Hessian, thus, the global curviness of the manifold is measured by means of these local Hessians. Hessian Eigenmaps are based on a similar concept: they simply replace the Laplacian manifold by the Hessian manifold. In contrast, LTSA aligns all local tangent subspaces to construct a global coordinate system for a nonlinear manifold.

The local tangent space is constructed by applying PCA in each neighbourhood, so a linear mapping is defined from a high-dimensional data point to its local tangent space. It is assumed that a similar mapping can be computed from the corresponding low-dimensional data point to the same local tangent space.

Since the process of dimensionality reduction using global methods is very computationally demanding in comparison to local methods [van der Maaten et al., 2009], [De Silva and Tenenbaum, 2003] proposed an acceleration procedure which initially reduces dimensionality of a small subset of “landmark” feature vectors. In turn, the rest of the embedding is approximated from these landmarks using the Nyström approximation [De Silva and Tenenbaum, 2003].

By definition, all presented methods are unsupervised algorithms; therefore they do not take into account the availibility of data labels when producing the embedded space. To address supervised learning problems (e.g. classification) a few extensions were proposed. They include discriminant Isomap [Yang, 2003], supervised LLE [De Ridder et al., 2003] and semi-supervised LE [Zheng et al., 2008].

2.2.2.2.2.2.6. Summary of Embedded-based Approaches

All methods discussed here are based on the assumption that the observed data are densely sampled, also called smoothly sampled, on the D-dimensional manifold in the data space and the underlying embedded manifold exists. In such case, local linearity assumption is valid for LE and LLE, thus, each patch can be characterised accurately by linear coefficients which encapsulate geometrical relationships between points. In the case of Isomap, it is assumed that the geodesic distance between nearby points is approximately linear. Thus, the geodesic distance between two near points is well approximated by the Euclidean distance in the high-dimensional data space.

These methods do not provide any explicit generic function for mapping between low and high dimensional spaces nor probabilistic density model. As a result, embedding for new unseen points cannot be obtained directly. Despite this limitation, these methods have proved very popular because they can handle efficiently very large high dimensional datasets (especially local methods) and scale well with dimensionality $\mathbf{d}$. Moreover, the analytical non iterative optimisation process guarantees a unique global solution.

2.2.2.2.2.3. Maximum Variance Unfolding

Maximum Variance Unfolding [Weinberger and Saul, 2005] (MVU) is a global nonlinear dimensionality reduction method inspired by KPCA and embedding based approaches. Since the choice of the kernel plays a crucial role in KPCA, MVU attempts to learn kernel matrix K from neighbourhood graph restrictions so that the kernel function does not need to be chosen manually.

The algorithm structure is similar to embedding based approaches (Figure 2.12). First, the neighbourhood for each data point is constructed as described in section 2.2.2.2.2.2.1 and the fully connected adjacency graph is assembled. In contrast to spectral methods, MVU employs a very simple rule for edge weights: a value of 1 is assigned to each pair of neighbours [Weinberger and Saul, 2005]. From such discretised approximation of the manifold, the kernel based matrix is derived.

MVU aims at preserving exact distances and angles between nodes in the graph. This is achieved by maximisation of the total variance which pulls embedding coordinates as far apart as possible:

$$\varepsilon = \frac{1}{2\mathbf{N}} \sum_{i,j} \left\| x_i - x_j \right\|^2 = \sum_i \left\| x_i \right\|^2 \qquad (2.17)$$

with local isometry constraints to maintain pairwise distances and implicitly angles:

$$\left(k_{ii} - 2k_{ij} + k_{jj}\right) w_{ij} = \left\| y_i - y_j \right\|^2 w_{ij} \tag{2.18}$$

where K is a cost Gram matrix where the inner products $k_{ij} = x_i \bullet x_j$. As a consequence of this formulation, a quadratic programming problem (2.17) is simplified to a linear programming problem:

$$\varepsilon = \sum_{i=1}^{\mathbf{N}} \left\| x_i \right\|^2 = \sum_{i=1}^{\mathbf{N}} k_{ii} = tr(K) \tag{2.19}$$

A low dimensional embedding is discovered by optimisation of the above constrained linear programming problem using semi definite program [Vandenberghe and Boyd, 1996] followed by the eigendecomposition of the obtained cost matrix K. The embedded space X is spanned by the eigenvectors given by the $\mathbf{d}$ largest eigenvalues.

Similarly to Isomap, MVU is a global method with a dense cost matrix, hence the optimisation process is computationally demanding. Inspired by landmark Isomap [De Silva and Tenenbaum, 2003], [Kilian et al., 2005] proposes a conceptually similar acceleration procedure, where a small subset of “landmark” feature vectors is used for dimensionality reduction. In turn, the rest of the embedding is approximated from these landmarks using a factorised approximation of the Gram matrix [Kilian et al., 2005]. Since it is easier to optimise a linear programming problem than a quadratic one, [Hou et al., 2009] presents a linear reformulation of the LLE and LE cost functions, whereas [Wang and Li, 2009] combines MVU and LE to design the distinguishing variance embedding method which maximises the global variance subject to a proximity preservation constraint derived from LE.

2.2.2.3. Probabilistic Frameworks

The main limitations of the previously described methods are the absence of an associated probability density and the lack of a generative model, which are essential in many applications. As a result, another class of dimensionality reduction methods evolved, the so-called latent variable models [Bishop, 1999] (LVMs). LVMs are statistical methods for modelling the covariance structure of high dimensional data using a small number of variables. Comprehensive overviews of probabilistic frameworks can be found in [Bishop, 1999, Carreira-Perpinán, 2001, Quirion et al., 2008].

Let's consider an unknown distribution $p(y)$ in an observed data space $\Upsilon \subseteq \mathbb{R}^{\mathbf{D}}$ ($y \in \Upsilon$) of which only samples $Y = \{y_i \mid i = 1..\mathbf{N}\}$ are known. The observed high-dimensional samples are assumed to be independent and identically distributed random variables, which are generated from an underlying low-dimensional process relying only on d degrees of freedom (Figure 2.15). Since this process is defined by a set of latent or hidden variables x_i, the entire space of these hidden variables is referred as the latent space $\chi \subseteq \mathbb{R}^{\mathbf{d}}$ ($x_i \in \chi, i = 1..\mathbf{N}$).

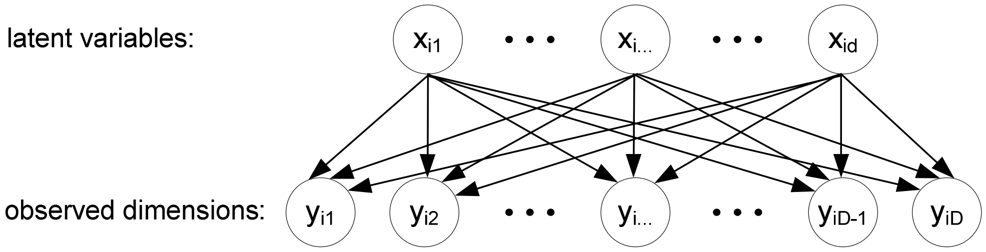


Figure 2.15. Latent variable model with D observed dimensions and d latent variables. The latent variables may or may not be independent.

A point x in the latent space is generated according to a prior distribution $p(x)$ and is related to a higher dimensional observed space through a continuous and fixed transformation $f : \chi \rightarrow \Upsilon$. Since $M = f(\chi)$ represents a low dimensional manifold, where the data would reside if there was no noise, the observed distribution of whole data space is approximated by adding the noise model $p(y \mid x) = p(y \mid f(x))$. Figure 2.16 illustrates the concept of latent variable models.

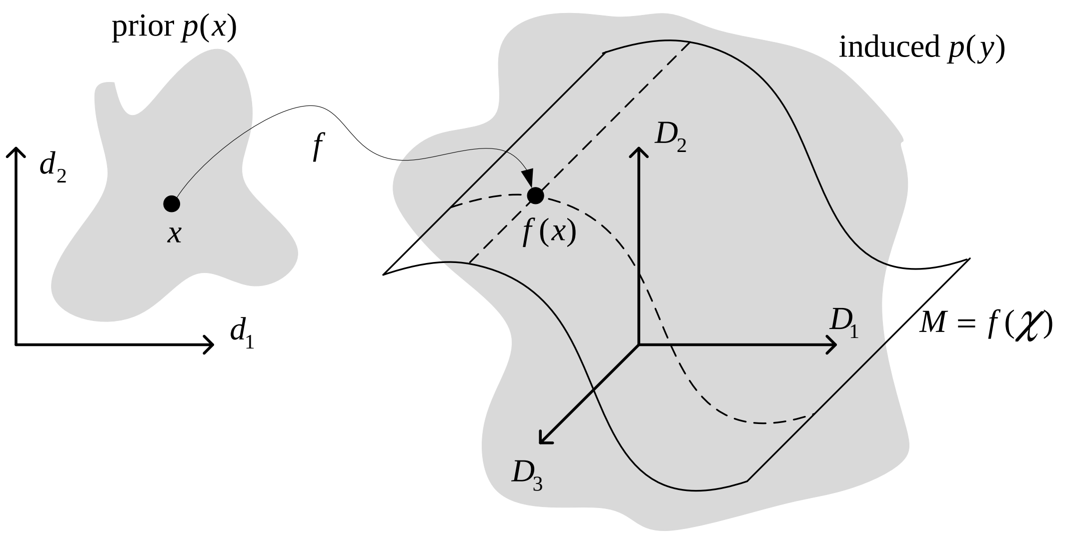


Figure 2.16. Illustration of a continuous latent variable model with a 2-dimensional latent space χ on the left and a 3-dimensional observed space Υ on the right.

The marginal distribution in data space is given by the joint probability density function $p(y,x)$ in the product space $\Upsilon \times \chi$ by integrating over the latent space:

$$p(y) = \int_{\chi} p(y,x)dx = \int_{\chi} p(y \mid x) p(x)dx \qquad (2.20)$$

This is called the fundamental equation of latent variable models [Bartholomew, 1984]. According to the above equation, any continuous latent variable model is defined by three main components:

- a prior distribution in the latent space $p(x)$,
- a smooth non-singular mapping from latent to observed space $f(x)$,
- a noise model in the data space $p(y \mid x)$.

Therefore, the objective is to find a combination of latent distribution $p(x)$ along with a uncertainty model $p(y \mid x)$ that approximates 'satisfactorily', given observed data, samples using the axiom of local independence ($i = 1..\mathbf{N}$):

$$p(y_i \mid x_i) = \prod_{j=1}^{\mathbf{D}} p_j(y_{ij} \mid x_i) \qquad (2.21)$$

This axiom states that, for some $\mathbf{d} \leq \mathbf{D}$, *the observed variables are conditionally independent given the latent variables* [Bartholomew, 1984, Everitt, 1984]. Hence, the goal is to identify the best latent variables for which this axiom holds. It is assumed that the density function $p(y \mid x)$ used for the noise model has the following properties [Carreira-Perpinán, 2001]:

- It is centred at $f(x)$, which would become a single point in the absence of noise.
- It decays gradually as the distance to $f(x)$ increases.
- It assigns nonzero density to every point in the observed space.
- It should have a diagonal covariance matrix to account for different scales in the different observed dimensions $y_j\,(j = 1..\mathbf{D})$.

The prior $p(x)$, the mapping function f and the noise model $p(y \mid x)$ usually rely on a set of parameters denoted collectively by Φ. These parameters relate the sets of latent and observed variables. Parameters Φ are usually obtained by iterative maximum likelihood estimation using for instance the Monte Carlo simulation [Fishman, 1996, Rubinstein and Kroese, 2008] or more often Expectation-Maximisation (EM) algorithm [Dempster et al., 1977, McLachlan and Krishnan, 2008]. Note that all LVMs rely on an optimisation process, thus they are sensitive to local optima.

Since the observed data points are assumed to be independent, the likelihood of the full data set is:

$$p(Y) = \prod_{i=1}^{\mathbf{N}} p(y_i) \tag{2.22}$$

where $p(y_i \mid \Phi)$ is given by equation (2.20).

Latent variable models are classified as linear and nonlinear according to the corresponding functional form used for mapping. In the following sections, the notation $\mathcal{N}(z \mid \mu, \Sigma)$ will denote a Gaussian distribution over z with mean μ and covariance Σ.

2.2.2.3.1. Linear Methods

Two well established linear models are factor analysis [Everitt, 1984, Bartholomew, 1987] (FA) and probabilistic principal components analysis [Tipping and Bishop, 1999b] (PPCA). The relationship between the latent variable and the observed data point is linear with added noise and expressed by the following generative model:

$$f(x_i) = A\,x_i + \mu \tag{2.23}$$

$$y_i = f(x_i; A) + \varepsilon_i \tag{2.24}$$

where the matrix A ($\mathbf{D} \times \mathbf{d}$) expresses the linear relationship between the latent-space χ and the data space Υ, while the parameter vector μ permits the model to have a non-zero mean. ε denotes the Gaussian noise:

$$p(\varepsilon) = \mathcal{N}(\varepsilon \mid 0, \psi) \tag{2.25}$$

while the corresponding conditional noise model is centred at $f(x)$ with the diagonal covariance matrix ψ:

$$p(y_i \mid x_i, A, \psi) = \mathcal{N}(y_i \mid f(x_i; A), \psi) = \mathcal{N}(y_i \mid Ax_i + \mu, \psi) \tag{2.26}$$

The latent variables are defined to be independent and Gaussian with a unit variance, $p(x_i) = \mathcal{N}(x_i \mid 0, I)$.

According to equation (2.20) and the defined distributions, it can be shown analytically that the marginal distribution in the data space is normal [Tipping and Bishop, 1999a, Tipping and Bishop, 1999b, Carreira-Perpinán, 2001] given the model parameters $\Phi = \{A, \psi\}$:

$$\begin{aligned} p(y_i \mid A, \psi) &= \int_{\chi} p(y_i \mid x_i, A, \psi)\, p(x_i)\, dx_i = \mathcal{N}(y_i \mid \mu, \Sigma) \\ \Sigma &= AA^T + \psi \end{aligned} \tag{2.27}$$

where the sample mean is $\mu = \mathbf{N}^{-1} \sum Y$.

The goal of LVM is to estimate the parameters Φ that best model the covariance structure of Y. A standard approach for fitting LVMs is to marginalise the latent variables by optimising the parameters via maximisation of the observed data likelihood given the parameters $p(Y \mid \Phi)$. The log-likelihood of a normal distribution (2.22) based on (2.27) for the sample Y is:

$$L(\Phi) = \ln p(Y \mid \Phi) = \sum_{i=1}^{\mathbf{N}} \ln p(y_i \mid \Phi) = -\frac{1}{2}(\mathbf{ND} \ln 2\pi + \mathbf{N} \ln |\Sigma| + tr(\Sigma^{-1} C)) \quad (2.28)$$

where C is a covariance matrix of the observations $C = (Y - \mu)(Y - \mu)^T$. Estimates for parameters Φ are obtained via maximisation of the log posterior $L(\Phi)$ using a variation of the EM algorithm [Rubin and Thayer, 1982, Tipping and Bishop, 1999b].

Applying the Bayes rule to equation (2.26), the posterior distribution of the latent variables x_i conditioned on the observation y_i with constant covariance is estimated by [Tipping and Bishop, 1999a, Carreira-Perpinán, 2001]:

$$p(x_i \mid y_i, \Phi) = \mathcal{N}(A^T \Sigma^{-1}(y_i - \mu), (I + A^T \psi^{-1} A)^{-1}) \quad (2.29)$$

The dimensionality reduction process F is performed by projecting the observed data into a representation of the reduced dimensionality according to the posterior mean vectors in (2.29):

$$x_i = A^T \Sigma^{-1}(y_i - \mu) \quad (2.30)$$

where the corresponding optimal least-squares linear reconstruction f of the observed data from the posterior mean vectors is expressed by:

$$y_i = A(A^T A)^{-1} \Sigma \, x_i + \mu \quad (2.31)$$

The main difference between FA and PPCA is in the assumed noise model. PPCA can be seen as a maximum likelihood FA in which the isotropic noise model is adopted, i.e. residual variances ψ_i of covariance matrix ψ are constrained to be equal, i.e. $\psi = I\sigma^2$. FA thus models the individual noise variability in each of the dimensions, whereas PPCA assumes all dimensions have an equal noise level. As a result, while FA parameters are estimated iteratively as any other LVM method (section 2.2.2.3), the parameters of PPCA can be computed explicitly by numerical singular value decomposition of the covariance matrix $C = U_D V_D S^T$, where $V_D = diag\{\boldsymbol{\lambda}_{\mathbf{1}},...,\boldsymbol{\lambda}_{\mathbf{D}}\}$ denotes the diagonal matrix of eigenvalues (ordered decreasingly), $U_D = \{u_j \mid j = 1..\mathbf{D}\}$ are the associated eigenvectors and S is an arbitrary rotation matrix. In [Tipping and Bishop, 1999b] it has been shown that, with $\Sigma = AA^T + I\sigma^2$, the log-likelihood (2.28) is maximised when:

$$A = U_d (V_d - I\sigma^2)^{1/2} \tag{2.32}$$

and the corresponding maximum-likelihood estimator for σ^2 is given by:

$$\sigma^2 = \frac{1}{\mathbf{D}-\mathbf{d}} \sum_{j=\mathbf{d}+1}^{\mathbf{D}} \lambda_j \tag{2.33}$$

The main advantage of both approaches is the existence of a bidirectional projection function between low and high dimensional spaces; however their effectiveness is limited because of their global linearity assumption.

2.2.2.3.2. Nonlinear Methods

2.2.2.3.2.1. Mixture of Local Linear Models

Mixture approaches capture nonlinear complexity of high dimensional space by a combination of local linear models [Everitt and Hand, 1981]. The objective of a finite mixture of LVMs is to perform concurrently clustering and dimensionality reduction so that a complicated global structure of high dimensional data can be characterised by a collection of simple models in different regions of the observed space. Mixture of local linear models (MLLM) can be composed of FA [Ghahramani and Hinton, 1997] (MFA) or PPCA [Tipping and Bishop, 1999a] (MPPCA).

The probability of observing sample y under a mixture model with **K** components is given by:

$$p(y)=\sum_{k=1}^{\mathbf{K}}\pi_k p(y\,|\,k) \tag{2.34}$$

where the π_k express mixing proportions ($\pi_k \geq 0, \sum \pi_k = 1$). $p(y\,|\,k)$ is a local LVM on latent space χ_k defined according to (2.20) for each model:

$$p(y\,|\,k)=\int_{\chi_k} p(y,x\,|\,k)dx=\int_{\chi_k} p(y\,|\,x,k)p(x\,|\,k)dx \tag{2.35}$$

where $p(x\,|\,k)$ is the prior distribution in the latent space of the k*th* component, $p(y\,|\,x,k)$ is its noise model, and $f_k = \chi_k \rightarrow \Upsilon$ is its mapping from latent to data space. Note that a separate mean vector μ_k is now associated with each local model along with a different set of parameters $\Phi_k = \{A_k, \psi_k, \pi_k\}$. The log-likelihood of observing a whole data Y according to equations (2.22) and (2.34) is:

$$L(\Phi)=\ln\prod_{i=1}^{\mathbf{N}} p(y_i\,|\,\Phi)=\ln\prod_{i=1}^{\mathbf{N}}\sum_{k=1}^{\mathbf{K}}\pi_k p(y_i\,|\,\Phi_k,k)=\sum_{i=1}^{\mathbf{N}}\ln\sum_{k=1}^{\mathbf{K}}\pi_k p(y_i\,|\,\Phi_k,k) \tag{2.36}$$

The maximum of the above log-likelihood with respect to the parameters $\Phi = \{\Phi_k \,|\, k = 1..\mathbf{K}\}$ can be found by adaptation of the iterative EM algorithm [Ghahramani and Hinton, 1997, Tipping and Bishop, 1999a].

Since a collection of local models does not provide a global parameterisation of the manifold, [Roweis et al., 2002] addressed this problem by proposing an extension of MFA called Global Coordination of Local Linear Models (GCM). This method encourages the global consistency along the manifold of disparate internal representations by incorporating an additional variational penalty term into the maximum likelihood objective function. Alternatively, [Teh and Roweis, 2003] present an automatic alignment procedure which is invoked after learning the local dimensionality reduction experts (LLC). Thank to the separation of the learning and coordination processes, the algorithm gains efficiency and avoids local optima in the coordination phase. Given an already trained mixture, the alignment is achieved by solving a variant of the LLE eigenvalue problem on the internal representations of the mixture components. This framework can be applied to any set of experts, especially MFA and MPPCA.

Modelling nonlinearity of high dimensional data by a combination of simple local reducers is an attractive and effective alternative to linear LVMs; however the main drawback is that a methodology to select automatically the number of mixture components has not yet been solved satisfactory by the research community.

2.2.2.3.2.2. Nonlinear Function Mapping

2.2.2.3.2.2.1. Generative Topographic Mapping

Generative Topographic Mapping [Bishop et al., 1998] (GTM) is a nonlinear LVM which has been proposed as a principled alternative to Self-Organizing Map [Kohonen, 1982]. The basic concept behind the algorithm is to define a discrete prior distribution $p(x)$ given by a sum of delta functions centred on the nodes x_k of a uniform grid in latent space:

$$p(x) = \frac{1}{\mathbf{K}} \sum_{k=1}^{\mathbf{K}} \delta(x - x_k) \tag{2.37}$$

This discrete prior can be seen as a fixed approximation of a continuous and uniform distribution using Monte Carlo sampling [MacKay, 1995]. It assigns nonzero probability only to the points $\{x_k \mid k = 1..\mathbf{K}\} \subset \mathbb{R}^{\mathbf{d}}$. The distribution of the noise model $p(y \mid x)$ is chosen to be an isotropic Gaussian centred on $f(x)$ and having a variance γ. The mapping f is performed using a radial basis function network (a special case of a generalised linear model):

$$y = f(x) = A\ \phi(x) \tag{2.38}$$

where ϕ is a vector of fixed basis functions. Each latent point x_k, after projection to a corresponding point in a data space y_k, forms the centre of a Gaussian density function, as illustrated in Figure 2.17. The set of model parameters includes a coefficient matrix A and a variance $\gamma : \Phi = \{A, \gamma\}$. By substituting (2.37) and (2.26) into (2.22), the distribution function in the whole data space takes the form:

$$p(Y \mid \Phi) = \prod_{i=1}^{\mathbf{N}} \frac{1}{\mathbf{K}} \sum_{k=1}^{\mathbf{K}} p(y_i \mid x_k, \Phi) \tag{2.39}$$

The model parameters Φ are determined by the maximum log-likelihood of the above posterior using the EM algorithm [Bishop et al., 1998]:

$$L(\Phi) = \ln p(Y \mid \Phi) = \sum_{i=1}^{\mathbf{N}} \ln \frac{1}{\mathbf{K}} \sum_{k=1}^{\mathbf{K}} p(y_i \mid x_k, \Phi) \tag{2.40}$$

Since the mapping function f is smooth and continuous, the projected points have a topographic ordering in the sense that any two points which are close in the latent space will be mapped to the close points in data space. The initial low dimensional representation is initialised using PCA.

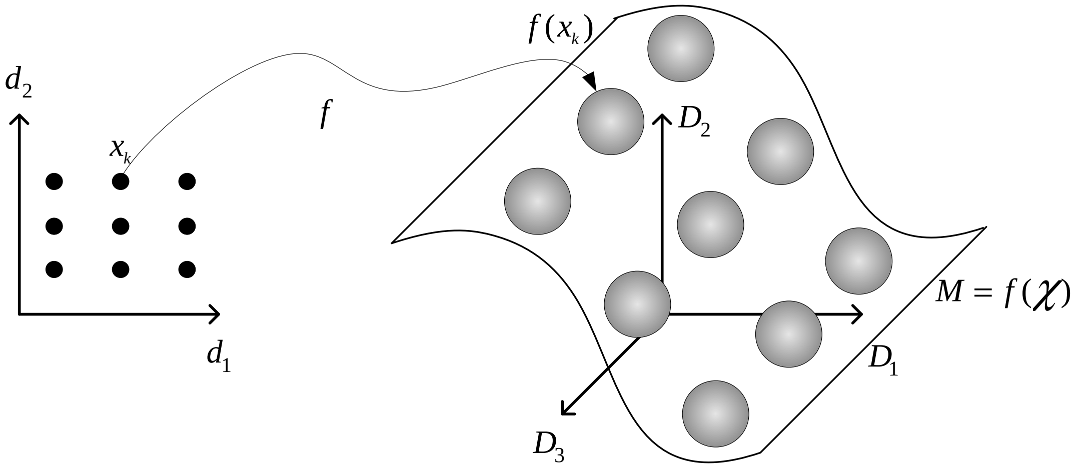


Figure 2.17. Discrete prior distribution $p(x)$ on the left consists of delta functions, located at the nodes of a regular grid in latent space. Each node x_k is mapped to a corresponding point $f(x_k)$ in observed space, where it forms the centre of a Gaussian distribution.

The major limitation of GTM is that, since it relies on Monte Carlo sampling, it requires a uniform discretised gridding of the latent space . As a result, both the numbers of latent grid points $\mathbf{K}$ and basis functions ϕ grow exponentially with the dimension of the latent space [Bishop, 1995, Carreira-Perpinán, 2001]. Another shortcoming is that EM estimates may converge to bad suboptimal maxima [Bishop, 1995, Carreira-Perpinán, 2001].

2.2.2.3.2.2.2. Gaussian Process Latent Variable Model

Gaussian Process Latent Variable Model was derived from the observation that a particular interpretation of probabilistic PCA is a product of Gaussian Process (GP) models [Lawrence, 2004, Lawrence, 2005].

2.2.2.3.2.2.2.1. Dual Probabilistic PCA

In standard approaches, such as PPCA [Tipping and Bishop, 1999b], GTM [Bishop et al., 1998] and MLLMs [Ghahramani and Hinton, 1997, Tipping and Bishop, 1999a], LVMs are learned by marginalising the latent variables X and optimising the parameters Φ via maximum likelihood estimation. In contrast, [Lawrence, 2004, Lawrence, 2005] introduces an alternative approach and suggests a novel probabilistic interpretation of PCA called dual probabilistic PCA (DPPCA). From a Bayesian perspective, the probabilistic model (2.22) is fitted to the training data by marginalising over mapping parameters A and optimising with respect to the latent variables X . The generative model of DPPCA follows regression equations (2.23) and (2.24), whereas the corresponding Gaussian noise and the noise model itself are given by (2.25) and (2.26) respectively. The key innovation is that a zero mean and spherical Gaussian prior is imposed over the generative function parameters A in each dimension of Y instead on the latent variables:

$$p(A) = \prod_{i=1}^{\mathbf{D}} \mathcal{N}(a_i \,|\, 0, I) \tag{2.41}$$

where a_i is i*th* row of the weight matrix $A = [a_1, a_2, \ldots]^T$ ($\mathbf{D} \times \mathbf{d}$). As a result, the marginal likelihood of observed data for every dimension $j = 1..\mathbf{D}$ is obtained by integrating over a space of mappings:

$$p(y_j \mid X,\psi) = \int \prod_{i=1}^{\mathbf{N}} p(y_{ij} \mid x_i, A, \psi) p(A) dA = \mathcal{N}(y_j \mid \mu, \Sigma) \qquad (2.42)$$

where the precision of noise model is expressed by the standard PPCA covariance matrix $\psi = I\sigma^2$ and the covariance matrix of the distribution is given by:

$$\Sigma = XX^T + \psi \qquad (2.43)$$

Here, the observed data Y is presumed to be centred at origin $Y = Y - \mathbf{N}^{-1}\sum_{i=1}^{\mathbf{N}} y_i$, thus, the mean is taken to be zero $\mu = 0$. Since the different dimensions of Y are expected to be independent, the complete joint likelihood of all observed data dimensions given the latent positions is:

$$p(Y \mid X,\psi) = \prod_{j=1}^{\mathbf{D}} \mathcal{N}(y_j \mid 0, \Sigma) = \frac{1}{(2\pi)^{\mathbf{DN}/2} \left|\Sigma\right|^{\mathbf{D}/2}} \exp(-\frac{tr(\Sigma^{-1}YY^T)}{2}) \qquad (2.44)$$

The maximisation of the above likelihood is equivalent to minimising the negative log likelihood of the model:

$$L(X) = -\ln p(Y \mid X,\psi) = \frac{1}{2}(\mathbf{ND}\ln 2\pi + \mathbf{D}\ln\left|\Sigma\right| + tr(\Sigma^{-1}YY^T)) \qquad (2.45)$$

The optimisation of the above objective function is performed by taking gradients of (2.45) with respect to the latent variables X [Magnus and Neudecker, 1999] and solving the eigenvalue problem equivalent to standard PCA [Lawrence, 2004, Lawrence, 2005].

2.2.2.3.2.2.2.2. Gaussian Process

Gaussian processes [O'Hagan, 1992, Williams, 1998, Rasmussen and Williams, 2006] (GPs) are a class of probabilistic models which specifies a distribution over a function space on points where the function is instantiated. GP can be seen as a natural generalisation of multivariate Gaussian random variables, where the Gaussian process describes a whole function over a finite number of variables. In the context of dimensionality reduction, any GP is parameterised completely by a mean function $\mu : \mathbb{R}^{\mathbf{d}} \to \mathbb{R}^{\mathbf{D}}$ and a covariance function, or kernel, $k : \mathbb{R}^{\mathbf{d}} \to \mathbb{R}^{\mathbf{D}}$.

Both functions must be of the space on which the process operates, so that a Gaussian distribution over an entire space of functions, $s:\mathbb{R}^{\mathbf{d}} \rightarrow \mathbb{R}^{\mathbf{D}}$, is given by:

$$\begin{aligned} & s \sim \mathcal{N}(\mu, \Sigma) \\ & \Sigma = \left\{ k_{ij} \mid i, j = 1..\mathbf{N} \right\} \end{aligned} \tag{2.46}$$

Usually, the mean function is taken to be zero ($\mu = 0$), whereas the covariance function k characterises the nature of the functions that can be sampled from the process and it is constrained to produce positive definite matrices Σ (i.e. satisfies Mercer's condition [Mercer, 1909, Courant and Hilbert, 1953]). GP regression adjusts the parameters Φ of the covariance matrix Σ over the space of functions in order to maximise the likelihood of the observed data given the GP [Rasmussen and Williams, 2006].

Let’s consider a simple GP prior over the space of functions that are fundamentally linear with additive Gaussian noise of varianceσ^2. The covariance function $k_{ij} = \kappa(x_i, x_j) = x_i^T x_j + \sigma^2 \delta_{ij}$ for such prior is evaluated on the whole embedding X to produce the following covariance matrix of the process where $\psi = I\sigma^2$:

$$\Sigma = XX^T + \psi \tag{2.47}$$

Note that the above expression can be recognised as the covariance matrix associated with each dimension of the marginal likelihood for DPPCA ((2.42) and (2.43)). For this reason, the complete marginal likelihood (2.44) can be seen as a product of **D** independent GPs, where each of them is associated with a linear covariance function.

2.2.2.3.2.2.2.3. Gaussian Process Latent Variable Model

GPLVM-based approaches aim at constructing a continuous d-dimensional latent space for D-dimensional data by defining a smooth nonlinear transformation from the latent to the observation space using a GP model on a training set of data points

[Lawrence, 2004, Lawrence, 2005]. A GP prior is imposed on a mapping function f in every dimension of the high dimensional space according to (2.46):

$$p(f \mid \Phi) = \prod_{j=1}^{\mathbf{D}} p(f_j \mid \Phi) = \prod_{j=1}^{\mathbf{D}} \mathcal{N}(f_j \mid 0, \Sigma) \tag{2.48}$$

Therefore, the corresponding likelihood for the observed dimension y_j ($j = 1..\mathbf{D}$) is obtained through marginalizing the mapping function f_j (in particular, the linear function defined in (2.42)):

$$p(y_j \mid X, \Phi) = \int \prod_{i=1}^{\mathbf{N}} p(y_{ij} \mid x_i, f_j, \Phi) p(f_j \mid \Phi) df_j = \mathcal{N}(y_j \mid 0, \Sigma) \tag{2.49}$$

As a result, since a GP model is completely specified by the covariance matrix Σ, a rich and flexible probabilistic distribution is defined. Thanks to this the linear covariance function ((2.43), (2.47)) of DPPCA can be replaced with a non-linear kernel function in order to produce a global and differentiable nonlinear mapping from latent to data space. A common choice for nonlinear covariance function is a radial basis function (RBF) because it smoothly interpolates the latent space [Lawrence, 2004, Lawrence, 2005] and satisfies Mercer's condition [Mercer, 1909, Courant and Hilbert, 1953]:

$$\begin{aligned} k_{ij} &= \kappa(x_i, x_j) = \alpha \exp(\frac{\gamma}{2}(x_i - x_j)^T (x_i - x_j)) + \sigma^2 \delta_{ij} \\ \Sigma &= \left\{ k_{ij} \mid i, j = 1..\mathbf{N} \right\} \end{aligned} \tag{2.50}$$

where the kernel hyperparameters $\Phi = \{\alpha, \sigma^2, \gamma\}$ respectively determine the output variance, the variance of the additive noise and the RBF width. δ_{ij} is the Kronecker delta function.

In general, there is no closed form solution for maximising (2.44) when nonlinear kernel functions are employed (2.50). Therefore, the learning process is performed using two-stage maximum a posterior (MAP) estimation. First, the latent variables are initialised, usually using PPCA or any spectral method. Secondly, latent positions and the hyperparameters are optimised iteratively until the optimal solution is reached. According to Bayes theorem, this is achieved by maximising the likelihood (2.44) with respect to the latent positions, X , and the hyperparameters, Φ using the following posterior:

$$p(X,\Phi \mid Y) \propto p(Y \mid X,\Phi)\, p(X)\, p(\Phi) \qquad (2.51)$$

where the priors of the unknowns are: $p(X) = \mathcal{N}(0,I)$, $p(\Phi) \propto \prod_i \Phi_i^{-1}$. These priors are introduced to prevent overfitting on small training sets [Grochow et al., 2004, Lawrence, 2005]. The maximisation of the above posterior is equivalent to minimising the negative log likelihood of the model with respect to X and Φ:

$$\begin{aligned} L(X,\Phi) &= -\ln p(X,\Phi \mid Y) = \\ &= \frac{1}{2}((\mathbf{DN}+1)\ln 2\pi + \mathbf{D}\ln\left|\Sigma\right| + tr(\Sigma^{-1}YY^T) + \sum_i \left\|x_i\right\|^2) + \sum_i \Phi_i \end{aligned} \qquad (2.52)$$

This optimisation process is performed numerically by taking the gradients of $L(X,\Phi)$ with respect to the kernel Σ and then combining them with the kernel gradients with respect to the latent positions X and the model parameters Φ through the chain rule. These gradients are used in combination with (2.52) in a non-linear optimiser to obtain a final latent variable model of the data. Typical numerical optimisation methods which are employed in this task include the Levenberg-Marquardt method [Levenberg, 1944, Marquardt, 1963], conjugate gradient [Johansson et al., 1992], scaled conjugate gradient [Möller, 1993] or L-BFGS [Nocedal and Wright, 2006],

2.2.2.3.2.2.2.4. Extensions

Recently, many researchers have exploited GPLVM in a variety of applications, thus designing a number of GPLVM-based extensions. The main ones are summarised in this section.

The learning process of standard GPLVM is computationally very expensive, since $O(N^3)$ operations are required in each gradient step to inverse the kernel matrix Σ (2.52). Therefore, in practice, a sparse approximation to the full

Gaussian process, such as ’fully independent training conditional’ (FITC) approximation [Lawrence, 2007, Urtasun et al., 2007] or active set selection [Lawrence, 2004], is exploited to reduce the computational complexity to a more manageable $O(m^2N)$ where m is the number of points involved in the sparse approximation [Lawrence, 2007]. The approximation process requires an additional set of representative variables, so called inducing variables [Lawrence, 2007] or active points [Lawrence, 2004], that are used in the lower rank approximation of the covariance Σ. Unfortunately, the number of inducing variables or active points has to be chosen empirically, since there is no optimal way to automate this process [Urtasun et al., 2007]. The selection of the wrong number of representative variables may come with the risk of overfitting or poor generalisation potential to unseen samples.

The different data dimensions have different intrinsic scales (or, equivalently, different levels of variance). This means that a small change in one dimension may have a larger impact on the observed space than a change in another dimension. To address this problem, scaled GPLVM [Grochow et al., 2004] (SGPLVM) generalises the GP models by introducing scaling parameters to account for different variances in the output dimensions of (2.49).

Since GPLVM focuses primarily on modelling the data global structure, there is no guarantee that the data local structure is retained in the latent space. The smooth mapping in GPLVM ensures that dissimilar points in a data space remain distant in a latent space. However, there is no constraint to prevent two points which are close in data space to be placed far apart in the latent space. A more faithful preservation of the observed space topology was supported by imposing high dimensional constraints on the latent space. Back Constrained GPLVM [Lawrence and Quinonero-Candela, 2006] (BC-GPLVM) enforces local distance preservation through the form of a kernel based regression mapping from the observed space to the latent space. Locally linear GPLVM [Lawrence and Quinonero-Candela, 2006] (LL-GPLVM) extends this concept by defining explicitly a cylindrical topology to maintain. This is achieved, first, by constructing advanced similarity measures (i.e. kernels) to reflect a priori knowledge in the back constrained mapping function. Secondly, a distance metric is adjusted in the LLE objective function [Roweis and Saul, 2000] and incorporated into the GPLVM framework to reflect a domain specific prior knowledge about observed data. Otherwise, Observation Driven GPLVM [Gupta et al., 2008] (OD-GPLVM) relates two different high dimensional observation spaces, e.g. image feature space and motion capture space, using a single latent space. This is achieved by learning a discriminative embedding from the observation image feature space to the latent space in addition to the standard generative mapping from the latent space to the observation pose space. As a result, OD-GPLVM aims at preservation of local distances of both observation spaces at the same time.

Alternatively, Gaussian Process Dynamical Model [Wang et al., 2006, Wang et al., 2008] (GPDM), augments SGPLVM with a dynamical model in the latent space by defining a nonlinear auto-regressive mapping on the latent space. The latent dynamical model favours preservation of local proximities between points. The GPDM is obtained by marginalizing out the parameters of both mapping processes and optimizing the latent coordinates of training data. In [Urtasun et al., 2006a] further smoothness of latent trajectories is encouraged by simply balancing the effect of the dynamics on the latent space based on the ratio between dimensions of data and latent spaces.

Finally, the problem of supervised data classification was addressed by integrating into GPLVM a prior distribution over the latent space that is derived from an adaptation of generalised discriminant analysis constraints [Urtasun and Darrell, 2007] or pairwise constraints [Wang et al., 2010].

2.2.2.3.2.2.2.5. Summary

The key strength of GPLVM approaches is a generative nonlinear probabilistic model which can be easily applied even to previously unseen data. However, their main limitation comes from the computational cost of their learning process which restricts their usage to relatively small datasets. Moreover, the objective function (2.52) is severely under-constrained in the general case. This means that the optimisation is very likely to converge towards local minima if the initialisation of the model is poor; hence, good initialisation is essential.

2.2.2.4. Projection Strategies

Once dimensionality reduction is performed, an important property of the method is its ability to generalise to a new unseen high-dimensional data point $\tilde{y}$ by embedding it using the existing low dimensional data representation. The process of transformation between high and low dimensional space is carried out by two contrary mapping (projection) functions (Figure 2.18). The forward mapping function:

$$G : Y \to X,\ Y \subseteq \mathbb{R}^{\mathbf{D}},\ X \subseteq \mathbb{R}^{\mathbf{d}} \tag{2.53}$$

projects data from a high dimensional space to the low dimensional space, whereas the inverse mapping function projects data in the opposite direction:

$$g : X \to Y,\ X \subseteq \mathbb{R}^{\mathbf{d}}, \mathrm{Y} \subseteq \mathbb{R}^{\mathbf{D}} \tag{2.54}$$

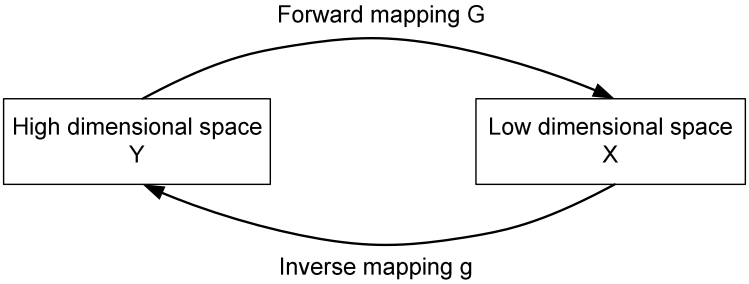


Figure 2.18. Mapping functions for generalisation of unseen samples of data.

The mapping functions are either an intrinsic property of a dimensionality reduction method or designed explicitly in a post processing step. The summary of basic properties together with available mapping functions in all discussed groups of dimensionality reduction methods are presented in Table 2.2.

Table 2.2. Overview of available mapping functions and basic properties of different algorithms. Convex: algorithms are considered convex if they have a unique solution, otherwise they may be subject to local optima.

	Mapping $Y \to X$	Mapping $X \to Y$	Nonlinear	Probabilistic	Global	Convex
PCA	Y	Y			Y	Y
PPCA	Y	Y		Y	Y	Y
KPCA	Y		Y		Y	Y
MDS					Y	Y
LLE, LE			Y			Y
Isomap, MVU			Y		Y	Y
FA	Y	Y		Y	Y	
MLLM	Y	Y	Y	Y		
GTM		Y	Y	Y	Y	
GPLVM		Y	Y	Y	Y	

2.2.2.4.1. Intrinsic Property of the Method

The forward mapping function is given directly by the dimensionality reduction process (2.5) and (2.13) for PCA and KPCA respectively. In the case of PCA, the corresponding inverse projection, or reconstruction of $\tilde{y}$ from $\tilde{x}$, is $\tilde{y} = A\tilde{x}$.

The principle of probabilistic LVMs is to focus on learning the mapping function during dimensionality reduction process. As a result, the forward mapping of PPCA and FA is equivalent to the dimensionality reduction function given by (2.30) and similarly the inverse mapping corresponds to the reconstruction function (2.31). In MLLM, first the posterior responsibility for generating a new data point is computed for every mixture in the model, and subsequently the projection is performed using the local mapping functions of the most probable mixture [Ghahramani and Hinton, 1997, Tipping and Bishop, 1999a]. Similarly to linear LVMs, GTM employs directly a reconstruction function f for inverse mapping according to equation (2.38).

The key advantage of GPLVM over other LVMs is that it provides a general-purpose probability distribution for new data points. In particular, the use of a GP to perform inverse mapping results in modelling uncertainty in the positions of the points in the data space. It can be shown that any point in X and especially any new one $\tilde{x}$ can be related with a data space as a Gaussian distribution [Williams, 1998]:

$$p(\tilde{y} \mid \tilde{x}, Y, X, \Phi) = \mathcal{N}(\tilde{y} \mid \mu(\tilde{x}), \sigma^2(\tilde{x}) I) \tag{2.55}$$

where the mean μ is the point that the model would predict for a given $\tilde{x}$, whereas the variance σ^2 indicates the uncertainty of this prediction (the certainty is greatest near the training data). Both are represented respectively by:

$$\begin{aligned} \mu(\tilde{x}) &= Y^T \Sigma^{-1} k(\tilde{x}, X) \\ \sigma^2(\tilde{x}) &= k(\tilde{x}, \tilde{x}) - k(\tilde{x}, X)^T \Sigma^{-1} k(\tilde{x}, X) \end{aligned} \tag{2.56}$$

In the general case, there is no closed form solution for estimating the latent position $\tilde{x}$ given a new data point $\tilde{y}$ in GPLVM. However, the forward mapping can be seen as a two-stage inference process [Grochow et al., 2004, Lawrence, 2005, Tian et al., 2005, Ek et al., 2007]. In the first stage the position on the latent space is initialised to the most likely x which may have generated the observed data $\tilde{y}$ according to (2.55). Afterwards, the position of $\tilde{x}$ is optimised by minimising the negative log likelihood of (2.55) using gradient descent optimisation

[Levenberg, 1944, Marquardt, 1963, Johansson et al., 1992, Möller, 1993, Nocedal and Wright, 2006]:

$$L(\tilde{x}) = \frac{\|\tilde{y} - \mu(\tilde{x})\|^2}{2\sigma^2(\tilde{x})} + \frac{\mathbf{D}}{2}\ln\sigma^2(\tilde{x}) + \frac{\mathbf{D}}{2}\ln 2\pi + \frac{1}{2}\|\tilde{x}\|^2 \tag{2.57}$$

where an isotropic spherical prior is imposed on the new latent position. Alternatively, BC-GPLVM provides directly the nonlinear forward mapping function [Lawrence and Quinonero-Candela, 2006, Urtasun and Darrell, 2007].

2.2.2.4.2. Out-of-sample Extension

The standard formulation of geometrically motivated approaches (LLE, LE, Isomap, and MVU) and MDS do not provide any explicit mapping between spaces. However, the forward mapping can be estimated by a nonparametric out-of-sample extension of these algorithms. For LLE, LE, Isomap, MDS, this is achieved by reinterpreting basic algorithms as the KPCA with method dependent kernel matrices and obtaining the eigenfunctions of these kernels through Nyström approximation [Bengio et al., 2003]. This allows embedding any new point using the standard KPCA forward mapping. Similar nonparametric out-of-sample extensions were proposed for Isomap in [De Silva and Tenenbaum, 2003, Choi and Choi, 2007]. MVU approximates the kernel eigenfunction using Gaussian basis functions [Chin and Suter, 2008].

2.2.2.4.3. Multilayer Perceptrons

Another possibility for designing mappings in geometrically motivated approaches (LLE, LE, Isomap, and MVU) as well as in MDS was presented in [Haifeng et al., 2006]. After the discovery of an embedded space, a multilayer feed-forward neural network, also referred to as a multilayer perceptron (MLP), is employed to simulate the projection procedure between spaces. The obtained low dimensional representation is used as supervision for a neural network training procedure.

Initially, the architecture of the neural network has to be designed [Haykin, 1998]. The neural network architecture can be seen as the organised topology of the interconnected neuron-like processing elements (Figure 2.19). These neurons are assembled into hierarchical layers. Any MLP is composed of one input layer, zero or more hidden layers and one output layer and is fully connected. This means that every neuron in each layer is connected in a weighted manner to every other neuron in the next layer and so on. Neurons in each layer are equipped with the same activation function, for instance tangent sigmoid function or pure linear function. Given such architecture, the learning process adjusts weights of synapses to best represent transformation from one space to another. The parameters of the network are determined using non linear optimisation techniques [Rumelhart et al., 1985, Johansson et al., 1992, Nocedal and Wright, 2006].

The forward mapping is performed with a neural network learned from a high to low dimensional space, where the input layer is composed of $\mathbf{D}$ neurons and the output layer consists of $\mathbf{d}$ neurons (Figure 2.19). The inverse mapping is carried out with another network trained in opposite direction with $\mathbf{d}$ neurons in the first layer and $\mathbf{D}$ in the last one.

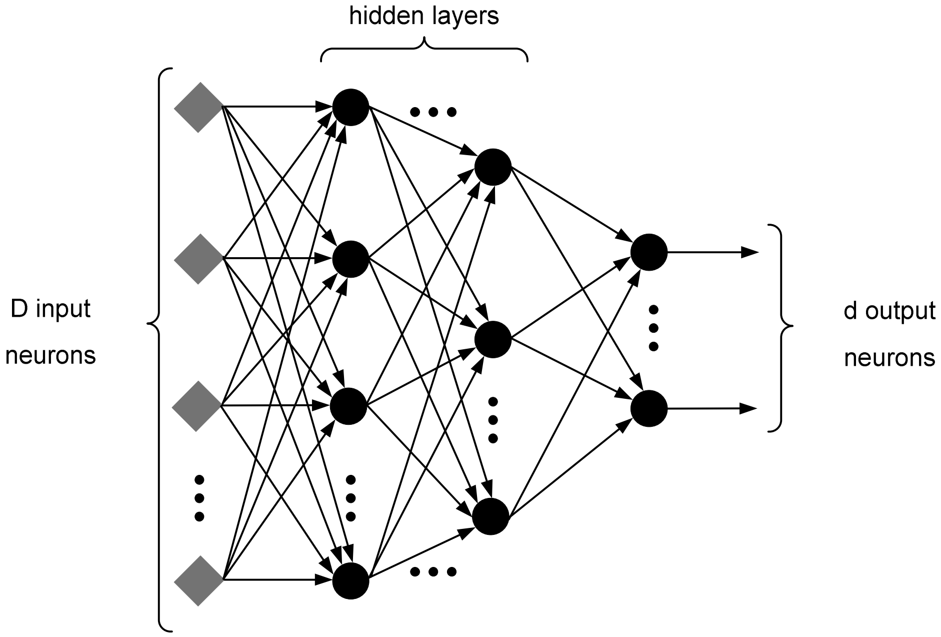


Figure 2.19. Forward mapping function using a multilayer feed-forward neural network.

The MLP is capable of approximating some nonlinear mappings [Lapedes and Farber, 1988] for a specific architecture [Haifeng et al., 2006]. In addition, it has better generalisation properties than the out-of-sample extension [Haifeng et al., 2006]. However, the main drawback of this approach is the manual process required to design the MLP architecture. It consists of:

- The selection of the number of hidden layers,
- The selection of the number of neurons in each hidden layer,
- The selection of the activation functions for neurons in hidden and output layers.

2.2.2.4.4. Generalised Radial Basis Function Network

Generalised Radial Basis Function Network (RBFN) is a conceptually simple and powerful alternative to the MLP model, which overcomes the problem of manual design of the network architecture. Whereas MLP may be viewed as a stochastic approximation, RBFN is motivated as a curve-fitting approximation problem in the high dimensional space [Haykin, 1998]. According to this viewpoint, the learning process is equivalent to finding a surface in a multidimensional space that provides the best fit to the training data in some statistical sense. Consequently, generalisation is performed by using this high dimensional surface to interpolate the new data. Note, that RBFN is a simpler variation of multilayer feedforward network which offers the comparable generalisation properties but in addition it is capable of implementing any nonlinear transformation [Haykin, 1998].

Figure 2.20 presents the architecture of a RBFN, which involves only three layers [Poggio and Girosi, 1990]. The input layer connects the network to its environment similarly to standard MLP. The second hidden layer applies a nonlinear transformation from the input space to a hidden space using an arbitrary radial basis functions (RBFs). The layer is parameterised by the RBF coefficients and centres, i.e. representative points in an input space which summarise the whole dataset. The number of neurons in the hidden layer corresponds to the number of centres $\mathbf{Z}$. The output layer is a weighted linear sum of the outputs of hidden units, providing the response of the network to the activation pattern which was supplied to the input layer. All layers are fully connected in a single direction as in MLP.

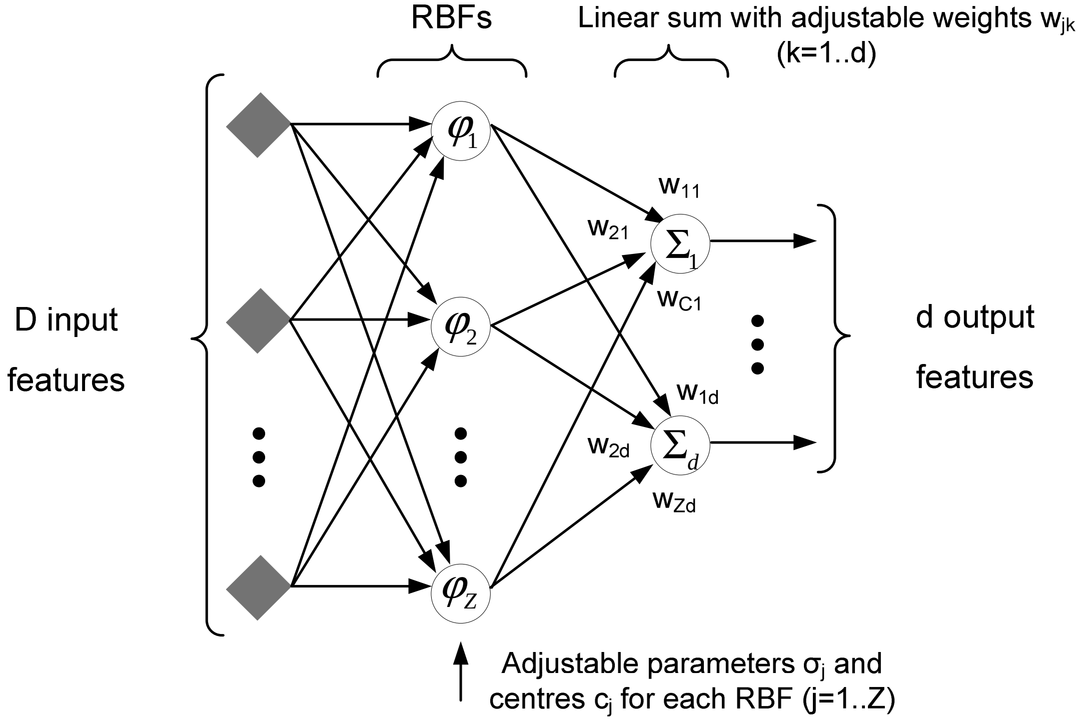


Figure 2.20. Forward mapping function using radial basis function network.

Let’s consider a highly nonlinear mapping g from a low dimensional space X to a high dimensional space Y. Such complex function is approximated by a combination of radial basis functions which are assumed to be linearly independent [Poggio and Girosi, 1990]:

$$g^k(x) = p(x) + \sum_{j=1}^{\mathbf{Z}} w_{jk}\varphi(\|x - c_j\|) \tag{2.58}$$

where k is the kth dimension in a high dimensional space, $p(x)$ is an optional linear low-degree polynomial term in the form: $p(x) = [1\,x]*\mathbf{t}$. φ is a real-valued basis function of $\mathbf{d}$ variables and w_{jk} are real coefficients. The RBFN structure is formed by the centres $C = \{c_j \mid j = 1..\mathbf{Z}\}$ which summarise the training data points in order to ensure generalisation properties of the network. The centres are determined in the input space using either the k-means clustering [Kanungo et al., 2002] or rival penalized competitive learning [Xu et al., 1993]. Finally, $\|*\|$ denotes the norm, usually Euclidean.

The radial activation function φ can be defined in various ways. However, it must satisfy Micchelli's theorem, which states that empirical kernel map $\psi(X)$ ($\mathbf{N}\times\mathbf{Z}$) [Schölkopf and Smola, 2002], which is constructed from these functions, must be nonsingular [Micchelli, 1986]. The entries of this interpolation matrix $\psi(X)$ are:

$$\psi(X)=\{[\varphi(\|x_i-c_1\|),\varphi(\|x_i-c_2\|),...,\varphi(\|x_i-c_Z\|)]\,|\,i=1..\mathbf{N}\} \qquad (2.59)$$

There is a large class of radial basis functions [Powell, 1987] that is covered by Micchelli's theorem, it includes Gaussian functions:

$$\varphi(\|x_i-c_j\|)=\exp(-\|x_i-c_j\|^2/2\sigma^2) \qquad (2.60)$$

and thin plate spline:

$$\varphi(\|x_i-c_j\|)=\sqrt{\|x_i-c_j\|^2+cons^2} \qquad (2.61)$$

which are of particular interest in the research on RBFN [Poggio and Girosi, 1990, Haykin, 1998]. Because of its excellent approximation properties [Poggio and Girosi, 1990], the Gaussian basis function is exploited in this research, where σ is set to the average distance between all centres.

The interpolation mapping (interpolation surface) is expressed by an over-constrained nonlinear system of equations without taking into account polynomial term ($t=0$):

$$y=g(x)=\psi\left(x\right)A \qquad (2.62)$$

where A is a $\mathrm{Z}\times\mathbf{D}$ matrix of network weights for D different nonlinear mappings g^k. The training phase of RBFN is performed by estimating the coefficients A for the interpolation surface g based on known centres and data points presented to the network in the form of input-output examples. The solution for A is found by applying the Moore–Penrose pseudo-inverse [Penrose, 1955] on matrix $\psi(X)$ in equation (2.62) and solving the obtained linear system of equations:

$$A = \psi(X)^{+}Y \tag{2.63}$$

Consequently, equation (2.62) can be seen as the inverse mapping which allows projecting any new point from the embedded space into the high dimensional space. Similarly to MLP, the forward mapping can be simulated by another RBFN, which is learned in the same manner by swapping the input and output space $X \leftrightarrow Y$.

Learning the RBFN from a high to low dimensional space may be more challenging because of the dimensionality curse, especially if there is not enough data in relation to input dimensionality $\mathbf{D}$. In such scenario, the forward mapping is approximated by the inversion of the inverse mapping [Poggio and Girosi, 1990]. This is achieved by exploiting the polynomial term ($\mathbf{t}=1$) in the interpolation function (2.58), thus, the interpolation matrix (2.59) is extended to:

$$\psi(x) = \begin{bmatrix} \psi(x) & 1\,\mathrm{x} \\ (1\,\mathrm{C})^{T} & 0 \end{bmatrix} \tag{2.64}$$

and the solution for A is determined in the same way by equation (2.63). However, since the vector $\psi(x)$ has a special form thanks to the linear polynomial part in the interpolation function, the forward mapping is approximated by the inversion of equation (2.62):

$$\psi\left(x\right) = yA^{+} \tag{2.65}$$

and taking directly the last $\mathbf{d}$ columns from the reconstructed vector $\psi\left(x\right)$ as the embedded coordinates. A^{+} denotes the Moore–Penrose pseudo-inverse of matrix A [Penrose, 1955].

RBFN has been successfully applied as the projection functions on the embedded spaces produced by LLE [Elgammal and Lee, 2004a, He et al., 2004, Elgammal and Lee, 2007, Ohbuchi et al., 2008, Lewandowski et al., 2009], Isomap [Shi et al., 2005, Blackburn and Ribeiro, 2007, Ohbuchi et al., 2008, Lewandowski et al., 2009] and LE [Ohbuchi et al., 2008, Lewandowski et al., 2009].

2.2.2.5. Summary of Feature Extraction Methods

Feature extraction by dimensionality reduction is a very powerful approach and has proved to be more flexible than feature selection, since it has been successfully applied in a variety of application domains including computer vision [Tian et al., 2005, Urtasun et al., 2006a, Hou et al., 2007, Wang et al., 2008], image processing [He et al., 2004], computer graphics [Grochow et al., 2004, Urtasun et al., 2008, Deena and Galata, 2009], robotics [Shon et al., 2006, Bitzer and Vijayakumar, 2009], speech recognition [Jain and Saul, 2004, Takiguchi and Ariki, 2007, Singh-Miller et al., 2007, Jafari and Almasganj, 2010, Errity, 2010], data visualisation [Tenenbaum et al., 2000, Belkin and Niyogi, 2002, Lawrence, 2004] and pattern recognition [Yang, 2003, De Ridder et al., 2003, Urtasun and Darrell, 2007, Zheng et al., 2008, Wang et al., 2010]. As a consequence, recently extensive research is carried out in the feature extraction field, especially in the domains beyond the range of interest of feature selection algorithms where it is unfeasible to design an intuitive evaluation criterion, in particular computer vision.

In the rest of the thesis, the term 'dimensionality reduction' will refer to feature extraction branch of dimensionality reduction approaches.

2.2.3. Frameworks for Time Series

All discussed dimensionality reduction methods assume that the observed data samples are independent; therefore any temporal correlation present between data samples is not taken into consideration. While this is a valid assumption for many applications, there are many situations when temporal structure is a key intrinsic property of data, thus an alternative approach is desired. In particular, when dealing with time series data, the assumption of independence between data points is clearly inappropriate since points at each time step are expected to be highly correlated. Since many real datasets are time series, the quality of low dimensional representation can be improved by modelling the temporal dependencies between points. This information is exploited twofold.

In the first case, temporal constraints are employed as a valuable clue for dimensionality reduction process. For instance, a temporal neighbourhood preserving embedding [Wu et al., 2009] uses a simple temporal model to represent each point as a linear combination of its sequential neighbours through linear projection from high to low dimensional space. In contrast, a spatio-temporal Isomap [Jenkins and Mataric, 2004] is a nonlinear global approach designated for time series. Initially, the original distance weights in the graph of local neighbours are empirically altered to emphasise similarity between temporal related points. Afterwards, the temporal dependencies are propagated globally via a shortest-path mechanism. In the case of LVMs, BC-GPLVM includes temporal coherence constraints to ensure the smoothness of the mapping between spaces [Lawrence and Quinonero-Candela, 2006], whereas [Bishop, 1997] extends the GTM algorithm to capture temporal dynamics of sequential data by incorporating this information as an emission density in a hidden Markov model [Rabiner, 1989].

An alternative approach to model the rich complexity of time series is to first reduce dimensionality assuming no temporal coherence and then learn a dynamical model on the latent space. In [Lin et al., 2006], a dynamic Bayesian network is constructed by adding links among the intrinsic coordinates of the GCM to account for temporal dependency. As a result, a global linear dynamical model is incorporated into the latent space. To handle more complex dynamics, [Li et al., 2007c, Li et al., 2010] use a generalisation of the switching linear dynamical model [Pavlovic et al., 2001] on the low-dimensional globally coordinated latent space. In turn, GPDM and its variants integrate time information by associating nonlinear, autoregressive dynamic model to the embedded space [Wang et al., 2006, Urtasun et al., 2006a, Wang et al., 2008, Gupta et al., 2008].

2.3. Human Motion Analysis

2.3.1. Introduction

Over the last two decades, human motion analysis has been very popular due to the wide range of potential applications and its inherent complexity. This section reviews the main research effort regarding computer vision based human motion analysis, i.e. human pose recovery (section 2.3.2) and action recognition (section 2.3.3). A comprehensive survey of both fields is beyond the scope of this thesis, thus, first we provide a brief overview of the most promising lines of research in each field. Afterwards, we focus on the application of dimensionality reduction in both areas. The main motivation of this section is to provide some background information about these two important computer vision tasks on which evaluation of our contribution is performed.

In this research, human motion is defined in terms of a starting pose, ending pose and a sequence of continuous transitions that takes the human body from a pose at time t_1 to a pose at time t_2 . In turn, a human body 'pose' corresponds to the configuration of the various body parts in a body-centric coordinate system regardless of the chosen digital representation.

2.3.2. Human Pose Recovery

Pose recovery refers to a process of estimating configuration of articulated human body skeleton from a single monocular image or multiple images captured at the same time in a multi-view setting. Alternatively, tracking is a special case of pose estimation, which is formulated as inference of the human pose over a set of consecutive image frames from a video sequence. According to this definition, the goal of pose recovery is to localise a person's joints and limbs in either an image plan (2D recovery - Figure 2.21) or a world space (3D recovery - Figure 2.22), which usually results in the reconstruction of a human skeleton in a body centric coordinate system. Pose recovery from video footage is a very active and broad research field in computer vision. In this work, the scope of interest is limited to 3D pose recovery from videos, where 3D motion will be defined as sequences of 3D human body poses at successive time instants. The corresponding 3D motion reconstruction is formulated as the problem of recovery a sequence of 3D human poses. This section discusses the recent research progress in 3D pose recovery. More comprehensive reviews can be found in [Moeslund et al., 2006, Poppe, 2007b, Ji and Liu, 2010].

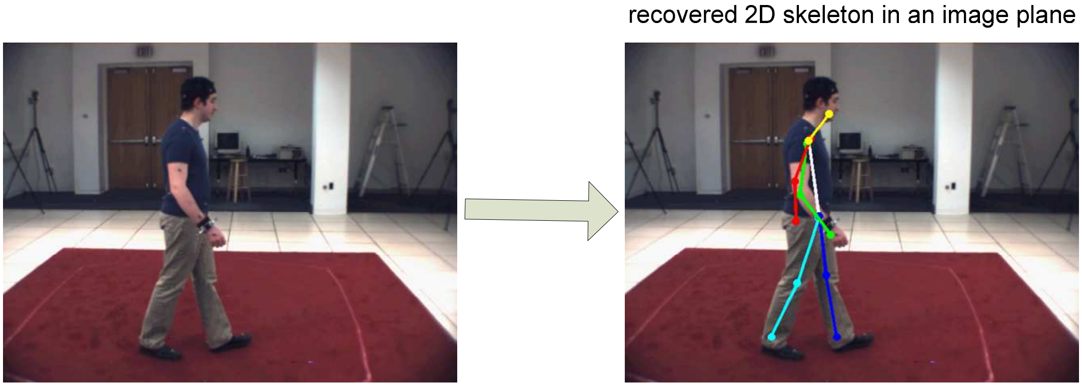

recovered 2D skeleton in an image plane

Figure 2.21. 2D pose recovery from a video frame.

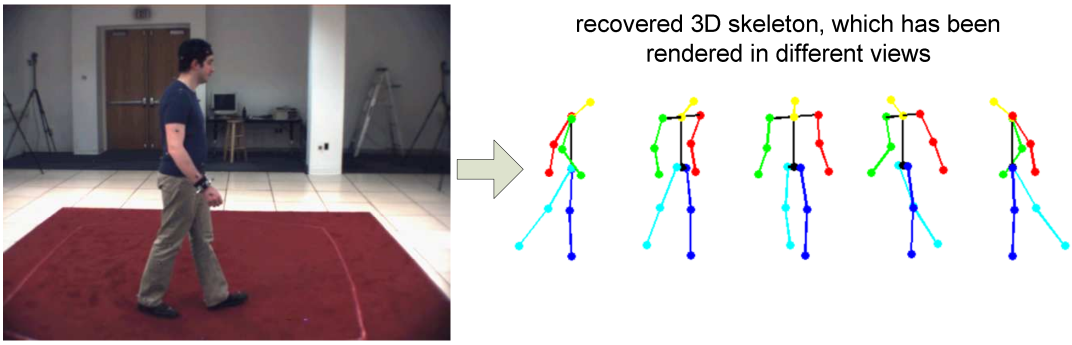

recovered 3D skeleton, which has been rendered in different views

Figure 2.22. 3D pose recovery from a video frame.

2.3.2.1. Activity Independent Methods

The most straightforward approach to 3D pose estimation from a monocular video is to compute the inverse kinematics from known 2D image positions of body joints under a simple scaled orthographic projection model [Taylor, 2000, Remondino and Roditakis, 2003, Barrón and Kakadiaris, 2003, Jian and Enhua, 2005]. Since such camera model handles only images with very little perspective effects, the perspective camera model was employed to overcome this limitation and deal with more realistic images [Zhao et al., 2005, Peng et al., 2009]. The main limitation of these approaches is that they require accurate detection of body joints in 2D image plane, which still remains a difficult problem in computer vision. Moreover, they presuppose an explicitly known parametric body model which is naturally constrained by body kinematics and dynamics.

2.3.2.2. Activity Constrained Methods

Activity-constrained learning approaches focus on learning the prior model of motion directly from carefully selected training data.

2.3.2.2.1. Object Tracking Framework

The problem of 3D motion reconstruction from images can be formulated as a Bayesian tracking process, where the objective is to construct statistical motion models from pre-recorded human motion data. These methods use an explicit 3D geometric representation of human shape and its kinematic structure to reconstruct a human posture by numerically optimising the similarity between observed images and predicted images rendered from a model. The 3D motion and pose extraction are usually implemented as a variation of tracking framework, especially the particle filter [Gordon et al., 1993]. The particle filter employs a stochastic sampling strategy for representing simultaneous alternative hypotheses. This is achieved by modelling arbitrary non-Gaussian probability density functions using a set of independent sample particles. The particle filter is derived from the Kalman filter and overcomes the constraint of a single Gaussian distribution [Kalman, 1960]. The performance of the particle filter depends on designing an appropriate sampling strategy which guides the tracking by reducing the complexity and size of the solution space.

Tracking in monocular video sequences has been addressed in a variety of different approaches [Brubaker et al., 2010]. A variant of the particle filter, called condensation algorithm [Isard and Blake, 1998], represents hypotheses by a spherical and randomly generated set which is iteratively propagated over time using a learned dynamical model. An annealed particle filter was presented by [Deutscher et al., 2000, Deutscher and Reid, 2005], that combines a deterministic annealing approach with stochastic sampling to gradually focus the search effort on promising areas of solution space. Alternatively, the hypothesis are generated by sequential importance sampling constrained by the prior over dynamics of the human body [Sidenbladh et al., 2000] or by a large database of example motions [Sidenbladh et al., 2002], to focus search in the neighbourhood of known trajectory paths. In [Sminchisescu et al., 2001, Sminchisescu and Triggs, 2003], the probable 3D body configurations over time are represented by a Mixture-of-Gaussians density model. A global search is performed by optimising a robust model-image matching cost metric which combines extracted edges, flow and motion boundaries, subject to 3D joint limits, non self-intersection constraints, and model priors. Model hypothesis are sampled from a defined distribution using cost-surface sensitive Covariance Scaled Sampling [Sminchisescu et al., 2001] or Kinematic Jump Sampling [Sminchisescu and Triggs, 2003]. Finally, in [Peursum et al., 2007], the stochastic search is guided by a variation of the hierarchical hidden Markov model to improve robustness of the particle filter against observation errors.

Multiple views reduce significantly depth ambiguity, and therefore may provide more accurate pose estimations. 3D visual hull reconstruction of a human body shape is a natural way for fusing information from multiple images which provides more informative cues about a recovered pose. In [Mikic et al., 2003], 3D visual hull representation is integrated within the extended Kalman filter tracking imposing angle limits. As a result, the system guarantees an automatic acquisition of a human body model and estimation of physically valid human postures. An alternative approach based on full 3D-to-3D nonrigid surface matching using spherical mapping is presented in [Starck and Hilton, 2005]. Alignment of a predefined skeletal model with the first frame allows the 3D motion to be recovered from the non-rigid surface motion over time. Recent work by [Caillette et al., 2005, Caillette et al., 2008] identifies Gaussian clusters of simple motions and trains a variable-length Markov model based on these clusters to direct local posture search towards better areas of the distribution.

The drawback of these approaches is that high dimensionality of observed features requires usage of many particles to sample a pose space with a sufficient density. Unfortunately, each particle comes with an increase in computational cost associated with the propagation of the particle according to a dynamical model and the evaluation of a likelihood function. In addition, a human body model has to be rendered and compared to extracted image descriptors for every particle. As a consequence, the optimisation process is expensive and requires good initialisation; and the problem always has many local minima. Another drawback of tracking in a high dimensional space is its sensitivity to the impoverishment sample problem, i.e a tendency of clustering particles on a very small region of the search space, therefore explicitly overconstraining the search space by decreasing the number of effective particles [King and Forsyth, 2000]. Further discussion about tracking frameworks can be found in [Wang and Rehg, 2006], where several common tracking schemes are evaluated quantitatively.

2.3.2.2.2. Example Based Approach

Example-based methods are two-stage approaches that first collect an image database of silhouettes from various viewpoints with corresponding 3D poses to adequately cover the entire space of possible solutions. Then, the pose estimation is conducted by similarity checking between the stored examples and a given image query.

For a pose recovery from monocular images, a representative system was presented in [Poppe, 2007a], where silhouettes images are encoded as a variant of histogram of oriented gradients [Dalal and Triggs, 2005] and query matching is performed with an entire training set using the Manhattan distance. A computationally more attractive approach was introduced in [Shakhnarovich et al., 2003], where parameter-sensitive hashing is applied on examples to speed up searching process. Another method was proposed by [Mori and Malik, 2006] in which the 2D joint locations in a query image are inferred according to stored examples using the technique of shape context matching in conjunction with a kinematic chain-based deformation model. Then the 3D posture is estimated based on the scaled orthographic projection algorithm introduced by [Taylor, 2000]. An alternative approach was proposed in [Agarwal and Triggs, 2006], where instead of explicitly storing and searching for similar training examples, a relevance vector machine [Tipping, 2000] is employed to learn a nonlinear regression of joint angles against histogram of shape context descriptors derived from silhouettes [Belongie et al., 2002]. As a result, a single compact model that has good generalisation to unseen examples is produced and employed for human pose estimation. A data-driven iterative approach is presented in [Lee and Cohen, 2006], where pose candidates are generated in a Markov chain Monte Carlo search guided by image observations.

When videos from multiple views are available, a human body posture can be inferred directly from a reconstructed 3D visual hull using a support vector machine trained on appearance-based shape descriptors [Cohen and Li, 2003].

The main limitation of these techniques is that a very large training set is required to provide satisfactory accuracy and generalisation properties.

2.3.2.2.3. Learnt Motion Model

Human motion resides in a very high dimensional space because of its complexity and rich dynamic. However, many studies have revealed that the space of many activities is intrinsically a low dimensional nonlinear subspace embedded in the high dimensional space [Grochow et al., 2004, Elgammal and Lee, 2004a, Lee and Elgammal, 2006b, Urtasun et al., 2006a, Ek et al., 2007, Hou et al., 2007, Elgammal and Lee, 2009]. Therefore, the reduction of data dimensionality to constrain vision based reconstruction of human movement from a single camera has become a very active research topic.

The pose recovery process shares some conceptual similarities with example based approaches. First, a low dimensional human body motion model is learned by reducing dimensionality of training MoCap data. Afterwards, the obtained model is used for pose estimation. Dimensionality reduction decreases computational and memory complexity of the pose estimation process in comparison to example based approaches. Moreover, it does not require an extensive training set of feasible human motions to generalise well to unseen data. In principle, the low dimensional representation can be exploited twofold for pose estimation.

2.3.2.2.3.1. Direct Approach

The straightforward approach is to use a directly learned model of human motion for inferring 3D poses.

A pose inference is formulated as estimating an embedding point on the low dimensional manifold which is subsequently projected back to the pose space. In [Elgammal and Lee, 2004a], a view based low dimensional representation of activity is discovered using LLE. Afterwards, mapping functions (i.e. RBFN) are learned between obtained representations and both the visual input space and the 3D body configuration pose space. The body pose is recovered in a closed form in two steps by projecting a new observed silhouette to the learned representations of the activity manifold followed by interpolating the 3D pose. On the basis of this work, the authors in [Lee and Elgammal, 2006b] present a new generative model to represent shape deformations according to view and body configuration changes on a conceptual two dimensional torus manifold. Similarly to previous work, the activity model is extended with a RBFN mapping functions, i.e. a style adaptive mapping function from a visual space to the low dimensional space and standard mapping function from the embedded to the pose space.

Another approach is to interpret a pose estimation procedure as an optimisation process. The space of human motion is parameterised by a set of linear subspace models obtained using PCA. Such parametric motion model is then used to formulate and restrict the tracking framework as a minimisation of differentiable objective function using a deterministic gradient descent optimisation [Urtasun et al., 2006b]. Alternatively, the parametric subspace model is used in construction of a generative human body motion model to constrain the solution space [Chen and Chai, 2009]. During pose inference the generative model is continuously deformed to best match 2D joint trajectories derived from monocular video sequences using a proposed gradient-based multi-resolution optimisation process.

[Grochow et al., 2004] presents a probabilistic framework which is based on a learned model of human poses and an inverse kinematics system. The model is obtained using SGPLVM which provides the probability distribution over all possible 3D poses and constrains pose reconstruction from known 2D joint locations extracted from images. Since it is difficult to obtain a 2D skeleton from an image, [Ek et al., 2007] proposes a shared and generative activity model, which encapsulates silhouette observations, joint angles and their dynamics using

GPLVM. As a result, the 3D pose is inferred directly from the model given a query silhouette.

2.3.2.2.3.2. Tracking Approach

The low dimensional motion model can be also incorporated in a 3D visual tracking framework to reduce the state space of tracker and to provide powerful human motion priors for a pose recovery. As a result, a particle filtering with the reduced state space is faster since significantly fewer particles are required to adequately approximate the state space posterior distribution.

For instance, [Elgammal and Lee, 2009] exploits a low dimensional torus manifold [Lee and Elgammal, 2006b] to constrain the particle filter tracker. Such torus is a natural continuous, low-dimensional representation of the joint (view and configuration) distribution which allows accurate 3D motion reconstruction. From a probabilistic perspective, the low dimensional model is learned using balanced GPDM [Urtasun et al., 2006a] or observation driven GPDM [Gupta et al., 2008] to ensure a continuous embedding of movement in the latent space for robust tracking and motion reconstruction. A conceptually equivalent approach is also presented in [Li et al., 2007c, Li et al., 2010], where the space of human motion is reduced using GCM [Li et al., 2007c] or LLC [Li et al., 2010] to provide prior information for a Multiple Hypothesis Tracker [Cham and Rehg, 1999]. In contrast, recent work [Guo and Qian, 2008] discovers two separate low-dimensional manifolds; one for silhouettes and one for 3D poses using GPLVM and balanced GPDM respectively. Then, bidirectional mappings between these two manifolds are established using a Bayesian mixture of experts [Xu et al., 1995] and relevance vector machine [Tipping, 2000]. The resulting motion model is used as a strong prior in the particle filter to explore the articulated space of human motion.

In [Hou et al., 2007], the problem of 3D pose estimation in a multiple view scenario is considered. First, the low dimensional embedding of example motions is learned using BC-GPLVM. Then, the latent space is partitioned into elementary motion sequences using an unsupervised EM clustering algorithm. The temporal dependencies between these elementary movements are efficiently captured by a Variable Length Markov Model. Tracking is then formulated in a particle filter based framework with a volumetric reconstruction algorithm to evaluate each candidate pose against image evidence captured from multiple views.

The main limitation of all already discussed methods is that they take into account only one particular activity in the learning process. To handle multiple activities, [Darby et al., 2010] defines a number of activity models obtained with PCA, each composed of a pose space with a unique dimensionality and an associated dynamical model. Consequently, each learned model is capable to recover a particular class of activity. Finally, all activities models are combined in a new variant of an annealed particle filter to perform robust 3D human motion reconstruction.

2.3.2.3. Dataset and Metrics

While research on articulated human motion and pose estimation has progressed rapidly in the last few years, a requirement for systematic quantitative evaluation of competing methods has emerged to establish the current state of the art. Although many datasets have been proposed (INRIA perception multi-cam dataset [Knossow et al., 2008], CMU MoCap dataset [CMU, 2010], CMU MoBo dataset [Gross and Shi, 2001]), HUMANEVA [Sigal et al., 2010] is the most extensive and established dataset for evaluation of human pose and motion estimation. The HUMANEVA dataset was collected using a hardware system in a laboratory setting. It provides:

- Synchronised videos recorded simultaneously by 3 and/or 4 cameras.

- Ground-truth 3D motion of the body captured using motion capture system (MoCap).
- 4 subjects (Figure 2.23) performing a set of 6 predefined actions in three repetitions (twice with video and motion capture, and once with motion capture alone).

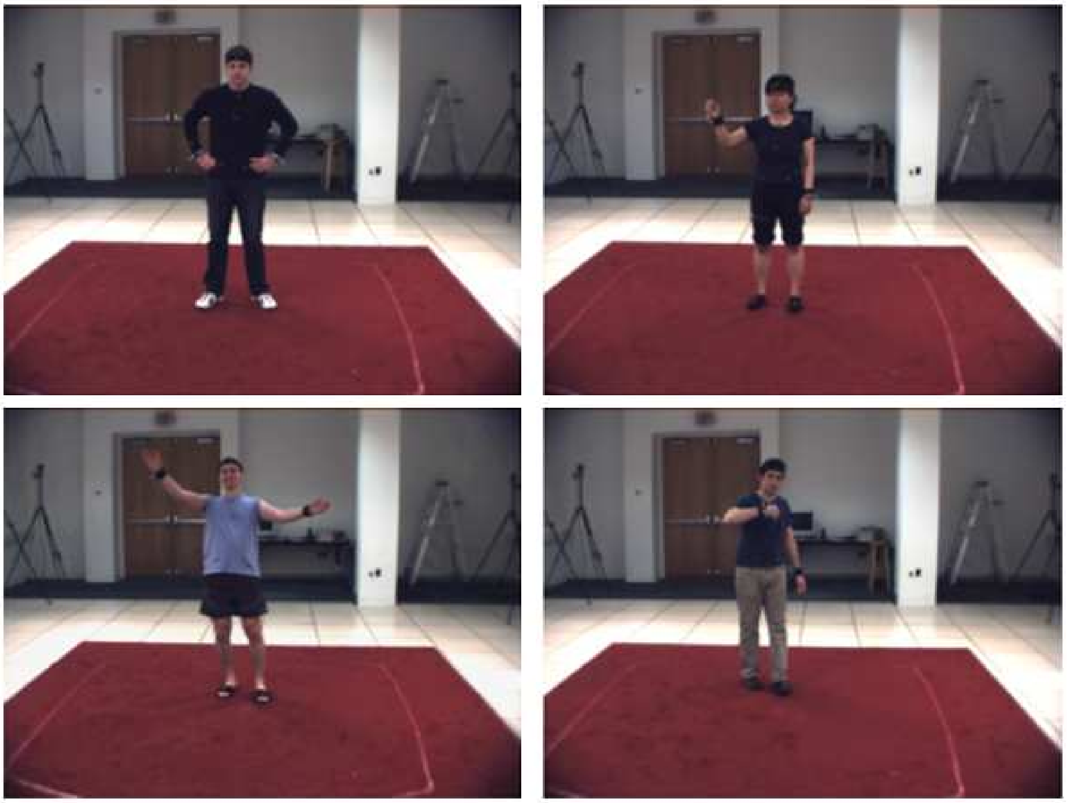


Figure 2.23. Four subjects available in HumanEva dataset.

2.3.2.3.1. Human Body Pose Description

In 3D pose recovery, a human body pose descriptor is usually based on an articulated hierarchical skeleton (Figure 2.24) which consists of joints and connecting rigid segments (i.e. bones) organised in a tree structure [Poppe, 2007b]. The joint is a connection point at which bone can rotate with respect to its parent. Lengths of bones are usually expected to satisfy the human body proportions, e.g. these defined by the Leonardo Da Vinci [Vinci, 1492]. In this thesis, the human skeleton model is composed of 13 joints (Figure 2.24). During motion, the skeleton is constrained by the 3D body kinematics and dynamics as well as the specific dynamic of the action being performed.

The learning of the prior model of human kinematics can be performed using data collected with marker-based human motion capture systems. The known correspondence between markers and joints together with the reconstructed 3D marker trajectories during movement provide the skeleton joint positions for each pose (15 dots in Figure 2.24).

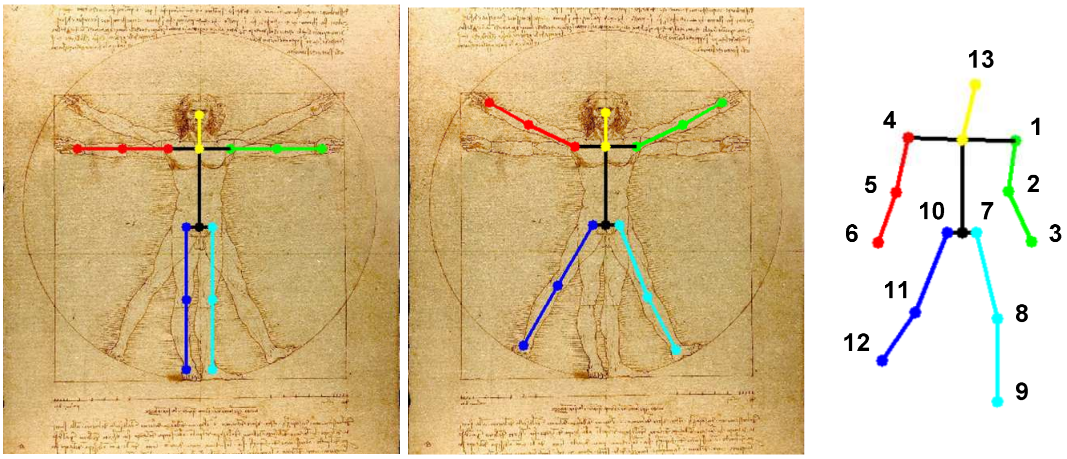


Figure 2.24. The Leonardo da Vinci human model and human skeleton model composed of 13 joints. The Leonardo da Vinci human model expresses the ideal human skeleton proportions of the body.

In the hierarchical model, the global position of the human body is defined at the root of the hierarchy. All other joints are located relatively to the parent following a hierarchical kinematic chain (or kinematic tree). Any moving object, either the entire skeleton or a particular joint in the skeleton, has some degrees of freedom (DOF). The term 'degree of freedom' refers to the number of parameters or variables that are allowed to vary independently from each other [Good, 1973]. In human motion, degree of freedom describes the number of ways in which an object can move [Rose and Christina, 2005]. An object can have at most 6 degrees of freedom in the three dimensional space, since in each dimension there are two possible types of movement: translation and rotation. When using a hierarchical model, joints are usually not allowed any translation. Here, apart from the root joint which has 6 DOFs, because it is responsible for moving the entire object in the three dimensional space, other joints have between one and three DOF and each DOF correspond to orthogonal rotation around one of the axis in a 3-dimensional space. The angle of each DOF is expressed either using Euler angles or quaternion. Moreover, each DOF can be constrained by minimum and maximum values using the human body kinematics constraints. The hierarchical skeleton model with the local angle representation for each joint allows to normalise the motion capture data. As a result, a human motion can be defined in a new coordinate system which is centrered on a moving person. This is crucial for extracting the intrinsic pattern of a motion, which is expected to be independent from the global position and rotation of the human with respect to the camera.

Quaternion [Hamilton, 1844] is a 4-dimensional vector which expresses orientations and rotations of objects in three dimensions. Any 3-dimensional rotation is described by just one real value angle and a vector of 3 imaginary dimensions (Figure 2.25):

$$q = a + ix + jy + kz \tag{2.66}$$

In comparison to Euler angles, quaternions are simpler to compose and can be smoothly interpolated. Moreover, quaternions avoid the problem of gimbal lock, i.e. the loss of one degree of freedom in a 3-dimensional space when two gimbals are in the same plane (a gimbal is a pivoted support that allows the rotation of an object about a single axis).

As a consequence, quaternions have the flexibility which make them particularly suitable for modelling local angles between joints in a skeleton according to a 3D kinematic tree.

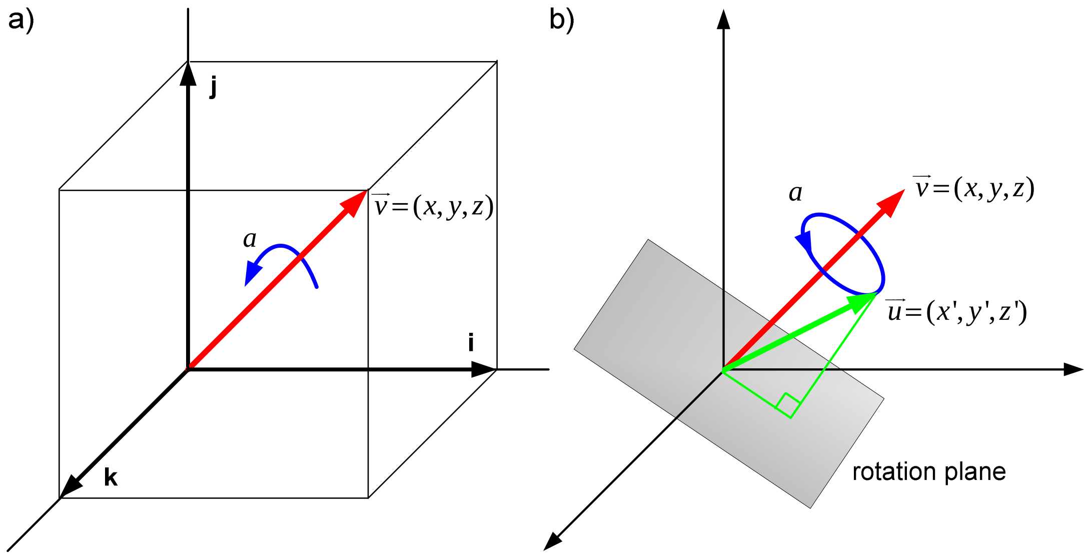


Figure 2.25. a) a 4-dimensional quaternion $(a,\vec{v})$; b) a rotation of vector $\vec{u}$ by the quaternion $(a,\vec{v})$.

The pre-processing of MoCap data is summarised as follows. First, all poses are converted into *normalised poses* [Elgammal and Lee, 2009], i.e. poses invariant to the subject's global rotation and translation. Then, the three angles defining each joint position are computed and represented by a single quaternion. An articulated human skeleton is then parameterised as a high dimensional feature vector by simply concatenating quaternions one by one for all joints in a single row vector. In this work, a 52-dimensional feature vector is constructed for each pose (13 joints multiplied by the 4-dimensional vector).

2.3.2.3.2. Evaluation Metrics

Various evaluation measures have been proposed for human motion tracking and pose estimation (see [Sigal et al., 2010] for overview). In this work, two metrics are exploited to evaluate estimated poses against motion capture data, which is our ground truth. Firstly, this thesis reports mean (over all angles) absolute difference errors between the true and estimated joint angle vectors (in degrees) to show performance independent on the skeleton limb sizes:

$$MAE(°)=\frac{1}{\mathbf{M}}\sum_{i=1}^{\mathbf{M}}\left|(x_i-x'_i)\bmod 180°\right| \tag{2.67}$$

Secondly, Root-Mean-Square error (RMS) is computed to facilitate the comparison between the reconstructed body and the ground truth data when the properties of body models are known. This is performed using Procrustes Analysis [Seber, 2004], which determines a linear transformation (translation, rotation, and scaling) of the reconstructed body to best match the ground truth by minimising RMS.

2.3.3. Action Recognition

Vision-based human action recognition is a high level process of image sequence analysis. This is achieved by assigning action labels that best describe action instances, even when performed by different subjects under different viewpoints, and in spite of large differences in manner and speed. In this work, we adopt the three level hierarchy of [Moeslund et al., 2006] to define the following notions:

- An action primitive (i.e. atomic action) is a simple motion pattern usually executed by a single person and typically lasting for a short duration (e.g. ‘jumping’, ‘running’, ‘sitting’, ‘drinking’).
- An action is a sequence of action primitives, which represent a more complex movement in a longer period of time (e.g. the ‘jumping hurdles’ action contains ‘starting’, ‘jumping’ and ‘running’ action primitives).
- An activity contains a number of successive actions performed by several humans who could interact with each other in a constrained manner (e.g. the ‘hurdling race’ activity which involves several people performing the ‘jumping hurdles’ action followed by the ‘resting’ action).

Most action recognition systems are composed of two pipelines: one for training (Figure 2.26) and one for classification (Figure 2.27). In the training phase

(Figure 2.26), first, relevant features are extracted from image sequences and used to produce a shape descriptor for each motion instance (section 2.3.3.1). Then, for each action, the shape descriptors are combined to create action models (section 2.3.3.2). In the classification phase (Figure 2.27), videos are pre-processed in the same way as for training and compared with the learned action models to perform the semantic interpretation of the action (section 2.3.3.3).

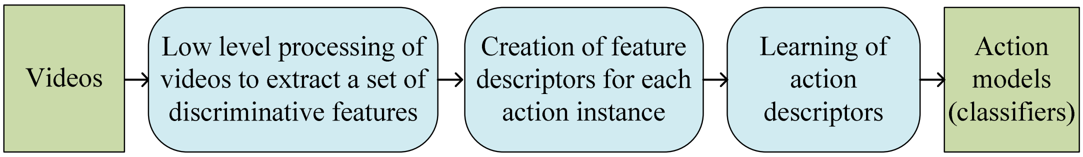

Videos
Low level processing of videos to extract a set of discriminative features
Creation of feature descriptors for each action instance
Learning of action descriptors
Action models (classifiers)

Figure 2.26. Training of action recognition framework.

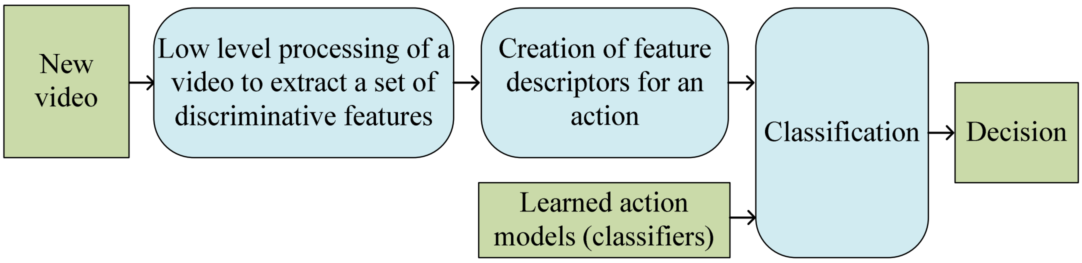

New video
Low level processing of a video to extract a set of discriminative features
Creation of feature descriptors for an action
Learned action models (classifiers)
Classification
Decision

Figure 2.27. A standard testing procedure for action recognition.

In this section, a state-of-art review of the very active field of human action recognition is presented. Although the identification of a human activity from a single video is the ultimate goal, we also report schemes based on multi-camera frameworks. Note that we limit our scope of interest to the most established and popular approaches in the research community with a special focus on recognition of action primitives. Therefore, if it is not stated otherwise, the term 'action' refers to 'action primitive' in the rest of the dissertation. A much more detailed overview of current advances in the field is provided by the surveys [Moeslund et al., 2006, Turaga et al., 2008a, Poppe, 2010, Weinland et al., 2010a].

2.3.3.1. Feature Descriptors

A variety of features has been used in the human action recognition task. Ideally, these should generalise over small variations in a person appearance, background, and viewpoint and action execution. At the same time, the representations have to be sufficiently rich to allow for the robust classification of an action. The temporal aspect is usually essential in an action performance and, therefore, most features take the temporal dimension into consideration. Any feature descriptor is classified as either local or global representation.

2.3.3.1.1. Local Feature Descriptors

Local feature descriptors decompose an observed action into a collection of local patches, which capture shape and motion only in the neighbourhoods of pre-selected points using some image measurements. These ‘interest’ points are locations in space and/or time where sudden changes of movement occur in the video. These locations are assumed to be the most informative for the recognition of a human action. The spatial and temporal sizes of a patch are usually determined by the scale of the interest point. As a result, before the computation of final local descriptors (section 2.3.3.1.1.2), a detector is applied to select spatio-temporal locations of the interest points and scales in a video by maximising specific saliency functions (section 2.3.3.1.1.1). An overview and broad evaluation of different detectors and local descriptors can be found in [Wang et al., 2009].

2.3.3.1.1.1. Detectors

2.3.3.1.1.1.1. Harris Detector

Harris detector is a combination of corner and edge detectors based on the local auto-correlation response function [Harris and Stephens, 1988]. The idea is to detect locations in a spatial image where the image values have significant variations in both directions. Laptev et al. proposes the Harris3D detector [Laptev and Lindeberg, 2003, Laptev, 2005], which is more attractive to the action recognition community, since it extends the Harris detector into the spatio-temporal domain. It requires that a point will be considered as ‘interesting’ only if the image value in local spatio-temporal volume has large variations in the spatial as well as the temporal dimension. Points with such properties will be the spatial interest points with distinct location in the time corresponding to local spatio-temporal neighbourhoods with non-constant motion. The spatio-temporal extents of the detected points are estimated by maximizing the normalised spatio-temporal Laplacian operator over independent spatial and temporal scales.

2.3.3.1.1.1.2. Cuboid Detector

Cuboid detector is proposed by [Dollar et al., 2005] and it is based on the spatio-temporal response function which is calculated at every location in the spatio-temporal image volume. Interest points are local maxima of this function. The response function is calculated by applying separable linear filters to video sequences. This function is derived from the 2D Gaussian smoothing filter applied spatially and a pair of 1D Gabor filters applied only along the temporal dimension.

2.3.3.1.1.1.3. Hessian Detector

Hessian 3D detector [Willems et al., 2008] selects a set of spatio-temporal interest points which are at the same time scale-invariant (both spatially and temporally) and densely cover a video content. This is achieved by simultaneous localisation of points in the spatio-temporal domain and over both scales (spatial and temporal) using the determinant of the 3D Hessian matrix as a saliency measure. The Hessian detector is computationally very efficient since the determination of interest points is a non iterative procedure. Moreover, the authors use the approximate box-filter operations on an integral video structure to further speed up the detector.

2.3.3.1.1.1.4. Dense Sampling

Dense sampling is a very naive approach, which extracts video blocks at regular positions and scales in space and time. There are 5 dimensions to sample from: two spatial dimensions, one temporal dimension and two scales (spatial and temporal). The spatial and temporal sampling of 3D patches is usually performed with overlap at multiple scales.

2.3.3.1.1.2. Local Descriptors

2.3.3.1.1.2.1. Cuboids

The cuboids are extracted at interest point and contain the spatio-temporally windowed pixel values [Dollar et al., 2005]. They are inherently local in nature, and therefore capture the local appearance and motion information. The size of a cuboid is set to contain most of the volume of data that contributes to the response function at that interest point; specifically, the cuboid consists of all (grey scale) pixel values within an area of six times the scale at which it was detected. The cuboid descriptor can be the vector of:

- flattened cuboid values,
- a global histogram of cuboid values,
- local histograms of cuboids values which are computed in partitioned regions of the cuboid.

The dimensionality of the final descriptors is reduced using the PCA.

[Ta et al., 2010a] uses cuboids to formulate the action recognition as a graph matching problem, whereas [Zhao and Elgammal, 2008] employs them to describe an action as a discriminative set of spatio-temporal key frames. Alternatively, [Ta et al., 2010b] combines cuboid descriptors with spatio-temporal relations among them to design the novel concept of a pair wise feature descriptor.

2.3.3.1.1.2.2. Histograms of Oriented Gradient

Histograms of Oriented Gradient [Dalal and Triggs, 2005] (HOG) are based on evaluating well-normalised local histograms of image gradient orientations in a dense grid. The basic idea is that local object appearance and shape are often characterised sufficiently by the distribution of local intensity gradients or edge directions, even without precise knowledge of the corresponding gradient or edge positions. In practice this is implemented by dividing the gradient image window into small spatial regions (cells), for each cell accumulating a local 1-D histogram of gradient directions or edge orientations over the pixels of the cell. The vectors of all cells are concatenated to give one global feature vector for the image window. For better invariance to illumination, shadowing, etc., it is also useful to contrast-normalise the local responses before using them. This is done by accumulating a measure of local histogram energy over somewhat larger spatial regions (blocks) and using the results to normalise all of the cells in the block.

For example, [Kaâniche and Brémond, 2010] generates local motion signatures based on the schema: people detection/feature selection followed by HOG descriptor generation/tracking. Similarly to the cuboids [Zhao and Elgammal, 2008], the HOG is used to represent an action as time series of a few snapshots of human-body parts in their most discriminative postures, relative to other activity classes (key poses) [Brendel and Todorovic, 2010]. In turn, [Kaâniche and Brémond, 2009] extends the HOG into the temporal domain by tracking 2D descriptors based on frame-to-frame HOG tracker using the extended Kalman filter. To characterise local motion and appearance, the HOG is combined with optical flow field [Roth et al., 2009] or histograms of optical flow accumulated in space-time neighbourhoods of detected interest points [Laptev et al., 2008]. Eventually, the self-similarity descriptor is derived from the HOG to deal with view changes [Junejo et al., 2008].

Alternatively, [Kläser et al., 2008] introduces the spatio-temporal HOG (HOG3D), which is based on histograms of 3D gradient orientations and generalises the HOG concepts to 3D, assuming videos as spatio-temporal volumes. Gradients are computed using the integral video representation. The descriptor, therefore, combines shape and motion information at the same time. A given 3D patch is divided into spatio-temporal cells. The corresponding descriptor concatenates gradient histograms of all cells and is then normalised. [Weinland et al., 2010b] uses a local partitioning of the dense HOG3D representation in a hierarchical classifier, which first performs a local classification followed by global, to provide robustness to both viewpoint changes and occlusions.

The HOG has proved to be the very powerful feature descriptor [Wang et al., 2009] and showed satisfactory results even for extremely challenging film based datasets such as YouTube [Brendel and Todorovic, 2010, Ikizler-Cinbis and Sclaroff, 2010, Matikainen et al., 2010], Hollywood [Kläser et al., 2008, Satkin and Hebert, 2010, Wang et al., 2009] and UCF television [Wang et al., 2009, Weinland et al., 2010b].

2.3.3.1.2. Global Feature Descriptors

Global feature descriptors are extracted from a region of interest centred on a person performing an action (so called bounding box of the action). As a result, usually input videos are pre-processed to segment the regions of interest, which are then encoded as a whole by taking into account all available pixel information. The segmentation can be performed using detection/localisation algorithms, background subtraction or tracking. In current action recognition research, it is assumed that the segmentation is a solved problem. The final feature vector is given by the normalised region of interest in the raster scan fashion where the dimension of vector is equal to the number of pixels in the entire region.

2.3.3.1.2.1. Silhouettes

A binary silhouette (i.e. binary shape or contour) is a very simple image descriptor with a featureless interior of a person and uniform black background. Silhouette representation is insensitive to colour, texture, and contrast changes, but at the same time provide sufficient discriminative information for many action recognition frameworks [Chin et al., 2007, Wang and Suter, 2007a, Wang and Suter, 2007b, Lv and Nevatia, 2007, Tran and Sorokin, 2008, Wang and Suter, 2008, Jia and Yeung, 2008, Fang et al., 2009, Vezzani et al., 2010, Zhang and Gong, 2010]. Although such approaches used pure silhouettes, in most cases silhouettes are converted to more discriminative features such as the temporal motion templates (section 2.3.3.1.2.2) or the space-time local features (section 2.3.3.1.2.3) to take into account some temporal information. Alternatively, silhouettes are used for a volumetric reconstruction of data to form a 3D volume of human body [Weinland et al., 2007, Pehlivan and Duygulu, 2010].

2.3.3.1.2.2. Temporal Motion Templates

Motion History Image [Bobick and Davis, 2001] (MHI) is the simplest temporal template motion feature, which represents an action by encoding a history of silhouette deformation over time using decaying weights. Conceptually, the MHI image contains the past images within itself, in which the most recent image is the brightest. To overcome limitations of the MHI, [Meng and Pears, 2009] proposes the Motion History Histogram by additionally storing frequency information as the number of times motion is detected at every pixel, further categorised into the length of each motion. 2D temporal templates can easily be extended into 3D to form the Motion History Volumes by considering voxels instead of pixels [Weinland et al., 2006b]. A binary version of MHI is called the Motion Energy Image [Bobick and Davis, 2001].

2.3.3.1.2.3. Space-Time Local Features

A silhouette is surrounded by a simple, closed contour. A more advanced representation of the silhouette is inferred by assigning to every internal pixel a value which depends on a relative position of that point within the silhouette [Gorelick et al., 2006]. This relationship is determined by placing a set of particles at the point which are then moved in the random walk until the contour is hit. Thanks to the statistics of this random walk, the final value of internal pixel corresponds to the mean time required for the particle to hit the boundaries of the silhouette. This mean time measure is computed by a partial differential equation, called the Poisson equation, with the silhouette contours providing the boundary conditions.

The above spatial concept has been extended to the temporal domain [Gorelick et al., 2007]. A sequence of binary silhouettes can be considered as the space-time shape surrounded by a closed surface. As a result, particles can wander randomly in the spatio-temporal volume of data. This allows representing each silhouette by local space-time saliency and orientation features extracted from the solution of the Poisson equation of the corresponding volumetric surface, which implicitly takes into account the time domain. The final global descriptor for a given temporal range is obtained by calculating the weighted moments over these local features.

2.3.3.1.3. Summary of Feature Descriptors

Global descriptors are powerful and discriminative since they encode much of the information. However, they rely on robust detection/localisation, background segmentation or tracking to determine the region of interest and may not be appropriate in the presence of cluttered dynamic background and serious self occlusions. In contrast, local descriptors are more robust against noise, variations of viewpoint and partial occlusions, although the extraction of a sufficient amount of relevant interest points is challenging and computationally expensive. Moreover, local descriptors usually required a set of empirical parameters.

2.3.3.2.Action Descriptors

2.3.3.2.1. Hidden Markov Model

Hidden Markov Model [Rabiner, 1989] (HMM) is a statistical generative model in which the system being modelled is assumed to be a Markov process with an unobserved state, i.e. the state is not directly visible, but the output, dependent on the state, is visible. In action recognition, these hidden states correspond to different phases in an action. HMM learns state transition probabilities that model the temporal extent of action and observation probability density distributions that model the observation process of hidden states. To keep the modelling of the joint distribution over representation and labels tractable, two independence statistical assumptions are introduced. First, the state transitions are conditioned only on the previous state, not on the state history. This is the Markov assumption. Secondly, observations are conditioned only on the current state, so subsequent observations are considered to be independent in time. Training of the HMM is done efficiently using the Baum-Welch algorithm [Baum et al., 1970] (generalised case of the Expectation-Maximisation algorithm).

For instance, [Feng and Perona, 2002] considers key poses as states, and sequences of movelet codewords to model dynamics between them using an HMM framework. In similar manner, a HMM is used to model temporal evolution of silhouettes [Vezzani et al., 2010] or dynamic textures patterns [Kellokumpu et al., 2008]. Another interesting application is proposed by [Martinez-Contreras et al., 2009], where HMM tracks the Self Organizing Map [Kohonen, 1982] behaviour on the temporal sequences of MHI. In multi view settings, [Ahmad and Lee, 2008] represents an action with a set of multidimensional HMMs for multiple views using combined features of optic and shape flow in the spatial-temporal action boundary. Alternatively, [Weinland et al., 2007] proposes the exemplar-based HMM to model two independent random processes: one for the orientation of a subject relative to a camera, and the other for a most discriminative view independent poses taken by a performer during the various stages of an action.

2.3.3.2.2. Conditional Random Fields

Conditional Random Field [Lafferty et al., 2001] (CRF) is a generalisation of the HMM that allows observation and possible transitions to be arbitrary functions. Moreover, it is discriminative and can use multiple overlapping features. The model predicts the conditional probability of the states given multiple observations on different time scales. In contrast to the generative HMM, the CRF is trained to discriminate between action classes rather than learning to model each class individually. As a consequence, it avoids the independence assumption and can represent rich relationships among observations and long range dependencies.

[Sminchisescu et al., 2006] uses the simple linear chain CRF, where the state dependency is a first-order, to recognise human motion and to show superiority of the method in comparison to the HMM. Another application of the CRF is presented in [Zhang and Gong, 2010], where the modified hidden CRF is used to model an action and a global optimal solution is guaranteed after the HMM pathing stage. In a multi view scenario, [Natarajan and Nevatia, 2008] introduces the two layer graph model of an action, where nodes in the top level correspond to events in each viewpoint and on the lower layer CRFs are used to encode the action and the viewpoint-specific pose observation. Finally, [Wang and Suter, 2007b] introduces the factorial CRF for an action recognition.

2.3.3.2.3. Bag of Words

The Bag of Words [Schuldt et al., 2004, Dollar et al., 2005, Niebles et al., 2008, Junejo et al., 2008, Kaâniche and Brémond, 2009, Kaâniche and Brémond, 2010, Reddy et al., 2010, Brendel and Todorovic, 2010] (BoW) is a simple and powerful approach for modelling an action as a large visual vocabulary (dictionary, codebook) of discriminative code words. This visual dictionary is formed by the vector quantization of local feature descriptors extracted from images using for instance the k-means algorithm [Kanungo et al., 2002]. Code words are then defined as the centres of the learnt clusters after pruning out the clusters with a too small number of members. Then each local descriptor is assigned to the closest code word. A sequence of images is summarised by the distribution of code words from the fixed codebook by computing a histogram of code word occurrences based on the assignment of local descriptors. Action classification is performed by constructing a feature vector for video based on the defined dictionary to relate "new" descriptors in query images to descriptors previously seen in training.

Since BoW models discard the spatio-temporal layout of the local features which may be almost as important as the features themselves, the main line of research tries to reintroduce this information back into the BoW model. For instance, Laptev et al. [Laptev et al., 2008] employs spatio-temporal grids to extend the BoW into the spatio-temporal domain. A conceptually different approach is to explore the spatio-temporal correlation between code words using the spatial correlogram and spatio-temporal pyramid matching [Liu and Shah, 2008] or relative location probabilities [Matikainen et al., 2010]. Alternatively, two codebooks are generated according to an appearance and a geometric similarity of spatio-temporally related pairs of cuboids [Ta et al., 2010b]. As a result, each image is represented by two histograms of visual words which are combined into a single feature vector. Similarly, [Liu et al., 2008] also defines two vocabularies (i.e.

spatio-temporal cuboids and spin images), however here these features are fused in the weighted manner using the Fiedler embedding of a graph. Eventually, even a hierarchy of vocabularies is constructed using neighbourhoods of spatio-temporal features, where each code word encodes the interest point and a loose configuration of neighbours to capture space-time relationships between words at successively broader scales [Kovashka and Grauman, 2010].

Another issue with BoW is that the k-means algorithm only considers appearance similarity; therefore visual words are not necessarily semantically meaningful. To address this problem the feature space is clustered using Information Bottleneck [Liu and Shah, 2008] or KL-divergence [Liu et al., 2009] to obtain compact yet discriminative semantic vocabularies.

2.3.3.2.4. Dimensionality Reduction

Action video sequences are very high dimensional because of the human motion complexity. However, different instances of the given action reside only in a part of the entire feature space. This subspace can be considered as a nonlinear manifold embedded in a space of image frames. As a result, the discriminative and low dimensional manifold of the action can be discovered by a dimensionality reduction process.

A naïve approach is to employ the linear PCA to discover a low dimensional representation of filtered images [Masoud and Papanikolopoulos, 2003] or HOG descriptors [Lu and Little, 2006]. Alternatively, the locality preserving projection (LPP) is employed for producing low dimensional space of actions [Wang and Suter, 2007a, Wang and Suter, 2008, Fang et al., 2009] (LPP is a linear approximation of the nonlinear Laplacian Eigenmap [He and Niyogi, 2004a]). Since human motion is highly nonlinear, a nonlinear action manifold is obtained by applying the LLE on silhouettes [Chin et al., 2007], Isomap on the implicit function distance representations [Blackburn and Ribeiro, 2007], or Isomap on the view invariant R-transform descriptors [Richard and Kyle, 2009]. Another approach for modelling nonlinearity of an action is to use the Grassmann and Stiefel manifold embeddings [Turaga et al., 2008b]. All these manifolds are learned in an unsupervised manner, which does not guarantee good discrimination between related action classes. To address this issue, [Jia and Yeung, 2008] proposes a novel dimensionality reduction method, called Local Spatio-Temporal Discriminant Embedding, which is tailored to the human action recognition task. In principle, LSTDE projects data points of the same class close in the manifold and those of different classes far away, while temporal relations are modelled in subspaces of the manifold.

2.3.3.2.5. Summary of Action Descriptors

HMM and in particular CRF are excellent in modelling the temporal development of an action and allow making a probabilistic decision in the classification task. However, the process of model learning is challenging because of the curse of dimensionality associated with the space of features. In contrast, the learning process of BoW is extremely simple, but, at the same time, the obtained action model proves to be very discriminative and efficient. The main drawback of the BoW model is that it is not view and scale invariant; moreover it is a black box with neither rigorous spatial nor temporal structural information about action. On the other hand, not only dimensionality reduction models are easy to learn, but also they can simultaneously extract conceptually meaningful motion patterns from actions. As a consequence, these models are more intuitive and understandable for a user and, therefore, easy to analyse and process. However, similarly to BoW, the temporal aspect of the action is generally not taken into account during the dimensionality reduction process.

2.3.3.3. Classifiers

The action models are used to 'train' a classifier which performs the final annotation of a new action. We will describe briefly three of the most popular approaches.

2.3.3.3.1. Nearest Neighbour Classification

The k-Nearest neighbour (NN) classifier uses some distance metric to assess similarity between the descriptor of an observed sequence and available training descriptors. The most common label among the k closest training sequences is chosen as the classification decision. In order to always obtain a majority vote, k is usually an odd number to prevent tie cases. This classifier is common for action models generated by a dimensionality reduction process [Masoud and Papanikolopoulos, 2003, Chin et al., 2007, Wang and Suter, 2007a, Blackburn and Ribeiro, 2007, Wang and Suter, 2008, Turaga et al., 2008b, Fang et al., 2009]. Usually a new instance of action is projected into the action manifold and similarity between the projection and the learned manifold is calculated. The NN classifier is also used in combination with the BoW by simply calculating a distance between an input descriptor and available code words in the dictionary [Dollar et al., 2005, Liu et al., 2008, Kaâniche and Brémond, 2009, Kaâniche and Brémond, 2010, Brendel and Todorovic, 2010]. Other approaches which exploit this metric include [Bobick and Davis, 2001, Weinland et al., 2006b, Gorelick et al., 2007, Zhao and Elgammal, 2008, Tran and Sorokin, 2008, Pehlivan and Duygulu, 2010].

2.3.3.3.2. Probabilistic Classification

Probabilistic classification matches an observed sequence to the trained model (for instance HMM or CRF) that maximises the observation probability. The probability of observing the given sequence is computed by the maximum a posteriori estimation [Weinland et al., 2007, Natarajan and Nevatia, 2008, Martinez-Contreras et al., 2009, Zhang and Gong, 2010, Vezzani et al., 2010] or using the efficient Viterbi algorithm [Feng and Perona, 2002, Sminchisescu et al., 2006, Lv and Nevatia, 2007, Kellokumpu et al., 2008, Ahmad and Lee, 2008].

2.3.3.3.3. Support Vector Machine

Support Vector Machine (SVM) is a discriminative model which focuses on separating two [Boser et al., 1992, Cortes and Vapnik, 1995] or more classes [Hsu and Lin, 2002] using decision boundaries, rather than modelling them. It constructs the hyper plane or set of hyper planes in a high or infinite dimensional space to optimally separate data. Intuitively, good separation is achieved by the hyper plane that has the largest distance to the nearest training pattern of any class, since, in general, the larger the margin between classes the lower the generalisation error of the classifier. The feature vectors that constrain the width of the margin are called *support vectors*. Although SVM is applied in the original finite high dimensional space of features, it often happens that in that space the sets of features cannot be linearly separated. For this reason, SVM uses a kernel function to transform the data into a higher (and potentially infinite) dimensional space to make the linear separation possible. Many kernel mapping functions can be used; some of them are presented in section 2.2.2.2.2.1. For more details see [Burges, 1998].

SVM has been usually trained on BoW and has proved to be a very powerful classifier for action recognition [Laptev et al., 2008, Kläser et al., 2008, Junejo et al., 2008, Wang et al., 2009, Meng and Pears, 2009, Liu and Shah, 2008, Schindler and van Gool, 2008, Yeffet and Wolf, 2009, Ta et al., 2010b, Weinland et al., 2010b, Ikizler-Cinbis and Sclaroff, 2010, Satkin and Hebert, 2010, Matikainen et al., 2010].

2.3.3.4. Datasets and Metrics

Since action recognition is a very dynamic area of research in the computer vision community, many datasets have been proposed to evaluate frameworks in different settings and scenarios. The most widely used datasets include: Weizmann [Gorelick et al., 2007], KTH [Schuldt et al., 2004], IXMAS [Weinland et al., 2006b], MuHAVi [MuHAVi, 2010], ViHASi [Ragheb et al., 2008], Hollywood [Laptev et al., 2008], YouTube [Liu et al., 2009], UCF television [Rodriguez et al., 2008] and UT-Iteraction [Ryoo and Aggarwal, 2009].

In this work, our contribution is validated on two well established datasets which are considered as the baseline for all action recognition frameworks. Both datasets will be described in the next sections followed by the description of the standard evaluation protocols used by the action recognition community.

2.3.3.4.1. Weizmann Dataset

The human action dataset recorded by [Gorelick et al., 2007] consists of 9 different subjects repeating several times 10 actions in outdoor environment (walk, run, jump, gallop sideways, bend, one-hand wave, two-hand wave, and jump in place, jumping jack and skip). The backgrounds are static and foreground silhouettes are included in the dataset. The dataset is useful for the evaluation of view dependent frameworks, because the viewpoint of provided videos is static. Example frames of actions and subjects are presented in Figure 2.28.

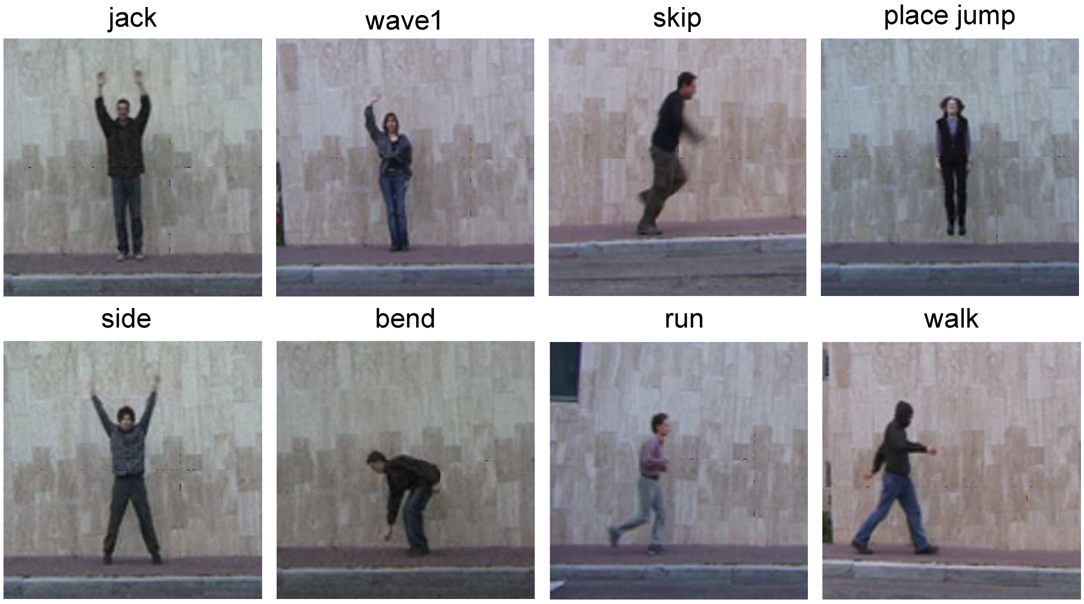


Figure 2.28. Examples of different actions and actors in the Weizmann dataset.

2.3.3.4.2. IXMAS Dataset

IXMAS dataset is introduced by [Weinland et al., 2006b] and contains videos of actions captured from five viewpoints. A total of 12 persons perform 3 times each of 13 actions (check watch, cross arms, scratch head, sit down, get up, turn around, walk, wave, punch, kick, point, pick up, throw). In this dataset, actors' positions and orientations are arbitrary since no specific instruction was given during acquisition. As a consequence, the action viewpoints are random and unknown. The camera views are fixed, with a static background and illumination settings. Silhouettes and reconstructed 3D visual hulls are provided by the dataset. Example frames of actions and subjects are presented in Figure 2.29.

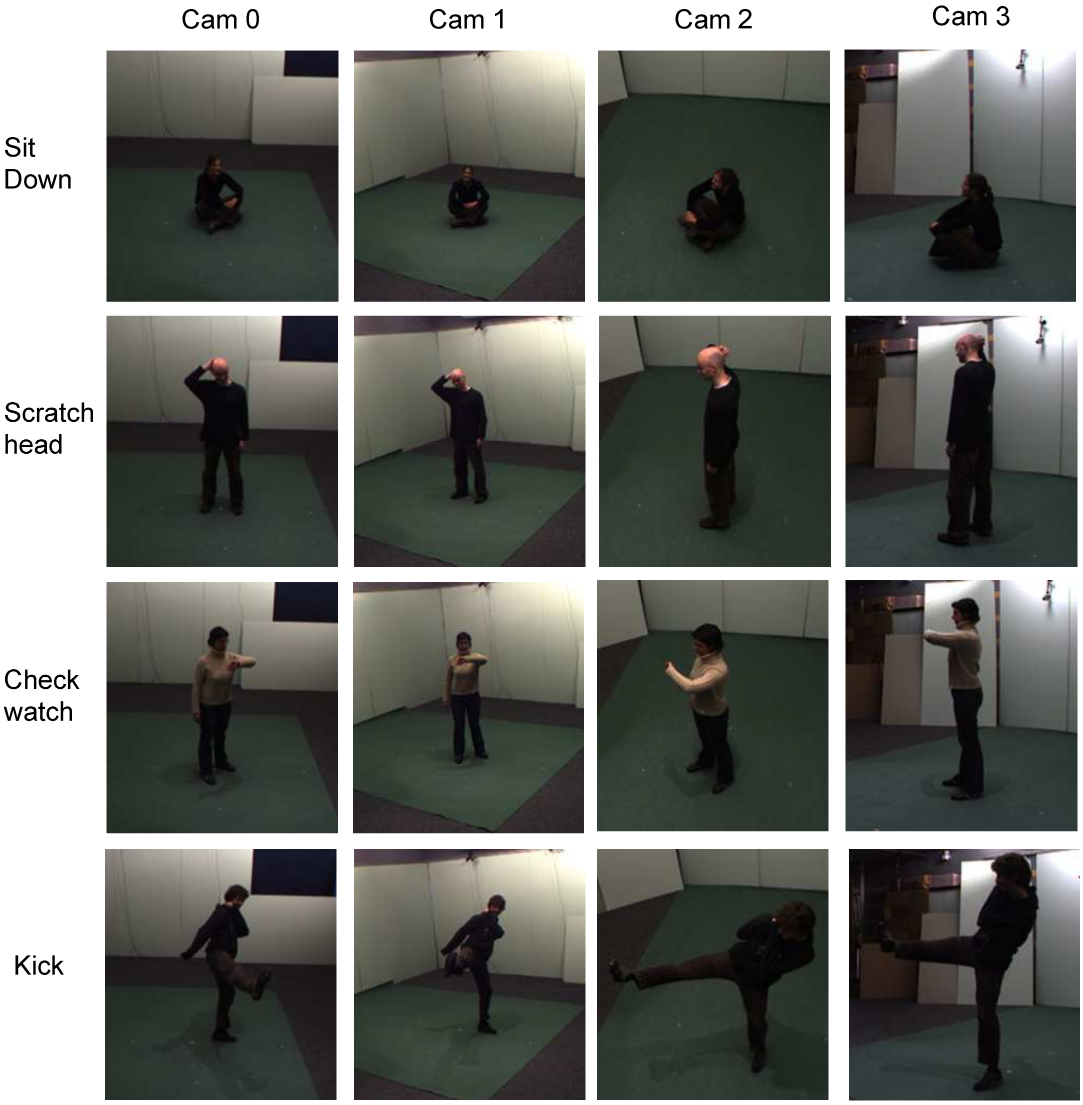


Figure 2.29. Examples of different actions, actors and views in the IXMAS dataset.

2.3.3.4.3. Evaluation Protocols and Metric

Different action recognition frameworks often use different experimental settings; therefore a direct comparison is not always straightforward. Popular evaluation schemas are divided into three groups:

- The holdout validation – a dataset is split into two sets of videos and one of them is used for training, while another one for testing. This schema is used in [Schuldt et al., 2004, Lv and Nevatia, 2007, Laptev et al., 2008, Meng and Pears, 2009].

- The K -fold cross validation – a dataset is partitioned into K groups of subject's dependent videos. For each of the K experiments, $K-1$ subjects are used for training and the remaining one for testing. A final error is estimated by the average error rate over all experiments. This schema is used in [Liu and Shah, 2008, Liu et al., 2008, Zhang and Gong, 2010, Matikainen et al., 2010].
- The Leave-one-out cross validation – this schema is a special case of the K-fold cross validation, where K is chosen as the total number of subjects. For each of K experiments, only action instances of one actor are used for testing and all remaining for training. A final error is estimated by the average error rate over all experiments. This schema is used in [Weinland et al., 2007, Gorelick et al., 2007, Wang and Suter, 2007a, Chin et al., 2007] [Kläser et al., 2008, Junejo et al., 2008, Tran and Sorokin, 2008, Yan et al., 2008, Turaga et al., 2008b, Kellokumpu et al., 2008] [Roth et al., 2009, Richard and Kyle, 2009] [Kaâniche and Brémond, 2010, Weinland et al., 2010b, Ta et al., 2010a, Ta et al., 2010b, Vezzani et al., 2010, Brendel and Todorovic, 2010, Kovashka and Grauman, 2010, Pehlivan and Duygulu, 2010].

Action recognition performance is measured by the average number of correctly classified actions in dataset over all subjects and views. In addition, detailed accuracy results are very often presented in a confusion matrix with respect to available actions.

2.4. Summary

In this chapter we outlined the background for the remainder of this thesis. We began with the theoretical foundations of dimensionality reduction. Then we reviewed algorithms of the two main strands of the current research and highlighted some of the strength and weaknesses of each group of dimensionality reduction methods. Afterwards, we discussed the main developments in the literature of human motion analysis with a special attention to the usage of dimensionality reduction transformations. This is essential to establish the general background for the evaluation of our contributions.

Although enormous effort has been already undertaken by the research community to design powerful dimensionality reduction tools to tackle a wide range of real-life problems, we have identified a few fundamental research gaps which we address in this dissertation in the following chapters.

3. Automatic Configuration of Spectral Dimensionality Methods

3.1. Introduction

Many real datasets are highly nonlinear and high dimensional. Since spectral methods can handle very large datasets with a reasonable computational cost, they have proved very popular (see section 2.2.2.2.2.2). However, the absence of explicit mapping between low and high dimensional spaces as well as manual tuning of parameters limits their usefulness. In this chapter we tackle these fundamental problems by proposing an advanced framework for the automatic configuration of spectral dimensionality reduction methods [Lewandowski et al., 2009, Lewandowski et al., 2010a]. This is achieved by introducing, first, the mutual information measure to assess the quality of discovered embedded spaces. Secondly, unsupervised graph-based Radial Basis Function network (G-RBFN) is designated for mapping between spaces where the learning process is derived from graph theory and based on Markov cluster algorithm. Exhaustive experiments on synthetic and real datasets demonstrate the effectiveness of the proposed methodology in a variety of applications, i.e. classification of hand written digits, face recognition and human pose recovery.

The rest of the chapter is organised as follows. Section 3.2 investigates advantages and disadvantages of embedded based family of dimensionality reduction methods and some competitive proposals which address their limitations. The proposed framework is described in detail in section 3.3. Its evaluation is given in section 3.4. Finally, a summary can be found in section 3.5.

3.2. Related Work

Spectral or embedding-based approaches model the structure of data by preserving some geometrical property of the underlying manifold. While the Isomap [Tenenbaum et al., 2000] method attempts to maintain global properties, LE [Belkin and Niyogi, 2002] and LLE [Roweis and Saul, 2000] aim at preserving local geometry which implicitly tends to keep the global layout of the data manifold. Since the brief description of these techniques is provided in section 2.2.2.2.2.2, here we focus on their limitations.

The main shortcomings of spectral methods are that first the quality of embedded space is extremely sensitive to the required free parameters and, secondly, they do not provide any mapping function between the low and high dimensional spaces. Despite research being conducted to improve these methods [De Silva and Tenenbaum, 2003, De Ridder et al., 2003, Donoho and Grimes, 2003, He and Niyogi, 2004b, Yang, 2003, He et al., 2004, He et al., 2005, Choi and Choi, 2007, Zhang and Wang, 2007, Kokiopoulou and Saad, 2007, Zheng et al., 2008, Yin et al., 2008b, Goldberg and Ritov, 2009, Wang and Li, 2009], they still rely on the emperical set of a few values, i.e. neighbourhood size, dimensionality of embedded space and mapping function parameters.

3.2.1. Selection of Free Parameters

All spectral approaches have two essential free parameters (Figure 2.12):

- the dimensionality of embedded space, $\mathbf{d}$,
- the neighbourhood size, $\mathbf{K}$,

which have to be specified apriori in order to perform dimensionality reduction.

3.2.1.1. Dimensionality of Embedded Space

The dimensionality $\mathbf{d}$ is used to choose the appropriate number of eigenvalues and corresponding eigenvectors, which are solutions of the eigenvalue problem. The eigenvectors form the basis of the low dimensional space. The optimal value of $\mathbf{d}$ should satisfy the 'principle of parsimony' [Bell and Wang, 2000], thus, it should be set to the smallest possible number of dimensions which allows maximal preservation of the original information. Such optimal dimensionality is defined as the intrinsic dimension of the high dimensional data. More formally, a dataset $X \subset \mathbb{R}^{\mathbf{D}}$ is said to have intrinsic dimensionality (ID) equal to $\mathbf{d}$ if its elements lie entirely within a d-dimensional subspace of $\mathbb{R}^{\mathbf{D}}$ (where $\mathbf{d} << \mathbf{D}$) [Fukunaga, 1982]. The estimation of the intrinsic dimensionality is a crucial problem, because knowing it the possibility of over- or under-fitting would be eliminated. In particular, if the number of dimensions is too low, important data features may be collapsed onto the same dimension. Therefore, the determination of $\mathbf{d}$ is a very well studied problem in machine learning and many approaches have been proposed (see [Camastra, 2003] for a detailed review). They include:

- projection methods, which use a low dimensional embedding to estimate ID:
  - eigenvalue-based estimator [Fukunaga and Olsen, 1971] (EE),
  - PCA estimator with cover sets [Fan et al., 2010],
- geometric approaches, which investigate the intrinsic geometric structure of data in order to estimate ID:
  - packing numbers [Kegl, 2003],
  - analysis of a geodesic minimum spanning tree [Costa and Hero, 2004],
  - fractal-based method [Camastra and Vinciarelli, 2002],
  - neighborhood convex hull method [Li et al., 2007b],

- probabilistic methods, which make a distribution assumption on data to build the ID estimator:
  - maximum likelihood estimation [Levina and Bickel, 2005, MacKay and Ghahramani, 2005],
  - incising ball algorithm [Fan et al., 2009].

However, none of them has achieved consensus as the most accurate method. Projection methods [Fukunaga and Olsen, 1971, Fan et al., 2010] are based on a heuristic basis [Camastra, 2003], whereas fractal-based [Camastra and Vinciarelli, 2002] and packing numbers [Kegl, 2003] methods are designed for low-dimensional datasets since their complexity grows exponentially with the dimension [Camastra, 2003]. In turn, the graph-based methods [Costa and Hero, 2004, Li et al., 2007b] are sensitive to a required neighbourhood size parameter and tend to overestimate the ID as the neighbourhood size increases [Fan et al., 2010]. Finally, the maximum likelihood estimation assumes that a surrounding of any data points can be correctly approximated by a uniform probability distribution function [Levina and Bickel, 2005]. However, in very high dimensional spaces with a relatively small number of observations due to the dimensionality curse (section 2.2), the assumption under which the method relies is not fulfilled [Ramos et al., 2007]. In a similar vein, [Fan et al., 2009] is based on a uniformity assumption. As a consequence, in practice, the choice of ID estimation procedure depends very often on a particular application and the nature of exploited datasets.

3.2.1.2. Neighbourhood Size

The selection of the optimal neighbourhood size is also a challenging problem. If it is too small, global feature information is lost since the manifold may be split into unconnected pieces. If it is too large, the LE and LLE assumption that a data point and its neighbours are locally linear is violated. In the case of Isomap, a large value of $\mathbf{K}$ introduces errors in geodesic distances. The main lines of research to discover the optimal value of $\mathbf{K}$ are following:

- Adaptive selection of local neighbourhood size for each data point.
- Assessing directly the quality of embedded spaces by a quantitative measure in order to infer the global optimal value of the neighbourhood size.
- Optimisation of already constructed neighbourhoods.

3.2.1.2.1. Adaptive Neighbourhood Selection

The main idea behind these algorithms is to adaptively estimate the neighbourhood size of each data point separately by interatively adding/removing points to a considered neighbourhood until a defined condition is violated.

For instance, [Wang et al., 2005] defines neighbourhood contraction and expansion procedures based on the estimation of local tangent space. However, the neighbourhood size parameter is indirectly replaced by several other user-specified parameters. [Mekuz and Tsotsos, 2006] overcome this problem by proposing a parameterless estimation procedure where a neighbourhood incrementally grows as long as candidates agree with a locally computed linear tangent orientation based on the estimated intrinsic dimensionality. Alternatively, instead of using the Euclidean distance for determination of neighbourhood candidates, [Wei et al., 2008] exploits, first, the manifold ranking method [Zhou et al., 2003] to choose the best candidate and, then, constructs a suitable local tangent space. In contrast, [Zhan et al., 2009a, Zhan et al., 2009b] proposes an algorithm that expands neighbourhood for each data point by measuring local linearity of its neighbourhood patch on a manifold using PCA under the assumption that neighbourhood should be as large as possible.

3.2.1.2.2. Quantitative Evaluation of Embeded Spaces

An alternative approach is to use a single global neighbourhood size for all points. This neighbourhood size parameter can be estimated automatically by assessing directly the quality of embedded spaces using a quantitative measure. The neighbourhood size, which leads to the best score for the corresponding embedded space, is chosen. Many measures have already been proposed, such as Residual Variance [Kouropteva et al., 2002, Samko et al., 2006], Spearman Rho [Samko et al., 2006, Karbauskait et al., 2007] and Procrustes Analysis [Goldberg and Ritov, 2009].

3.2.1.2.2.1. Residual Variance

The residual variance [Kouropteva et al., 2002, Samko et al., 2006] expresses how well the distance information is preserved between two corresponding sets of variables X and Y consisting of $\mathbf{N}$ examples each, i.e. it reflects the degree of linear relationship between these variables. The metric is expressed by the following formula and a value of 0 implies that there is no linear relationship:

$$\rho = \arg\min(1 - r_{XY}^2) \tag{3.1}$$

where r_{XY} is the standard linear Pearson's product-moment correlation between high and low dimensional spaces:

$$r_{XY} = \frac{1}{\mathbf{N}} \sum_{i=1}^{\mathbf{N}} \frac{(x_i - \mu_X)}{\sigma_X} \frac{(y_i - \mu_Y)}{\sigma_Y} \tag{3.2}$$

Here μ and σ denote respectively the mean and standard deviation of a dataset.

3.2.1.2.2.2. Spearman’s Rho

Spearman's rho [Samko et al., 2006, Karbauskait et al., 2007] measures the accuracy of the low-dimensional manifold in retaining the order of pair wise distances of data points of the high-dimensional. A value of 1 or -1 corresponds to the highest correlation, whereas correlation is equal to zero implies no association between the two variables. In principle, SR is simply a special case of the Pearson product-moment coefficient in which variables x_i and y_i are converted to rankings $r_x(i)$ and $r_y(i)$ before calculating the coefficient. The Spearman rho is expressed by the following equation:

$$\rho = 1 - \frac{6\sum_{i=1}^{\mathbf{N}}(r_y(i) - r_x(i))^2}{\mathbf{N}^3 - \mathbf{N}} \tag{3.3}$$

3.2.1.2.2.3. Procrustes Analysis

The procrustes analysis measure [Goldberg and Ritov, 2009] reflects the matching of two sets of variables X and Y in terms of distances. It determines how well a linear transformation (i.e. translation, reflection, orthogonal rotation, and/or scaling) of the points in one space conforms to the points in the other space. The smaller the value of the procrustes measure, the better the correlation between spaces:

$$\rho = trace[(X-(AY+b))(X-(AY+b))^T] \tag{3.4}$$

where A is the Procrustes rotation matrix which is computed explicitly by the singular value decomposition of $X^T HY$, where H is the centering matrix [Sibson, 1979]. The Procrustes translation vector b is given by the difference between means of X and Y [Goldberg and Ritov, 2009].

3.2.1.2.3. Neighbourhood Optimisation

This class of methods assumes that neighourhoods have already been found using any of the already discussed techniques. These neighbourhoods are then optimised to better represent the high dimensional data. For example, an approach based on path algebra of graph is investigated in [Wen et al., 2007], where better neighborhoods were obtained for Isomap by considering an implicit correlation among data points. Alternatively, [Wen et al., 2008] applies locally estimated geodesic distances to refine the neighborhoods.

3.2.1.2.4. Summary of Neighbourhood Selection Procedures

Although methods which adaptively select neighbourhood, show promising results [Wang et al., 2005, Mekuz and Tsotsos, 2006, Wei et al., 2008, Zhan et al., 2009a, Zhan et al., 2009b], their main drawback is very strong dependency on local constraints. Real datasets are high dimensional and nonlinear but, at the same time, a sampling density is usually poor because of the dimensionality curse (see section 2.2). As a consequence, locally linear patches in such space as well as the estimated neighbourhoods tend to be very small and may produce disjoint graphs in different areas of a manifold. In such case, the global topology of the data manifold is completely lost during a dimensionality reduction process, since it has to be performed independently on each of the graphs. In addition, the assembled graph may provide less constraints for an optimisation process, since it may be not well connected.

To overcome these limitations, a single global neighbourhood size for all points can be estimated, which reduces significantly the probability of generating disjoint graphs because of weak local costraints. $\mathbf{K}$ nearest neigbhours can always be determined, even though they lie only on an approximately linear patch of a manifold. In such case, spectral methods are still capable to discover a meaningful low dimensional representation because the fully connected graph can be assembled.

Neighbourhood optimisation techniques assume that initial neighourhood sizes have already been provided for each data point, thus they can be considered as the post processing optimisation step of any of the above methodologies.

3.2.2. Design of Mapping Function

An inherent limitation of spectral dimensionality reduction approaches is that they do not provide any explicit mapping function between low and high dimensional spaces. Such function is essential to allow a projection of data between spaces and an interpolation of the low dimensional representation to unseen examples. Among the different strategies that have been applied to address this issue (see section 2.2.2.4 for overview), Radial Basis Function network [Poggio and Girosi, 1990] (RBFN) tackles this problem quite satisfactory by approximating the optimal mapping function [Elgammal and Lee, 2004a, He et al., 2004, Shi et al., 2005, Elgammal and Lee, 2007, Blackburn and Ribeiro, 2007, Ohbuchi et al., 2008, Lewandowski et al., 2009]. The entire process of learning RBFN has been summarised in section 2.2.2.4.4. However, peformance of the obtained RBFN relies on the careful selection of a few parameters which are usually chosen empirically.

The RBFN structure is based on centres $C = \{c_i \mid i = 1..\mathbf{Z}, \mathbf{Z} \ll \mathbf{N}\}$ which summarise training data points in order to provide generalisation properties to the network. The performance and generalisation potential of RBFN critically depend upon the choice of these centres [Chen et al., 1991]. K-means clustering [Kanungo et al., 2002] and rival penalized competitive learning [Xu et al., 1993] are currently the most popular and well studied methods which address this task.

3.2.2.1. K-means Clustering

The most common form of the K-means clustering algorithm [Kanungo et al., 2002] (KMC) uses an iterative refinement heuristic known as the Lloyd's algorithm [Lloyd, 1982]. It starts by random initialisation of centres and then two steps alternate points' assignment and centres relocation. In the first step, all points are assigned to the closest centre to form clusters. Afterwards, means of obtained clusters are computed and become new centres. These two steps are repeated until convergence of the following objective function:

$$\varepsilon = \sum_{j=1}^{\mathbf{Z}} \sum_{i=1}^{\mathbf{N}} \left\| x_i^j - c_j \right\|^2 \tag{3.5}$$

The above equation corresponds to a minimisation of total intra-cluster variance in dataset. In terms of performance the algorithm is not guaranteed to return a global optimum. The quality of the final solution depends largely on the initial set of clusters. Moreover, a key drawback of the KMC algorithm is that it requires prior knowledge of the correct number of centres.

3.2.2.2. Rival Penalized Competitive Learning

The rival penalized competitive learning [Xu et al., 1993] (RPCL) algorithm is capable of finding the optimal localisation of centres as well as their correct number $\mathbf{Z}$ in an automatic way. First, $\mathbf{Z'}$ centres are randomly initialised ($\mathbf{Z'} >> \mathbf{Z}$). Subsequently, in each iteration, the algorithm randomly selects a sample s from the training set and moves the closest centre (the so called competition winner c_W) towards the considered point s by a weighted distance $w1$. In the same step the second closest centre (or rival c_R) is pushed away from the sample s by a weighted distance $w2$ (where $w1 >> w2$). Learning rates, i.e. $w1$, $w2$ are monotonically decreased after each iteration. The entire procedure is repeated until it converges or reaches a given threshold. This mechanism allows automatic determination of the centres' positions by locating them at the core of data point clusters and gradually driving unrequired centres away from those clusters. The discussed procedure is illustrated in Figure 3.1.

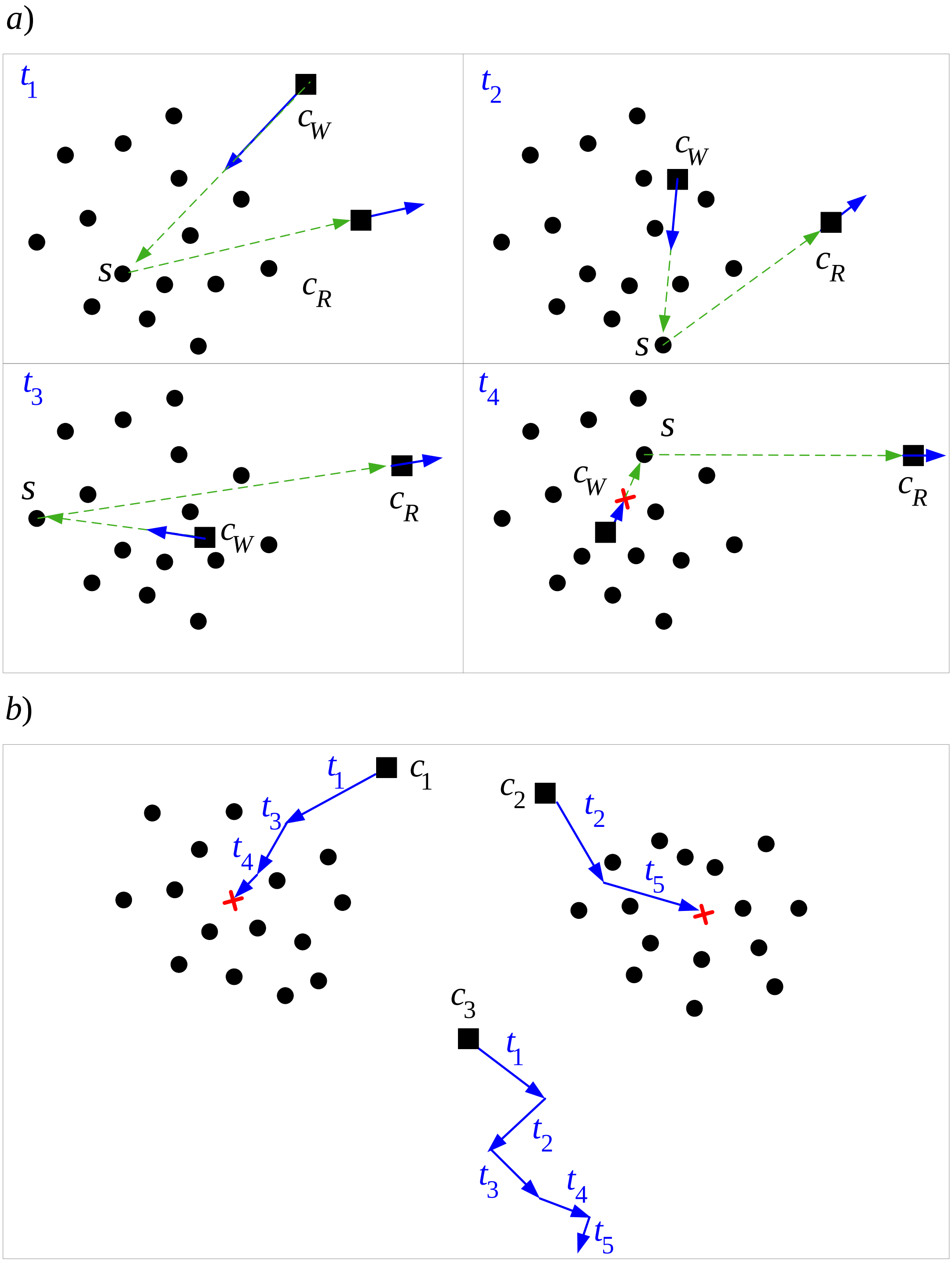


Figure 3.1. Rival Penalized Competitive Learning: a) the rival c_R is pushed away from the cluster that the winner c_W is approaching at each time step. b) The correct number of centres is determined by pushing away unnecessary centre such as c_3.

3.3. Proposed Methodology

We propose a general framework for the automatic configuration of spectral dimensionality reduction methods which contribute to the current state of the art by addressing two essential problems: the selection of the optimal neighbourhood size K and the inherent absence of mapping function between spaces. First, we propose to estimate the optimal neighbourhood size by assessing the quality of discovered embedding spaces using the mutual information (MI) measure [Cover and Thomas, 1991]. Secondly, we overcome the deficiency of mapping function by extending RBFN to design the optimal structure of the network in an unsupervised manner using spectral graphs which are constructed in the first step of the embedded based approaches. In principle, our framework can be applied to any spectral dimensionality reduction approach which shares the structure of the algorithm illustrated in Figure 2.12. This includes Isomap, LLE, LE and many of their extensions. In agreement with the previous research in the field (section 3.2.1.2), we assume that the intrinsic dimensionality d (ID) is known or it is estimated using any dimensionality estimation technique (section 3.2.1.1).

3.3.1. Mutual Information Measure

The selection of the optimal neighbourhood size K is still an open and challenging problem as discussed in section 3.2.1.2. Our proposed inference procedure follows the most promising line of research which focuses on estimating globally the neighbourhood size (section 3.2.1.2.2). Although many measures have already been proposed (section 3.2.1.2.2), experiments suggest that their accuracy depends not only on the choice of intrinsic dimensionality but also on the dataset nature. Consequently, they are not suitable when dealing with complex nonlinear high dimensional data of unknown nature [Lewandowski et al., 2009, Lewandowski et al., 2010a].

The optimal neighbourhood size $\mathbf{K}$ can be identified directly by assessing embedded space quality, by the following process: First, data are divided into training and testing sets. Then, for a given value of $\mathbf{K}$, dimensionality reduction is applied on the training set and a mapping function is built between the original and embedded spaces. Finally, test data are projected into the low dimensional space and an error metric is calculated. This process is repeated for a range of $\mathbf{K}$ values so that the optimal neighbourhood size is identified.

Since this process requires calculating computationally expensive mapping functions for all possible values of $\mathbf{K}$, quantitative metrics have been proposed to evaluate the quality of an embedded space without mapping. The standard procedure of optimal neighbourhood size estimation using a quantitative metric is summarised in the following pseudo-code (algorithm 1).

Algorithm 1. Estimation of optimal neighbourhood size

```
Input: high dimension dataset, maximum $\mathbf{K}$ (maxK), ID estimate $\mathbf{d}$
Output: optimal $\mathbf{K}$
Find minimum $\mathbf{K}$ (minK) which produces a fully connected graph
for each $\mathbf{K}$ in range < minK, maxK > do
    Reduce dimensionality of the dataset using a spectral method
    Use metric to assess the quality of the embedded space
end for
Select optimal $\mathbf{K}$ according to metric
```

In our framework, we adopt an advanced metric to assess the quality of spaces. This metric can deal with features without any linear relationship. We propose to use the mutual information (MI) measure [Cover and Thomas, 1991] which has proved to be able to discover even marginal dependency between two spaces of variables, since, in contrast to linear correlation coefficients, it is also sensitive to dependencies which do not manifest themselves in the covariance. MI is null if and only if the two random variables are strictly independent. The first idea would be to design a cost function directly in the spectral dimensionality reduction framework using MI; however since MI expresses relationship between two sets of variables rather than individual points, it is not an appropriate metric for that purpose. As a consequence, we propose to employ it in a post processing step to evaluate the quality of spaces.

The most straightforward and widespread approach for estimating MI is to partition the data and approximate MI by the following finite sum:

$$I(X,Y)=\sum_{i=1}^{\mathbf{N}}\sum_{j=1}^{\mathbf{N}} p(i,j)\log\frac{p(i,j)}{p_x(i)p_y(j)} \tag{3.6}$$

where $p(i,j)$ is the joint probability distribution function, and $p_x(i)$ and $p_y(j)$ are the marginal probability distribution functions of X and Y respectively. This formulation is equivalently expressed as [Cover and Thomas, 1991]:

$$I(X,Y)=H(X)+H(Y)-H(X,Y) \tag{3.7}$$

where $H(X)$ and $H(Y)$ are the marginal entropies and $H(X,Y)$ is the joint entropy of X and Y.

However, this standard approach can only be applied for $\mathbf{D}=\mathbf{d}=1$, because the estimation of entropy is based on data binning. Since, in our framework, we need to estimate MI measure for higher dimensional variables ($\mathbf{D}>1,\mathbf{d}\geq 1$), we calculate the entropy using K-nearest neighbour statistics as proposed in [Kraskov et al., 2004]. Assuming that some metric is defined on the spaces spanned by X and Y, all neighbours of a given data point w_i are ranked according to their distance to that point. As a consequence, the entropy $H(W)$ ($w\in\{x,y\}$) can be estimated by the average distance to the K-nearest neighbour, averaged over all w_i. This leads to the following equation [Kraskov et al., 2004]:

$$H(W) = \mathbf{N}^{-1}\sum_{i=1}^{\mathbf{N}}\left(\psi(n_w(i)+1) - \psi(\mathbf{N}) - \log c_{d_w} - \frac{d_w}{\mathbf{N}}\sum_{i=1}^{\mathbf{N}}\log\varepsilon(i)\right) \qquad (3.8)$$

Here, $n_w(i)$ denotes the number of points whose distance from w_i is strictly less then $\varepsilon(i)$, i.e. count($\|w_i - w_j\| < \varepsilon(i)$), where $\varepsilon(i)$ is a distance between w_i and its K*th* neighbour. In turn, $\psi(\cdot)$ is the digamma function [Kraskov et al., 2004], whereas d_w ($w \in \{x, y\}$) denotes the dimension of w and c_{d_w} is the volume of the d-dimensional unit ball. Similarly, the joint entropy of X and Y for a given $\mathbf{K}$ [Kraskov et al., 2004] is expressed by:

$$H(X,Y) = \psi(\mathbf{K}) - \psi(\mathbf{N}) - \log(c_{d_x} c_{d_y}) - \frac{d_x + d_y}{\mathbf{N}}\sum_{i=1}^{\mathbf{N}}\log\varepsilon(i) \qquad (3.9)$$

Combining equations (3.7), (3.8) and (3.9) results in the expression of multi-dimensional MI:

$$I(X,Y) = \psi(\mathbf{K}) + \psi(\mathbf{N}) - \mathbf{N}^{-1}\sum_{i=1}^{\mathbf{N}}\left(\psi(n_x(i)+1) + \psi(n_y(i)+1)\right) \qquad (3.10)$$

Although mutual information has never been used in this context, its multidimensional extension allows MI becoming an intuitive measure for analysing the mutual correlation between high and low dimensional spaces.

3.3.2. Graph-based Radial Basis Function Network

All spectral approaches suffer from the deficiency of lacking a mapping function. Very popular solution to this problem is to use RBFN based mapping [Elgammal and Lee, 2004a, He et al., 2004, Shi et al., 2005, Elgammal and Lee, 2007, Blackburn and Ribeiro, 2007, Ohbuchi et al., 2008]. However, this process relies on manual adjustment of RBFN structure according to data. In the case of standard KMC algorithm, it means that prior knowledge about the correct number of centres is required.

Our first attempt to automate the mapping learning process was to apply RPCL for training of RBFN in the context of manifold learning [Lewandowski et al., 2009]. However, RPCL, as KMC, depends on the initial random localisation of centres and relies on the Euclidean distance, which is not the most appropriate metric to model high dimensional relationships [Aggarwal et al., 2001]. In order to improve accuracy, we extend our idea of unsupervised mapping learning and propose to use the Markov cluster algorithm [Dongen, 2000] (MCL) to identify the suitable number and localisation of centres automatically by exploiting the adjacency graph constructed during spectral dimensionality reduction [Lewandowski et al., 2010a]. As a consequence, the novel graph-based radial basis function network is introduced (G-RBFN) which is tailored to spectral methods. As it will be demonstrated in the results section, the computational cost of the mapping learning process is greatly reduced and the obtained mapping exhibits better accuracy in comparison to standard approaches such as KMC and RPCL.

At the heart of the MCL algorithm [Dongen, 2000] lies the idea of simulating flow within a graph: flows are promoted where current is strong and demoted where current is weak. Flow simulation is achieved by transforming a graph into a Markov graph using the standard definition of a random walk on a graph. Then, a flow is defined by two simple algebraic operations, i.e. expansion and inflation, which are applied connectively on a stochastic (Markov) matrix in the iterative estimation, so that the flow becomes thicker in regions of higher current and thinner in regions of lower current. The process converges quadratically in the neighbourhood of so called doubly idempotent matrices (idempotent under both expansion and inflation) [Dongen, 2000].

According to this paradigm, if natural groups are present in the spectral graph obtained in the first step of dimensionality reduction, then current across borders between different groups will wither away. As a result, a fully connected graph is divided into few sub graphs (Figure 3.2), thus revealing the optimal number $\mathbf{Z}$ as well as coordinates of clusters $C = \{c_i \mid i = 1..\mathbf{Z}\}$. Application of this procedure enables the discovery of more representative clusters of high dimensional data and subsequently customises RBFN structure to dataset in an automatic and efficient manner. For instance, clusters obtained using KMC or RPCL mixes points from different branches of the manifold (Figure 3.3a), thus a global structure is lost, whereas MCL clusters follow appropriately a high dimensional curvature of the dataset (Figure 3.3b),

Once the clusters are determined, the learning process of RBFN follows the standard procedure described in section 2.2.2.4.4.

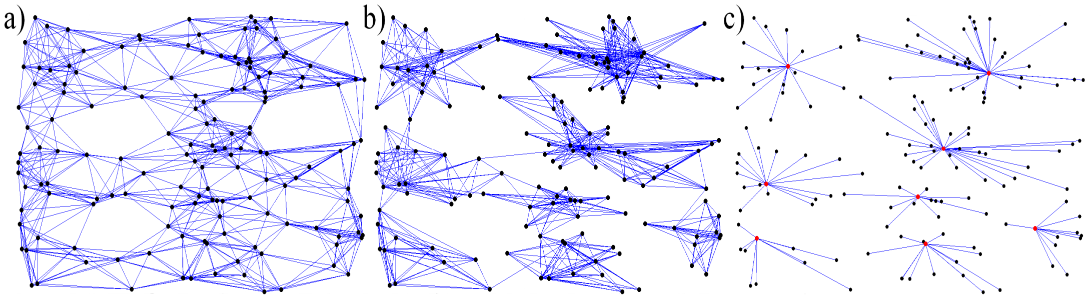


Figure 3.2. 2D representation of successive iterations of flow simulation using the MCL process for discovery of the localisation and the number of centres in RBFN.

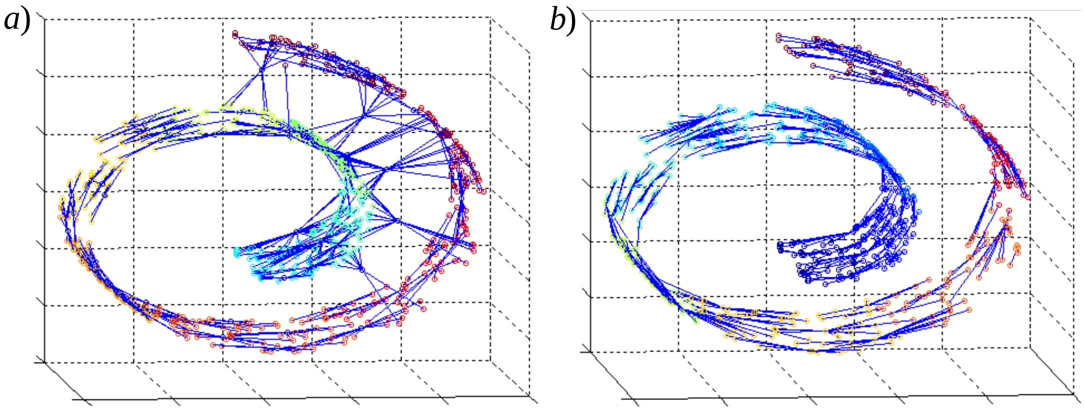


Figure 3.3. Customisation of RBFN structure for swissroll using a) KMC/RPCL and b) MCL.

3.4. Evaluation

The proposed framework was validated with both artificial and real datasets. Standard datasets were selected to extensively evaluate the performance and robustness of the proposed methodology in different scenarios.

In this section, first, all datasets, which are used in the evaluation process, are introduced in section 3.4.1. Then, a setup of experiments is explained in section 3.4.2.1 followed by a definition of performed experiments in section 3.4.2.2. Subsequently, sections 3.4.3, 3.4.4 and 3.4.6 provide results of experiments, whereas section 3.4.5 presents a practical application of the proposed methodology, i.e. 3D human pose recovery. Finally, the broad discussion about obtained results is provided in section 3.4.7.

3.4.1. Datasets

Figure 3.4 and Figure 3.5 illustrate the datasets used for evaluation.

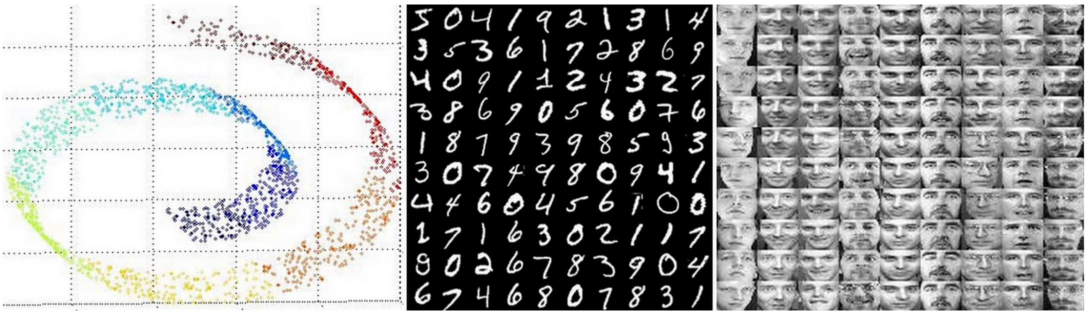


Figure 3.4. Datasets used in the experiments: from left to right, swissroll manifold, handwritten digits (the MNIST dataset) and face images (the ORL dataset).

Figure 3.5. Variety of actors from HumanEva dataset which were used in the evaluation process. From left to right: S1, S2, S3.

The swissroll dataset is a synthetic and nonlinear example of a two dimensional flat submanifold which lies in a three-dimensional space. The ideal low dimensional representation is a two dimensional rectungular structure, which is expected to be revealed by unrolling the three dimensional swissroll shape. This dataset exhibits significant disagreement between geodesic and Euclidean distances (Figure 3.4a). Two thousand points were randomly sampled from the manifold and used in all our experiments. In addition, we generated a second smaller dataset consisting of 1000 points (denoted by a star in our experiments) in order to compare Isomap results with those of the original Isomap paper [Tenenbaum et al., 2000].

The MNIST dataset [LeCun, 2000] consists of handwritten characters images containing digits from 0 to 9 (Figure 3.4b). The size of each image is 28×28 pixels, with 256 grey levels per pixel. Due to computational and memory constraints, in our experiments we used a subset of the MNIST database consisting of 6000 images. According to [Camastra and Vinciarelli, 2001], the optimum intrinsic dimensionality of handwritten digits is 7, whereas the upper bound of the intrinsic dimensionality as determined by EE equals 10.

The ORL (formerly Olivetti) face database contains 400 images of 40 distinct subjects [Samaria and Harter, 1994] (Figure 3.4c). All images were captured against a dark homogeneous background with the subjects in an up-right, frontal position, with tolerance for some side movements. There are variations in facial expression (open/closed eyes, smiling/nonsmiling), and facial details (glasses/no glasses, different skin colours). The images are grey-scale with a resolution of 64×64 pixels. The analysis of relation between recognition rates and dimensionality of embedded space in [Yin et al., 2008b] suggests a value of 10 as the optimal intrinsic dimensionality for this dataset. The upper bound of the intrinsic dimensionality as determined by EE equals 40.

The HumanEva (HE) dataset has been introduced in section 2.3.2.3. In this evaluation only sequences of "walking in a circle" are processed using the actors depicted in Figure 3.5. In all cases, trail 3 of subject 3 was used for training the spectral dimensionality reduction methods (S3T3). We chose frames 750 to 1750 to include a variety of walking postures. Testing was performed using three ground truth datasets and one dataset composed of pose estimates. Datasets were carefully selected to validate robustness of the framework with different actors, who differ in size, body shape, motion style and gender. The ground truth datasets consist of: frames 55 to 315 for male subject 3 in trail 1 (S3T1), frames 340 to 760 for male subject 2 in trial 1 (S2T1) and frames 1 to 400 for female subject 1 in trial 1 (S1T1). The last dataset is a set of body configuration estimates obtained through our auto calibration technique [Kuo et al., 2009] for subject 2 (S2EST) (see also section 3.4.5.1 for details). Intrinsic dimensionality determined by EE equals 2 which is in agreement with other research on modelling walking action [Grochow et al., 2004, Elgammal and Lee, 2004a, Urtasun et al., 2006a, Darby et al., 2010].

3.4.2. Experimental Framework

The proposed methodology is evaluated through qualitative and quantitative analyses of performance using the representatives of the three main spectral families (section 2.2.2.2.2.2), i.e. Isomap, LLE and LE.

3.4.2.1. Setup

All experiments were performed using $\mathbf{K}$ values in the range $<4,30>$. The lower bound of the range corresponds to the minimum neighbourhood size, which allows generating a fully connected graph for the selected datasets. The upper bound is motivated by [van der Maaten et al., 2009]. Their exhaustive evaluation of spectral dimensionality reduction methods on various datasets (including MNIST and OCL) suggests that an upper bound of 15 neighbours allows obtaining the optimal results across most spectral methods. In our work, we took a conservative approach: this value was doubled to ensure that the neighbourhood range is sufficiently large to find the optimal neighbourhood size in all conducted experiments even for datasets not investigated by [van der Maaten et al., 2009]. In multidimensional spaces, geodesic distances are used, whereas on the plane we employ Euclidean distances as suggested in [Samko et al., 2006]. RBFN was trained from high to low dimensional space as described in section 2.2.2.4.4 and a resulting mapping function is given by the equation (2.62).

3.4.2.2. Experiments

First, we evaluate qualitatively the novel MI estimator against current approaches, i.e. residual variance (RV), spearman rho (SR) and Procrustes analysis (PA) measures. This is performed using the synthetic dataset for which the underlying structure is known so the quality of embedded space can be judged visually (section 3.4.3).

Then, two classical pattern classification problems, face and handwritten digit recognition, are considered in order to analyze the quantitative performance of the MI metric (section 3.4.4). We do not perform any pre-processing or normalisation of the data in order to prevent any information lost. It is important to note that, in this work, we did not focus on designing a state of art classification system, but on comparing existing metrics with the one we proposed using a standard classification framework based on a real application.

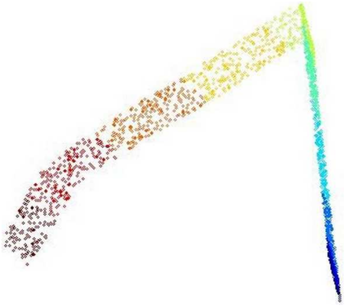
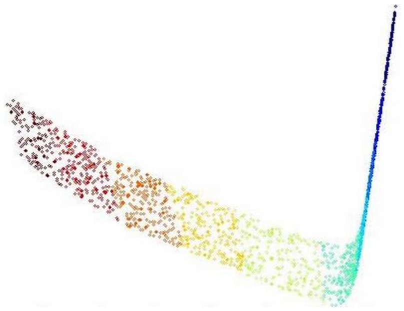
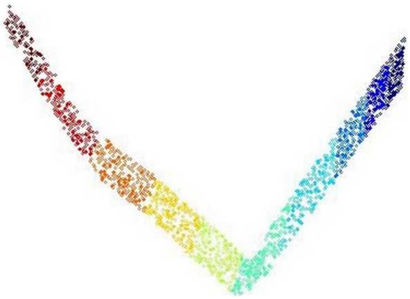
In addition, we apply all measures in a pose recovery application (section 3.4.5), where human motion is represented by motion capture data as described in section 2.3.2.3. In the first experiment, we consider the simplest scenario, in which we train and test with the same subject, i.e. S3, using different trials. This should allow finding the lowest bound of the reconstruction error which can be obtained within our framework. In the next two experiments, we evaluate our approach using MoCap data from subjects 2 (S2T1) and 1 (S1T1). Both datasets differ considerably from training set as actions are performed by very different subjects, see (Figure 3.5). Since input data are ideal estimates, the reconstruction error should highlight differences introduced by variations of walking styles and body frames between testing and training characters. Afterwards, in the fourth experiment, we take the corrupted 3D pose estimates produced by our algorithm for subject 2 on a walking sequence (S2EST) and we refine them according to the framework (see section 3.4.5).

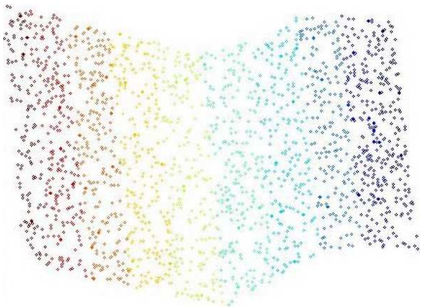
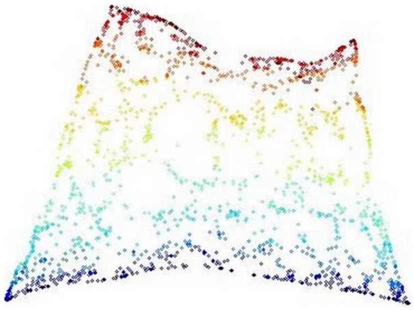
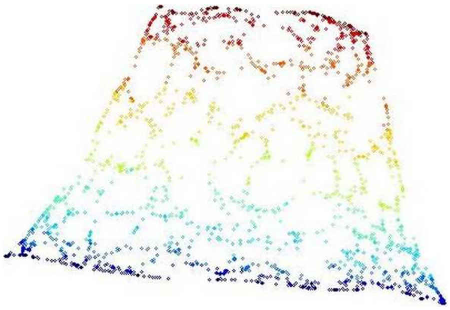
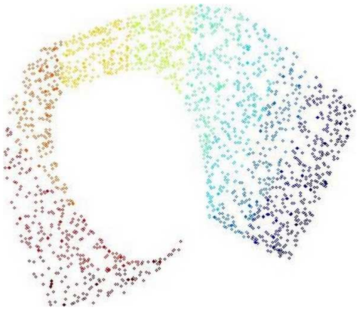
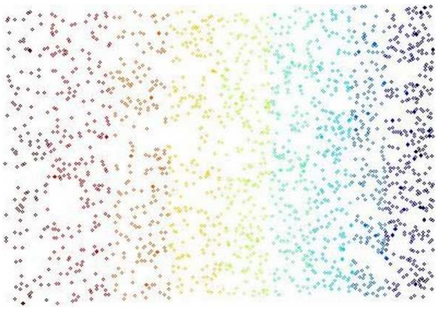
Finally, in the last experiment we show superiority of graph-based RBFN in comparison with standard RBFN (section 3.4.6). This is achieved by repeating the classification experiments with digits and faces recognition and pose recovery experiments using the new mapping function whose structure is inferred automatically from the spectral graphs.

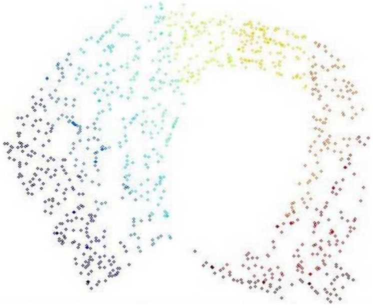
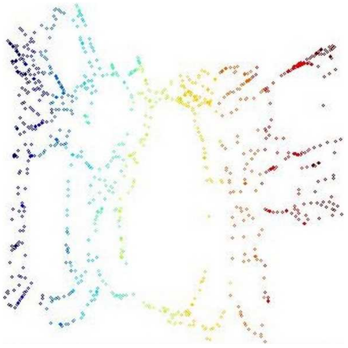
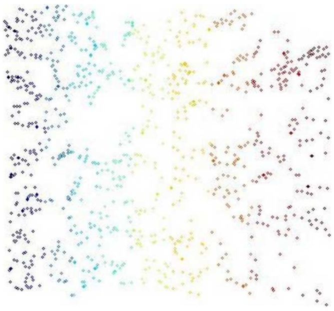
3.4.3. Artificial Dataset Evaluation

Table 3.1 presents the low dimensional spaces of the swissroll dataset produced by Isomap, LE and LLE using the estimated neighbourhood sizes calculated by RV, SR, PA and the proposed MI measure.

Table 3.1. The low dimensional spaces of swissroll with estimated and recommended neighbourhood sizes for Isomap, LE and LLE according to coefficients RV, SR, PA and MI.

Method (recommended K)	Coefficient (estimated K)	Visualisation
LLE (20) [Roweis and Saul, 2000]	residual variance (11)	
	spearhman rho (22)	
	procrustes analysis (8)	

	mutual information (20)	
LE (5–15) [Belkin and Niyogi, 2002]	residual variance (8)	
	procrustes analysis, *mutual information* (5)	
Isomap (–)	residual variance (21)	
	spearhman rho, procrustes analysis, *mutual information* (18)	

Isomap (7) [Tenenbaum et al., 2000] (The swissroll dataset with 1000 points instead of 2000 points)	residual variance (9)	
	spearhman rho (4)	
	procrustes analysis, *mutual information* (7)	

In all cases, the MI measure was able to identify very good low dimensional representation of swissroll dataset, i.e. an embedded space which manages to unroll manifold and preserves local structure. Moreover, estimated values of K using MI are in agreement with parameters which were recommended in the original papers [Tenenbaum et al., 2000, Roweis and Saul, 2000, Belkin and Niyogi, 2002]. Although other measures usually select reasonable low dimensional representations, their quality is not consistent. For instance, the local structure is distorted in most experiments involving RV/SR. Although PA seems to behave similar to MI, in the case of LLE the very different neighbourhood size returned by PA leads to the production of an embedded space of inferior quality.

3.4.4. Classification Evaluation

In this experiment, grey-level images of digits and faces are vectorized using raster-scan order into 784 and 4096 dimensional feature vectors respectively. The recognition of either digits or faces is performed according to the 10-fold cross validation strategy, where we divide a dataset into ten distinct partitions. For each partition, we reduce dimensionality of remaining dataset and train RBFN with the standard RPCL algorithm. Then, each partition is projected into the low dimensional space and classification is performed using a first nearest neighbour classifier (Ho, 1998). Finally, classification accuracy is calculated by averaging over the ten partitions. For each dataset, estimation of optimal neighbourhood size for dimensionality reduction is calculated using RV, SR, PA and MI. Moreover, the actual optimal K (Opt) is calculated experimentally by an exhaustive evaluation of classification accuracy for all values of $\mathbf{K}$ within the range $< 4, 30 >$(see section 3.4.2). In addition, using the optimal value, we evaluate the classification accuracy of the scheme (Opt*) which includes graph-based RBFN (G-RBFN).

Table 3.2 and Table 3.3 show the results of these experiments which were conducted with two sets of IDs as defined in section 3.4.1. Note that the huge computational cost of applying PA on the very high dimensional faces dataset (dimensionality of 4096) did not allow us to obtain the results for this measure using our processing capabilities (16-node cluster).

Table 3.2. Percentage accuracy of handwritten digits recognition, where Opt is calculated experimentally using RBFN by exhaustive evaluation of all considered neibhourhood sizes, whereas Opt* corresponds to the usage of graph-based RBFN

%	ID	RV	SR	PA	MI	Opt	Opt*
Isomap	10	88	88	88	88	89	90
LLE		62	63	59	78	78	82
LE		79	79	80	80	80	84
Isomap	7	85	82	84	85	85	87
LLE		56	63	53	74	74	77
LE		75	75	74	76	77	80

Table 3.3. Percentage accuracy of faces recognition, where Opt is calculated experimentally using RBFN by exhaustive evaluation of all considered neibhourhood sizes, whereas Opt* corresponds to the usage of graph-based RBFN

%	ID	RV	SR	PA	MI	Opt	Opt*
Isomap	40	76	73	–	77	77	77
LLE		78	78	–	80	80	80
LE		67	67	–	67	68	73
Isomap	10	65	57	–	76	76	76
LLE		55	55	–	61	62	62
LE		62	50	–	63	63	63

In agreement with our previous experiments, neighbourhood sizes estimated by the MI measure produce consistently better classifications than those suggested by other metrics regardless of the chosen intrinsic dimensionality. Moreover, the performance of nearest neighbour classifier is optimal or near-optimal when using MI, for a given dimensionality reduction method. Results also reveal that unlike LLE and Isomap, LE is not very sensitive to neighbourhood size selection. As expected, decrease of intrinsic dimensionality results in a decline of accuracy since more discriminative information is discarded during dimensionality reduction. Two dimensional visualisation of the best low dimensional space obtained with Isomap for the digit dataset is presented in Figure 3.6.

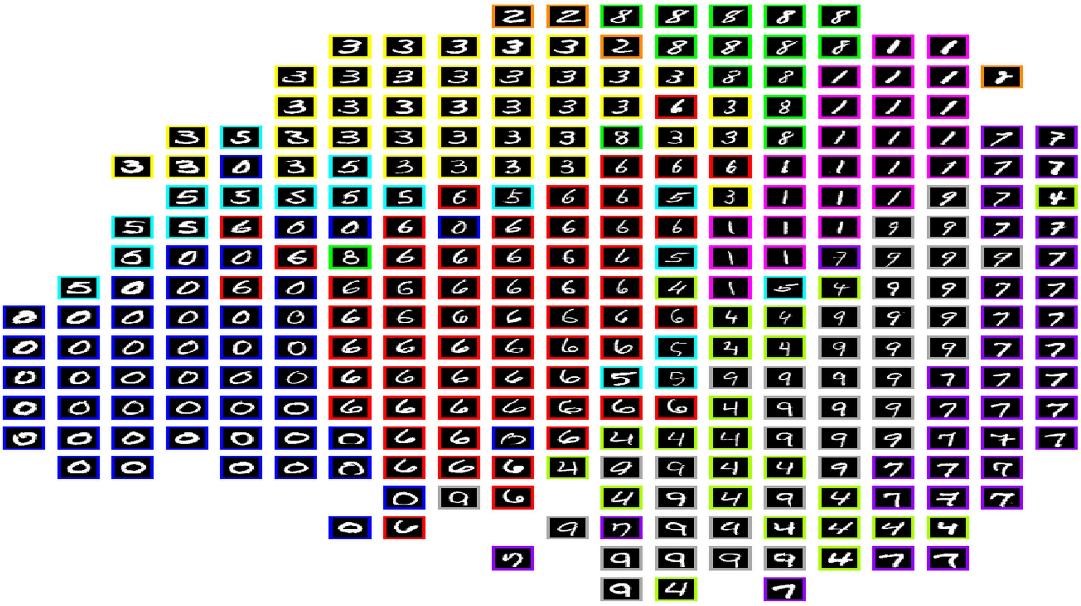


Figure 3.6. Two dimensional visualisation of the best low dimensional space obtained with Isomap for MNIST data subset.

Although we used classification experiments to compare quantitatively the measures, our aim was not to produce a state of the art classifier, but to demonstrate that our innovations could be applied successfully to representatives of the three main spectral families, i.e. Isomap, LLE and LE. We would suggest readers with a special interest in classification to apply our advanced techniques to spectral methods which were developed especially to handle that task. They include discriminant Isomap [Yang, 2003], supervised LLE [De Ridder et al., 2003] and semi-supervised LE [Zheng et al., 2008].

Finally, the proposed G-RBFN (Opt*) achieves better classification accuracy in comparison to the standard RBFN (Opt) for all considered dimensionality reduction methods (Table 3.2 and Table 3.3). The efficiency improvement is especially noticeable for large datasets like handwritten digits (Table 3.2), when the enormous graph can be assembled with distinctive groups of well connected points due to usage of the **K** -nearest neighbour procedure, which allows to form representative centres for the mapping. Further evaluation of the novel mapping function is provided in section 3.3.2.

3.4.5. Application to Pose Recovery

To demonstrate the performance of the proposed methodology for automatic configuration of spectral dimensionality reduction methods, it is applied to the refinement of 3D body pose estimates in pose recovery application. Here, a human motion is represented by a sequence of 52-dimensional feature vectors extracted from motion capture data as described in section 2.3.2.3.

3.4.5.1. 3D Pose Recovery Framework

Our proposed 3D pose recovery framework aims at estimating a 3D pose from 2D joint locations using a single uncalibrated camera [Kuo et al., 2009]. Figure 3.7 shows an in-depth insight of this process.

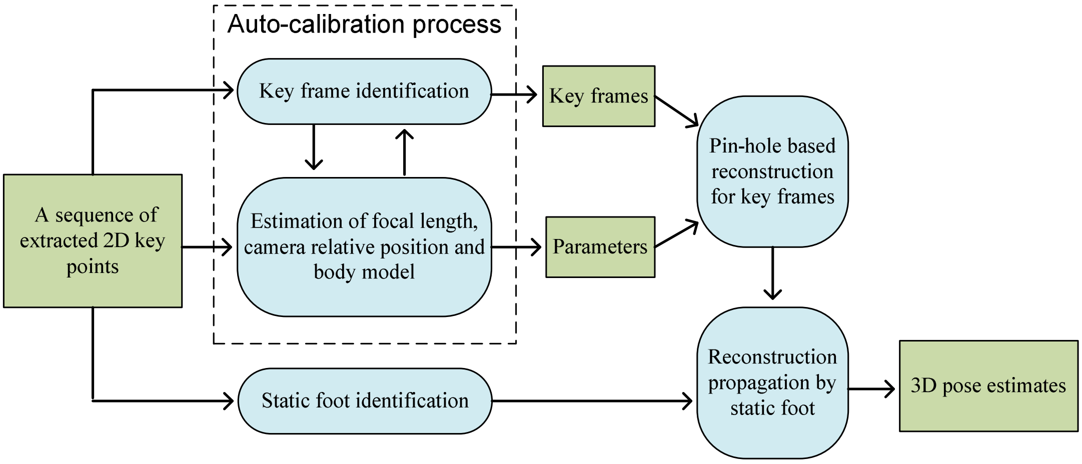

Auto-calibration process
Key frame identification
Key frames
Pin-hole based reconstruction for key frames
A sequence of extracted 2D key points
Estimation of focal length, camera relative position and body model
Parameters
Static foot identification
Reconstruction propagation by static foot
3D pose estimates

Figure 3.7. Generation of 3D body pose estimates.

First, we assume that 2D joint positions of the human body have been extracted from a video sequence using any 2D pose recovery method (for instance

[Kuo et al., 2008]). These 2D key points are employed to perform camera auto-calibration for a set of key frames automatically selected in the sequence [Kuo et al., 2007]. This is an iterative process, which consists of two steps:

- Selection of specific key frames by exploiting a human bipedal motion constraint that certain body joints become coplanar within a motion cycle.
- Estimation of camera calibration parameters, i.e. focal length and camera relative position, using Tsai's coplanar calibration method [Tsai, 1987].

In addition to the camera parameters and key frames identification, the process generates also a 3D coplanar model representing the 3D configuration of the set of coplanar body joints at these frames. The obtained 3D coplanar points correspond to shoulders (3 points) and hips (2 points) according to human biomechanics. Since sufficient knowledge has been accumulated, a pin-hole projection model is applied to reconstruct other parts of the 3D figure in the world space, i.e. limbs and head (Figure 3.8). The projection line of each key body point on the image is established using the estimated focal length. Their corresponding 3D points are located on the projection lines according to the camera relative position and the body model. The body model is a 3D skeletal representation of the human body (see section 2.3.2.3). It is constructed from the calibrated 3D coplanar model with known body ratios. Since this problem is ill-constrained ($\mathbb{R}^2 \to \mathbb{R}^3$), multiple postures are generated. A pose selection mechanism is then required to extract the correct posture. In this experiment, we have chosen poses with the smallest error in comparison to the ground truth.

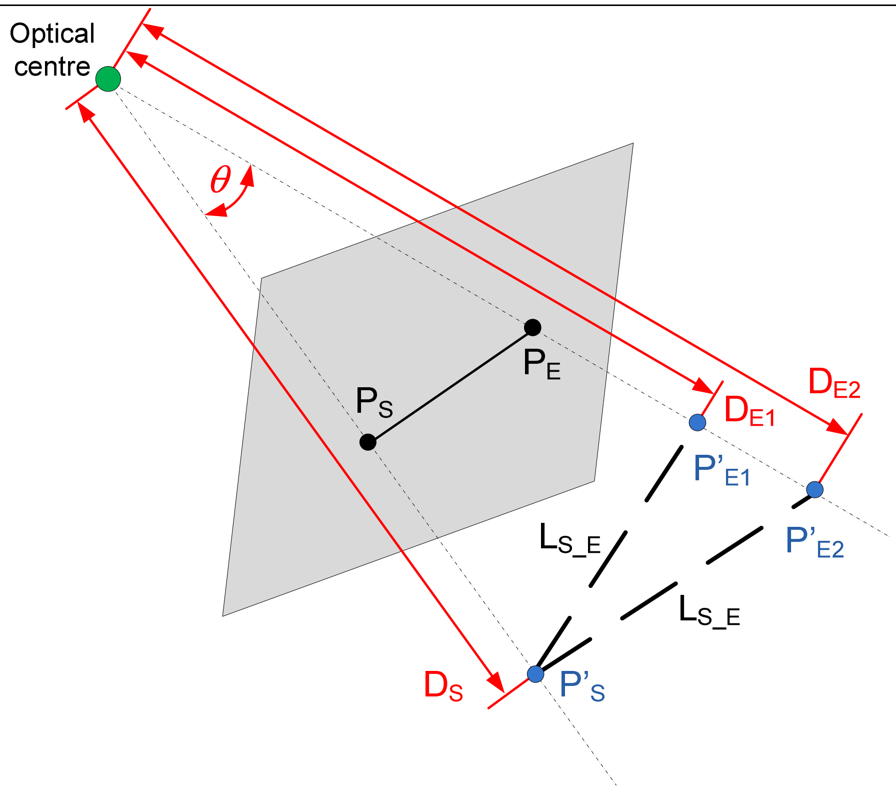


Figure 3.8. Pin-hole camera model for 3D pose reconstruction applied to the reconstruction of the left arm: P_S and P_E – shoulder and elbow image points, P'_S – known 3D shoulder coplanar point, P'_{E1} and P'_{E2} – two proposals for 3D elbow reconstruction by taking into account depth ambiguity, L_{S_E} - expected segment length between two successive key points, $D_?$ – distance between 3D point and optical centre.

In order to recover poses for other frames, another human bipedal motion constraint, i.e. the presence of a foot on the ground (so called static foot), is exploited to propagate the parameters of the pin-hole projection model from one frame to another. Human biomechanics reveals that at any moment at least one foot is in contact with the ground in most types of bipedal motion. This static foot exists because the body requires at least one limb to support its weight. Motion is achieved by switching weight support to the other foot. Both feet can only be off ground for a short moment if any, e.g. running. The static foot is identified effectively and accurately by comparing displacement of foot points between consecutive frames [Kuo et al., 2009]. The obtained static foot constraint is exploited for pose recovery, since knowledge of the 3D posture at one frame also provides the 3D coordinates of one foot in the next frame. Therefore, these coordinates are used as the starting point of pin-hole pose reconstruction for the next frame. As a result, postures are propagated recursively forward and backward in time from the reconstructed key frames to their neighbouring frames. Since there are multiple key frames within a given sequence, a linear combination of the propagated postures from each key frame is calculated to generate the final one. Weights are introduced to penalise postures which are temporally further away from their key frames.

3.4.5.2. 3D Pose Refinement Framework

Because of the $2D \rightarrow 3D$ ambiguity, the pipeline described in the previous section as well as many other activity independent methods (section 2.3.2.1) produces imperfect 3D estimates of poses. The accuracy of this estimation can be significantly improved by incorporating learned prior models of activity into pipeline. The 3D pose refinement framework is presented in Figure 3.9 [Lewandowski et al., 2009]. It is composed of two parts: activity learning, which is an automatic offline process, and the online procedure of pose refinement.

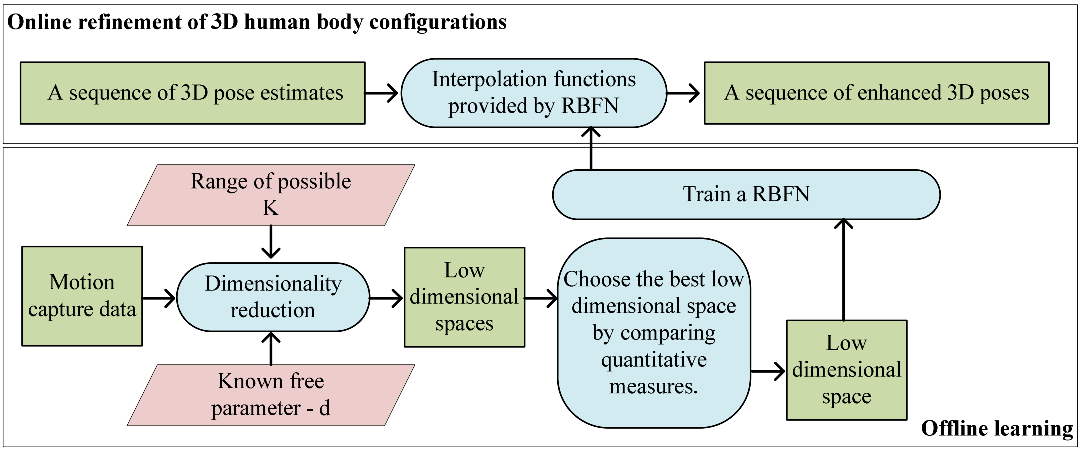

Online refinement of 3D human body configurations
A sequence of 3D pose estimates
Interpolation functions provided by RBFN
A sequence of enhanced 3D poses
Range of possible K
Train a RBFN
Motion capture data
Dimensionality reduction
Low dimensional spaces
Choose the best low dimensional space by comparing quantitative measures.
Known free parameter - d
Low dimensional space
Offline learning

Figure 3.9. Automatic refinement of 3D body pose estimates.

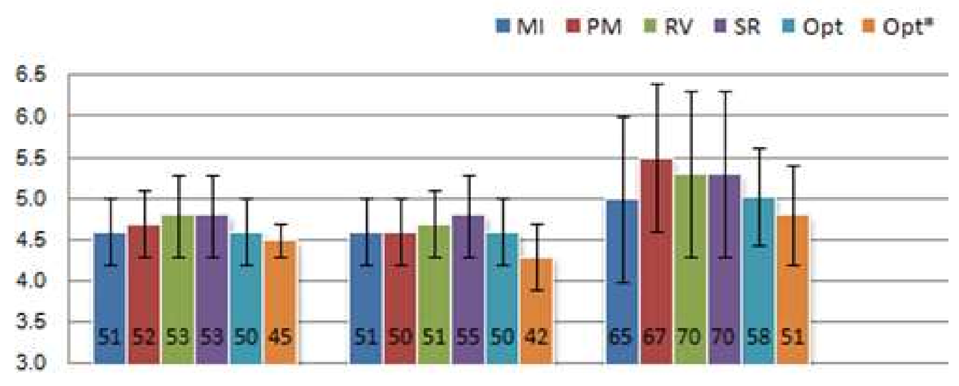
During the learning stage, the space of human motion is reduced following the procedure proposed in section 3.3.1 and the best low dimensional representation is chosen according to the quantitative metric. Then the obtained space is employed, first, for designing the structure of the RBFN and, subsequently, for training automatically the network in order to provide a bidirectional projection mechanism between spaces.

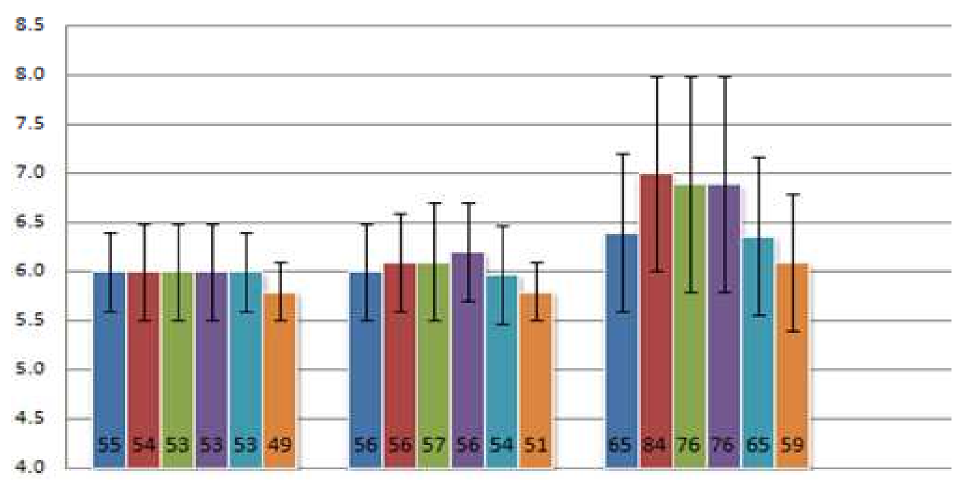
The online module of the framework deals with the actual problem of 3D pose recovery. In principle, it can be applied to pose estimates produced by any activity independent method (section 2.3.2.1). In this experiment, we consider the output of algorithm described in the previous section 3.4.5.1 as a sequence of 3D pose estimates. In the refining process, an inaccurate 3D skeleton is projected into the embedded space using the corresponding mapping function. Then, this projection is associated to its nearest low dimensional training neighbour according to the Euclidean distance. Finally, the determined neighbour is projected back to the human motion space as the refined 3D pose estimate.

3.4.5.3. Results

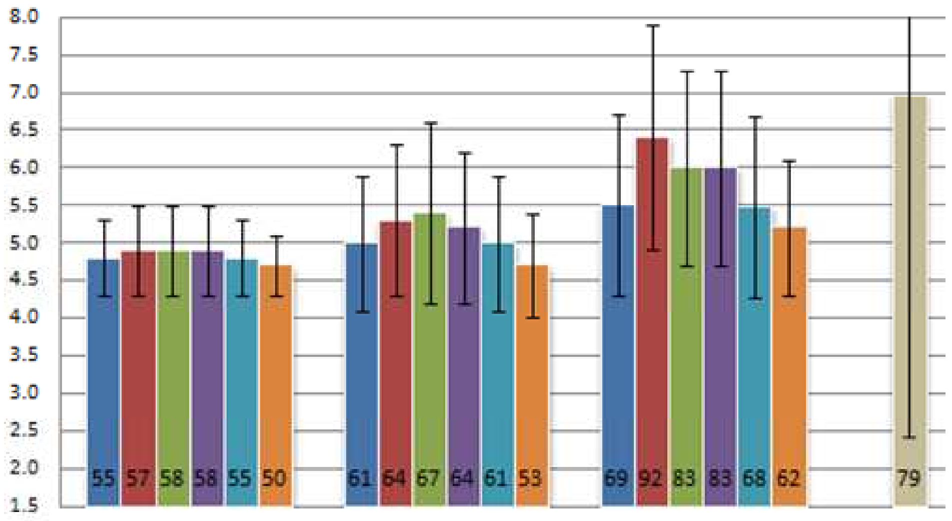
In the Table 3.4, the MAE angle error and the corresponding RMS error (see section 2.3.2.3.2 for details about these metrics) are provided for all considered methods obtained in the second, third and fourth experiment using human motion capture data (section 3.4.2.2). Detailed quantitative results of the first experiment are not provided, because all methods performed very well and the impact of neighbourhood size estimation on results is marginal. The errors for the embedded space calculated using the optimal $\mathbf{K}$ (Opt) are compared with those obtained for the low dimensional space selected by the four quantitative measures. The actual optimal $\mathbf{K}$ is calculated experimentally by an exhaustive evaluation of all spaces similarly like to the classification experiments. In addition, using the optimal value, the pose recovery accuracy of the scheme, which includes graph-based RBFN (G-RBFN) (Opt*), is evaluated.

Table 3.4. Mean of absolute angle error (MAE) and standard deviation for the best low dimensional spaces according to four coefficients discovered by different methods for S2T1, S1T1 and S2EST datasets. S2EST corresponds to the initial estimation error. The root mean square error (RMS) error in mm is depicted within bars.

<table>
<tr>
<td>Second experiment with S2T1 dataset</td>
<td>


</td>
</tr>
<tr>
<td colspan="2">Isomap LE LLE</td>
</tr>
<tr>
<td>Third experiment with S1T1 dataset</td>
<td>


</td>
</tr>
<tr>
<td colspan="2">Isomap LE LLE</td>
</tr>
</table>

Fourth experiment with S2EST dataset	
	Isomap LE LLE S2EST error

Analysis of the different quantitative measures used to choose **K** , demonstrates once again the superiority of MI in all conducted experiments. Although it does not always identify the optimal **K** , it outperforms the other metrics and systematically produces more stable and accurate results. In this evaluation both Isomap and LE seems to be less sensitive to the selection of **K** than LLE. As expected, the accuracy of estimation decreases when subjects differ the most from the one used for training, however it is worth to point out, that all methods enhanced significantly the quality of pose estimates in the fourth experiment which validates the proposed refinement methodology. An example of low dimensional manifold discovered by Isomap is depicted in Figure 3.10.

Similarly to the classification experiment (section 3.4.4), the proposed G-RBFN (Opt*) outperforms the standard RBFN (Opt) for all considered dimensionality reduction methods. Further comparison of both mapping functions is presented in section 3.3.2.

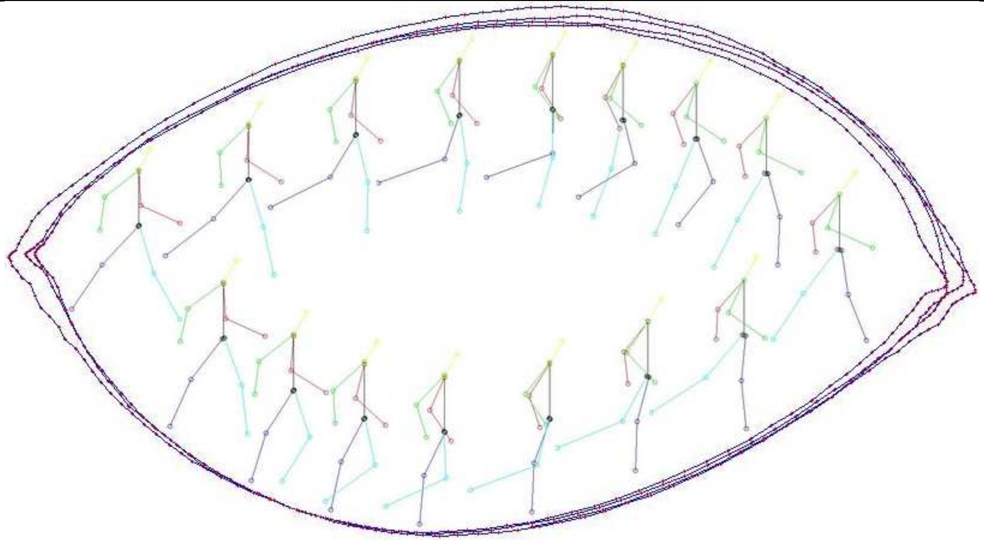


Figure 3.10. Representation of the best low dimensional manifold with corresponding poses discovered by Isomap according to the MI measure for training set S3T3.

A more detailed comparison of pose estimates before and after refinement for the fourth experiment is shown in Figure 3.11 where Isomap was applied using the **K** value predicted by MI. Since we use different subjects for training and testing, the quality of refinement cannot go over a certain threshold which expresses individual differences between walkers. Therefore, for some frames 350-410, 561-568, 605-655 pose estimates are worst after refinement. However, in average, accuracy is improved significantly (30%) and our scheme provides much more stability in pose prediction: standard deviation drops from 4.5° (i.e. 41.6mm) to 0.5° (11.8mm). Figure 3.12 illustrates the effect of our framework by showing refined poses against initial estimates and ground truth.

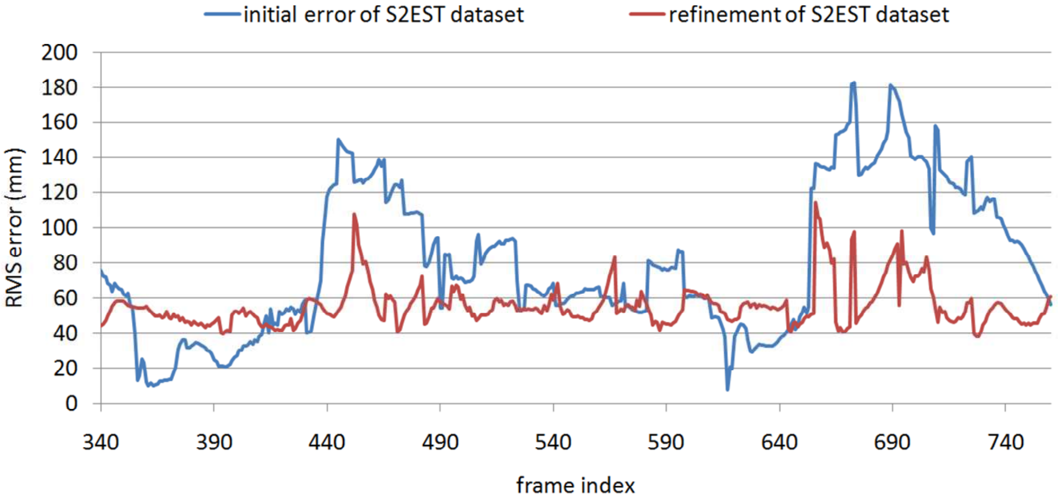


Figure 3.11. Refinement results of the dataset S2EST for each frame.

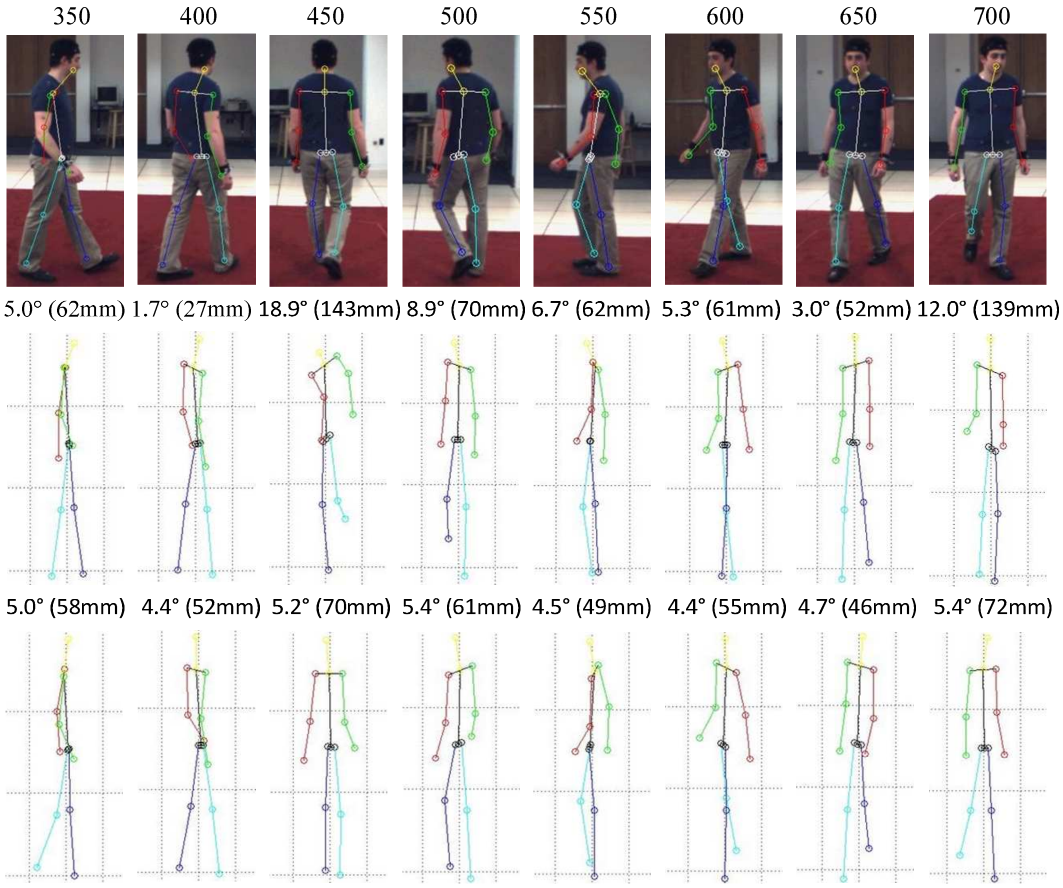


Figure 3.12. Refinement results; first row: ground truth with frame index; second row: estimated pose with initial MAE and RMS error; third row: refined pose with output MAE and RMS error. As it can be noticed, on average, accuracy of estimation in the third row is improved significantly (~30%) in comparison to the second one. Moreover, our scheme provides much more stability in pose prediction.

3.4.6. Graph-based Mapping Evaluation

Regarding the efficiency of graph-based RBFN, Table 3.2, Table 3.3 and Table 3.4 show that this new scheme improves significantly the quality of the mapping produced by standard RPCL RBFN in all experiments. Further comparison between those two mapping methods is provided in Figure 3.13 and Figure 3.14, where classification accuracy and processing time are measured for various sizes of the digits dataset. Here, LE is used for dimensionality reduction as a representative of spectral methods.

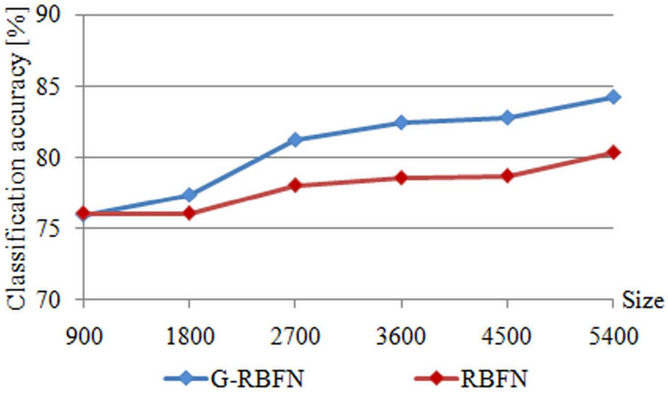


Figure 3.13. Classification accuracy comparisons between graph-based RBFN and standard RPCL RBFN according to digits dataset size (ID = 10).

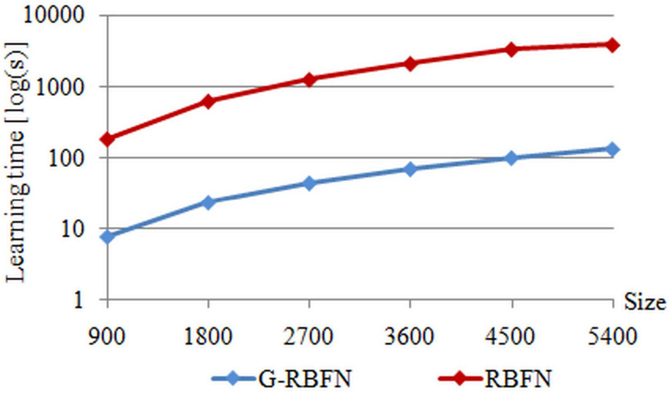


Figure 3.14. Classification processing time comparisons between graph-based RBFN and standard RPCL RBFN according to digits dataset size (ID = 10).

First, whatever the size of the training set, classification accuracy using graph-based RBFN is higher than for standard RBFN. Moreover, graph-based RBFN is in the order of magnitude computationally more efficient (Figure 3.14).

3.4.7. Discussion

Consistently across all experiments, the MI metric demonstrates its accuracy in the identification of the optimal neighbourhood size. In contrast to other metrics, the proposed MI is validated successfully in various domains, including artificial data (section 3.4.3), face and handwritten digit recognition (section 3.4.4) and human pose recovery (section 3.4.5). This suggests that MI is very versatile since it is less sensitive than the other metrics to a dataset nature.

In parallel, we prove the advantageous of G-RBFN over the standard RBFN in all conducted experiments. The superiority of the novel mapping function is especially evident for handwritten digits recognition (section 3.4.4) and human motion refinement (section 3.4.5.3), when the appropriate localisation of centres is significantly more challenging because of vast amount of training features and their high dimensionality.

3.5. Summary

In this chapter, a framework is proposed to automatically configure spectral dimensionality reduction methods. This is achieved twofold.

First, we introduce the MI metric to estimate neighbourhood size. All experiments demonstrate that MI outperforms previously used metrics independent on the spectral methods and the dataset. Embedded spaces produced by MI are visually convincing. Moreover, our quantitative study, i.e. classification and pose recovery experiments, confirms its superiority: low dimensional spaces selected by the MI measure consistently provide better accuracy regardless of the estimated ID.

Moreover, unlike PA, MI proved its ability to handle very high dimensional datasets.

Secondly, we propose graph-based RBFN to provide mapping between embedded and data spaces using the efficient MCL algorithm, as part of the learning process of spectral dimensionality reduction methods. This scheme outperforms significantly standard RBFN mapping in both accuracy and computational efficiency.

To conclude, the effectiveness of our contribution has been validated qualitatively and quantitatively in various domains. Results prove that the proposed MI-based neighbourhood selection procedure in combination with the graph-based RBFN allow to automatically configure the representative approaches (LLE, LE, Isomap) of the three main families of spectral dimensionality reduction methods. As a consequence, our flexible and unified methodology overcome limitations of embedded based approaches and thus may benefit to many areas where scientists face the problem of analysing high dimensional data.

4. Temporal Laplacian Eigenmaps

4.1. Introduction

The previous chapter demonstrates flexibility and usefulness of spectral dimensionality reduction methods for exploration of highly nonlinear and multivariate datasets. Although the preservation of some geometrical property of an underlying manifold is a valid goal for various applications, there are many situations in which an alternative approach is desired. In particular, when dealing with multidimensional time series data, the temporal order which is imposed on observations is expected to be more intuitive and advantageous constraint for a dimensionality reduction process.

A multidimensional time series is a collection of high dimensional data observations measured sequentially through the time, which are very often nonlinear. Time series data are widely available in different fields including medicine, finance, science, engineering and computer vision. Therefore modelling of time series data effectively becomes an essential challenge for the machine learning community. Since multidimensional time series can bear a lot of data variations, noise, redundancies and correlations hiding important relationships, it is extremely difficult to understand and process them. A dimensionality reduction process should eliminate these undesired properties from the time series, while ensuring the maximum possible preservation of original information. Analysis of time series using dimensionality reduction methods has only recently been investigated by the research community.

The standard dimensionality reduction methods (section 2.2.2) are clearly inappropriate for this task, since they assume that the observed data samples are independent, thus any temporal correlation between data samples is ignored. Since successive points at each time step of time series are expected to be highly correlated, a few temporal extensions of the standard methods were proposed including Spatio-Temporal Isomap [Jenkins and Mataric, 2004], Back-Constrained Gaussian Process Latent Variable Model [Lawrence and Quinonero-Candela, 2006] and Gaussian Process Dynamical Model [Wang et al., 2006]. Although, these approaches exploit some temporal constraints during a dimensionality reduction process, we will show that they are not designed to preserve the global topology of the time series manifold. As a result, they fail to produce a unique and informative low dimensional representation in the presence of data variations between time series. Moreover, they are computationally expensive and often require a set of empirically chosen parameters. These algorithms are discussed in details in the subsequent section 4.3.

In this research, we address these limitations and contribute to the state of the art by introducing a novel spectral dimensionaliy reduction method, called Temporal Laplacian Eigenmaps [Lewandowski et al., 2010c] (TLE). Our proposed algorithm exploits temporal relationships and dependencies of time series as key constrains during the nonlinear dimensionality reduction process. In contrast to previous approaches, we introduce two types of constraints: temporal within time series and spatio-temporal between different time series. This is achieved by introducing two forms of intuitive temporal graphs which are incorporated into the LE framework. In addition, neighbourhood sizes of both graphs are derived automatically from data analysis. As a consequence, our method aims at preserving a temporal structure of multivariate time series instead of the commonly used geometric structure. This fundamentally different concept allows automatically producing meaningful and generalised low dimensional representations tailored to multivariate time series data. Exhaustive experiments on a couple of computer vision applications demonstrate the effectiveness of the proposed methodology for modelling different types of multidimensional time series and its superiority in comparison to other dimensionality reduction techniques in terms of accuracy and efficiency. In addition, its lower computational cost and generalisation abilities suggest it is scalable to larger datasets.

The remainder of this chapter is organised as follows. The next section 4.2 introduces formally the concept of multivariate time series. Then, section 4.3 discusses the relevant work in the dimensionality reduction of time series data. The detailed information about basic version of Laplacian Eigenmaps and our temporal extension of Laplacian Eigenmaps are introduced in section 4.4. Then results of evaluation are presented in section 4.5. Section 4.6 concludes the chapter.

4.2. Multivariate/Multidimensional Time Series

A time series $S=\{y_t \mid t=1..\mathbf{t},\ y_t\in\mathbb{R}^{\mathbf{D}},\mathbf{D}\in\mathbb{Z},\mathbf{D}>0\}$ is a sequential collection of observations generated by a dynamical system (i.e. time series source) for a specific phenomenon. Here, we define time as a set of discrete values which is indexed by t, whereas $\mathbf{t}$ denotes the number of observations in the sequence S. If $\mathbf{D}$ is equal to one then the time series is referred to as univariate/one-dimensional, and if it is greater than one the time series is referred to as multivariate/multidimensional [Hannan, 1970, Chatfield, 1996] (MTS).

In this work, we are interested in modelling multivariate time series ($\mathbf{D}>1$), since their high dimensionality creates challenges for machine learning and data mining algorithms. As a result, our space of high dimensional features is now a set of $\mathbf{L}$ multivariate time series, each with $\mathbf{T}_l$ (l=1..$\mathbf{L}$) features:

$$\begin{aligned} Y &= \{s_l \mid l=1..\mathbf{L}\} = \{y_{lt} \mid l=1..\mathbf{L}, t=1..\mathbf{T}_l\} \\ &= \{y_i \mid i=1..\mathbf{N}, y_i\in\mathbb{R}^{\mathbf{D}}, \mathbf{D}>1, \mathbf{D}\in\mathbb{Z}\} \end{aligned} \tag{4.1}$$

where $\mathbf{N} = \sum_{l=1}^{\mathbf{L}} \mathbf{t}_l$. In addition, a set Y may consist of MTS which are issued by different sources (for example S_1, O_1, P_1 in Figure 4.1b) and/or repetitions from a single source (for example S_1, S_2, S_3 in Figure 4.1a). The intra-data variations between MTS from different sources are not on the same scale as the inter-data variations from a single source. This can be conceptually expressed in terms of relative difference between dynamic time warping distances (DTW) (see appendix A.1) for any pair of time series:

$$DTW(S_1,S_2) \ll DTW(S_1,O_1) \tag{4.2}$$

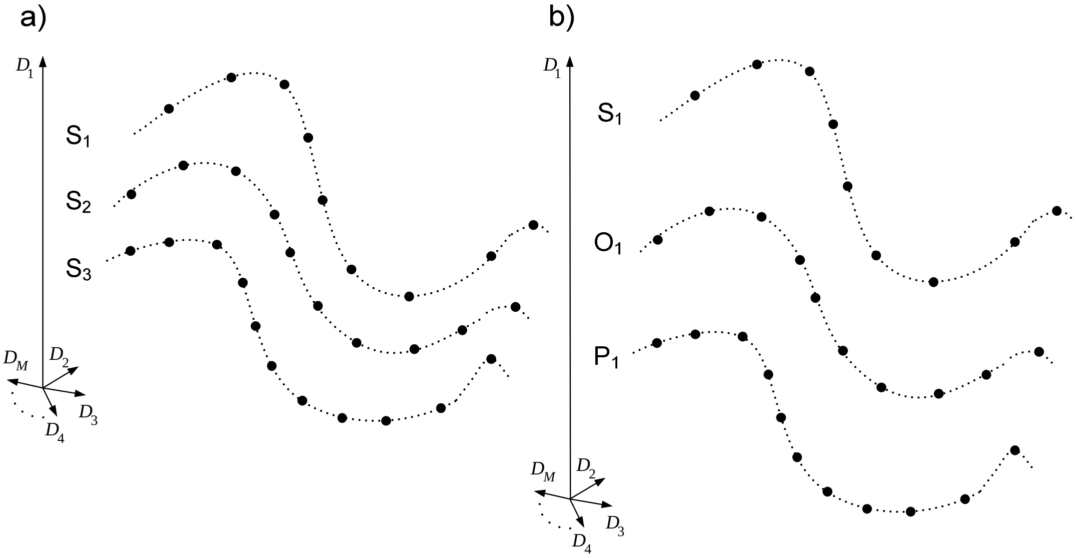


Figure 4.1. Example of multivariate time series data issued from a single source (a) or different sources (b).

Finally, we define 'style' as the intra-data and inter-data variations between two or more time series representing a similar phenomenon. They may be produced by different sources and/or multiple repetitions (or cycles) from a single source.

In the rest of the thesis, the term ‘time series’ refers to multivariate/multidimensional time series. Moreover, the term ‘time series repetitions’ bears on time series repetitions issued by single and/or multiple sources.

4.3. Related Work

A presentation about relevant time series frameworks has been provided in section 2.2.3. In this section, we discuss in more details the current state of the art methods for the dimensionality reduction of time series data. These approaches are used as references in the evaluation process (section 4.5). In addition, for completeness and clarity reasons, we provide short descriptions of three techniques in appendix A which are used in our implementation; although alternative approaches could be chosen. They are dynamic time warping distance (DTW) (A.1), optical flow (A.2) and Hausdorff distance (A.3).

4.3.1. Spatio-Temporal Isomap

The spatio-temporal Isomap [Jenkins and Mataric, 2004] (ST-Isomap) is an extension of standard Isomap algorithm designed for time series data. The structure of the algorithm remains the same (see section 2.2.2.2.2.2) with two extra steps added for temporal windowing and temporal augmentation of data. First, the input data is windowed into temporal blocks of a pre-defined size. As a result, some temporal history is introduced into each data point. Then, the standard local neighbourhood graph is constructed (section 2.2.2.2.2.2.1) and the corresponding matrix of distances between neighbouring points is computed using pre-processed data (section 2.2.2.2.2.2.2).

The key novelty of ST-Isomap is a definition of adjacent temporal neighbours and K-nearest nontrivial neigbhours. Adjacent temporal neighbours are adjacent points in the sequential order of the current point, whereas K-nearest nontrivial neighbours are defined by Jenkins et al. as follows:

> *"A point y_j to be a nontrivial match within the local neighbourhood of a point y_i if it is closest matching point on its trajectory through the neighbourhood"*

These neighbours are used to empirically alter the original distances in the graph (matrix) of local neighbours to emphasise similarity between spatio-temporal related points using constant pre-defined factors. As the value of this factor increases, the distance between data pairs with spatio-temporal correspondences decreases and their similarity is strengthened. These spatio-temporal relationships are then propagated globally via a shortest-path mechanism and the dense matrix of distances between all points is genearated. Finally, the MDS (section 2.2.2.2.1.2) is applied on obtained matrix to produce **d** dimensional embedded space similarly to standard Isomap.

The ST-Isomap algorithm has pioneered in research on dimensionality reduction of time series. However, a few important drawbacks limit its usefulness in many applications. First, ST-Isomap cannot discover the global temporal pattern of time series, since the temporal information is just employed to alter the geometric relationships between points. As a result, ST-Isomap conceptually still aims at preserving the global geometric topology of data instead of the temporal topology. In addition, the introduction of the temporal information into the geometric constraints requires two pre-defined constant factors. These factors, which have to be chosen manually, control similarity between data pairs with spatio-temporal correspondences. Another crucial disadvantage is the requirement of the prior knowledge about the number of the K-nearest nontrivial neighbours. In fact, the algorithm is very sensitive to this parameter. It fails to produce a meaningful representation whenever it is chosen inappropriately, especially if it exceeds the actual number of time series repetitions in data. Moreover, the naive procedure for selecting the nontrivial neighbours depends heavily on the size of the pre-defined searching window and does not take into account neither spatial nor temporal similarity between different time series. Finally, ST-Isomap inherits from its parent method the computational complexity, thus the processing time grows cubically with the number of points in a dataset.

4.3.2. Back-Constrained Gaussian Process Latent Variable Model

Back Constrained GPLVM [Lawrence and Quinonero-Candela, 2006] (BC-GPLVM) imposes high dimensional constraints on a latent space to enforce the local distance preservation and implicilty the temporal coherence of time series.

Since the standard GPLVM (see section 2.2.2.3.2.2.2) focuses primarily on modelling the data global structure, there is no guarantee that the local temporal order of time series is retained in a latent space. The smooth mapping in GPLVM implies that dissimilar points in the data space remain distant in the latent space. However, there is no constraint to prevent two points which are nearby in the data space to be placed far apart in the latent space, thus creating discontinuities of time series in the low dimensional representation. As a consequence, GPLVM can be seen as a dissimilarity preserving method [Lawrence and Quinonero-Candela, 2006].

To tackle this problem and obtain a continuous representation of time series, BC-GPLVM constrains a latent space to be a smooth mapping from a data space, i.e. it forces two points to be always nearby in the latent space if their data space counterparts are also relatively close. Therefore, rather than maximising the likelihood (2.44) with respect to X directly, each element of X is replaced with the form of the kernel based regression mapping from the observed space to the latent space ($m = 1..\mathbf{d}$):

$$x_{im} = g_m(y_i) = \sum_{j=1}^{\mathbf{N}} w_{mj} \kappa(y_i, y_j) \tag{4.3}$$

where $W = \{w_{mj} \mid m = 1..\mathbf{d}, j = 1..\mathbf{N}\}$ are the mapping parameters and κ is the Gaussian RBF kernel (equation (2.12)). The maximisation of the likelihood (2.44) is then performed with respect to the mapping parameters, W , and the hyperparameters, Φ, using the posterior (2.51) with substituted (4.3). As a result, the learned model is composed of the dissimilarity preserving, probabilistic GP-LVM mapping from a latent to data space, and the local distance preserving mapping from a data to latent space referred to as back-constrained mapping.

During dimensionality reduction, BC-GPLVM takes only into account the local temporal ordering of time series. Although this constraint is not explicitly modelled, temporal sequences are mapped to smooth paths in a latent space, because consecutive points of time series tend to be similar. On the other hand, BC-GPLVM cannot handle any spatio-temporal relationships between different time series, thus the global topology of time series is ignored during dimensionality reduction. As a result, BC-GPLVM is not able to discover the unique time series pattern in the presence of time series repetitions, in particular when generated from different sources [Urtasun et al., 2008]. In addition, BC-GPLVM is computationally expensive, since the processing time grows cubically with the number of points in a dataset and linearly with the number of iterations in the optimisation process. Finally, BC-GPLVM has two free parameters, the inverse width of the back constrained mapping, which controls the smoothness of mapping [Lawrence and Quinonero-Candela, 2006], and the number of representative variables for the sparse approximation of covariance matrix (see section 2.2.2.3.2.2.2).

4.3.3. Gaussian Process Dynamical Model

Gaussian Process Dynamical Model [Wang et al., 2006, Wang et al., 2008] (GPDM), augments SGPLVM [Grochow et al., 2004] (see section 2.2.2.3.2.2.2) with a dynamical model in a latent space to model time series observations. It comprises the GPLVM based generative nonlinear mapping from a latent to data space (equation (2.24)):

$$y_i = f(x_i; A) + \varepsilon_{y,i} \tag{4.4}$$

and the nonlinear auto-regressive mapping on the latent space with first-order Markov dynamics:

$$x_i = h(x_{i-1}; B) + \varepsilon_{i,x} \tag{4.5}$$

where $\varepsilon_{y,i}$ and $\varepsilon_{i,x}$ denote zero-mean, white Gaussian noise processes, whereas f and h are nonlinear mappings parameterised by coefficients matrices A and B respectively. Both mappings are expressed by linear combinations of often nonlinear basis functions φ:

$$f(x; A) = \sum_i a_i \varphi_i(x) \tag{4.6}$$

$$h(x; B) = \sum_j b_j \varphi_j(x) \tag{4.7}$$

From a Bayesian perspective, the specific forms of functions f and h as well as the numbers of basis functions are incidental, and therefore should be marginalised out. With a zero mean and spherical Gaussian prior over the generative function parameters (equation (2.48)) and following the equation (2.49), the marginalisation over function f yields:

$$p(Y \mid X, \Phi_Y) = \prod_{j=1}^{\mathbf{D}} \mathcal{N}(y_j \mid 0, \Sigma_Y) = \frac{|W|^{\mathbf{N}}}{(2\pi)^{\mathbf{DN}/2} |\Sigma_Y|^{\mathbf{D}/2}} \exp(-\frac{tr(\Sigma_Y^{-1} Y W^2 Y^T)}{2}) \tag{4.8}$$

where the kernel matrix Σ_Y over all points is defined by the equation (2.50) with the following kernel hyperparameters $\Phi_Y = \{\alpha_Y, \sigma_Y^2, \gamma_Y\}$. The scaling matrix $W = diag\{w_1, w_2, ..., w_D\}$ accounts for different variances in the different data dimensions [Grochow et al., 2004, Wang et al., 2008].

Similarly, the complete joint likelihood over the latent coordinates is obtained by marginalisation over the dynamic function h:

$$p(X \mid \Phi_X) = \int p(X, h \mid \Phi_X) dh = \int p(X \mid h, \Phi_X) p(h \mid \Phi_X) dh \tag{4.9}$$

The incorporation of the first-order Markov dynamics (4.5) results in:

$$p(X \mid \Phi_X) = p(x_1) \int \prod_{i=2}^{\mathbf{N}} p(x_i \mid x_{i-1}, h, \Phi_X) p(h \mid \Phi_X) dh \tag{4.10}$$

Assuming a zero mean and spherical Gaussian prior over the generative function parameters B in each column, the above equation is simplified to:

$$p(X \mid \Phi_X) = p(x_1) \frac{1}{(2\pi)^{\mathbf{d}(\mathbf{N}-1)/2} |\Sigma_X|^{\mathbf{d}/2}} \exp(-\frac{tr(\Sigma_X^{-1} X_O X_O^T)}{2}) \tag{4.11}$$

where $X_O = \{x_i \mid i = 2..\mathbf{N}\}$. The kernel function of the matrix Σ_X is based on the equation (2.50) with the additional linear term and the following hyperparameters $\Phi_X = \{\alpha_X, \sigma_X^2, \gamma_X, \beta_X\}$ [Wang et al., 2008]:

$$\begin{aligned} k_{ij} &= \kappa(x_i, x_j) = \alpha_X \exp(\frac{\gamma_X}{2}(x_i - x_j)^T (x_i - x_j)) + \sigma_X^2 \delta_{ij} + \beta_X x_i^T x_j \\ \Sigma_X &= \{k_{ij} \mid i, j = 1..\mathbf{N}-1\} \end{aligned} \tag{4.12}$$

The learning process is performed using a two-stage maximum a posterior (MAP) estimation (see section 2.2.2.3.2.2.2) by maximising the likelihood (4.8) with respect to the latent positions, X, and all hyperparameters, using the following posterior:

$$p(X, \Phi_X, \Phi_Y, W \mid Y) \propto p(Y \mid X, \Phi_Y) p(X \mid \Phi_X) p(\Phi_Y) p(\Phi_X) p(W) \tag{4.13}$$

where uninformative priors are placed on the hyperparameters: $p(\Phi_Y) \propto \prod_i \Phi_{Yi}^{-1}$ and $p(\Phi_X) \propto \prod_i \Phi_{Xi}^{-1}$ to discorage overfitting. In turn, the broad half-normal prior is placed on W [Wang et al., 2008]. The maximisation of the above posterior is equivalent to minimising the negative log posterior of the model with respect to X, Φ_X, Φ_Y and W:

$$\begin{aligned} & L(X, \Phi_X, \Phi_Y, W) = -\ln p(X, \Phi_X, \Phi_Y, W \mid Y) = \\ & = \frac{1}{2}((\mathbf{DN}+1)\ln 2\pi + \mathbf{D} \ln|\Sigma_Y| + tr(\Sigma_Y^{-1} Y W^2 Y^T)) - \mathbf{N} \ln|W| \\ & + \frac{1}{2}((\mathbf{d}(\mathbf{N}-1)+1)\ln 2\pi + \mathbf{d} \ln|\Sigma_X| + tr(\Sigma_X^{-1} X_O X_O^T) + x_1^T x_1) \\ & + \sum_i \Phi_{Xi} + \sum_i \Phi_{Yi} + \frac{1}{2*10^6} tr\left(W^2\right) \end{aligned} \tag{4.14}$$

This optimisation process is performed numerically (see section 2.2.2.3.2.2.2).

The latent dynamical model favours preservation of local proximities between consecutive points, therefore a low dimensional representation respects the temporal continuity of time series data. On the other hand, similarly to BC-GPLVM, GPDM cannot model any spatio-temporal relations between different time series, thus the global topology of time series does not constrain the dimensionality reduction process. As a consequence, GPDM cannot produce the unique time series representation in the presence of time series repetitions, in particular when generated from different sources [Urtasun et al., 2008]. Moreover, the inclusion of the latent dynamical model in the learning process results in a further cubical increase of the processing time in comparison to BC-GPLVM. Finally, GPDM has one free parameter, i.e. the number of representative variables for the sparse approximation of covariance matrix (see section 2.2.2.3.2.2.2).

4.4. Proposed Methodology

We introduce a novel parameterless nonlinear dimensionality reduction method to process efficiently multidimensional time series data. The Temporal Laplacian Eigenmaps is a powerful extension of the standard LE framework, which aims at preserving the temporal structure of the data manifold instead of its local geometry as basic LE does. This is achieved by extensively exploiting the key property of standard LE framework of preserving approximated distances between neighbourhood points in a low dimensional space. In principle, points closeness in the embedded space can be flexibly controlled by creating connections between corresponding high dimensional features in the Laplacian graph. Therefore, powerful temporal constraints are introduced based on the innovative concept of temporal neighbourhoods. These constraints encapsulate effectively spatio-temporal dependencies of time series, in particular when issued from various sources. We propose to construct two complementary graphs from these temporal neighbourhoods which are simultaneously exploited to constrain an optimisation process. As a consequence, the proximity of points in the embedded space is governed by these graphs. This allows respecting the temporal consistency of each time series as well as modelling spatio-temporal similarity of time series repetitions during dimensionality reduction even in the presence of significant data variations.

In this section, first, the standard Laplacian Eigenmaps method is comprehensively described (section 4.4.1). This is fundamental before introducing the proposed algorithm in section 4.4.2.

4.4.1. Background of Laplacian Eigenmaps

Laplacian Eigenmaps [Belkin and Niyogi, 2002, Belkin and Niyogi, 2003] is a nonlinear and unsupervised geometrically motivated dimensionality reduction method which is based on a simple intuition that nearby high dimensional input features should be mapped to nearby low dimensional output points. As a result, the algorithm aims at faithfully preserving locality structure of high dimensional data (Figure 4.2).

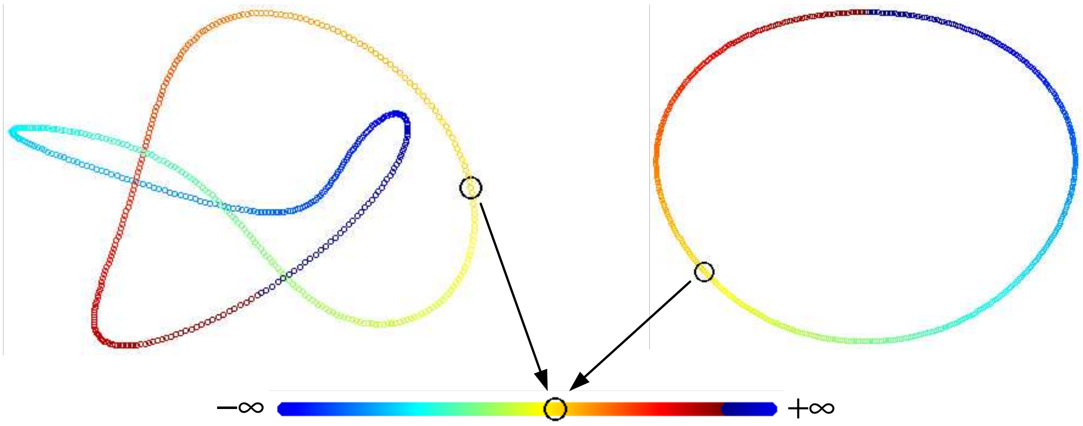


Figure 4.2. Laplacian Eigenmap aims at maintaining the local properties of high dimensional data.

4.4.1.1. Justification

The mathematical justification for the Laplacian Eigenmap actually involves an interconnection between several areas of mathematics such as differential geometry, spectral graph theory, partial differential equations and linear algebra.

In principle, in the manifold learning setting, the underlying manifold is usually unknown. Therefore the functional form of the manifold need to be estimated using a high dimensional cloud of data points. In the case of a compact infinitely differentiable manifold, such functional map applied on underlying manifold is given by the Laplace Beltrami operator [Belkin and Niyogi, 2002, Belkin and Niyogi, 2003, Zheng, 2008]. The Laplace Beltrami operator is a positive semi-definite self-adjoint operator and has a discrete spectrum on a compact manifold. It has been shown that the optimal embedding of a high dimensional manifold is equivalent to finding the nonzero eigenvalues as well as its corresponding eigenvectors associated with the Laplace Beltrami operator [Belkin and Niyogi, 2002, Belkin and Niyogi, 2003, Zheng, 2008].

As a consequence, the objective of manifold learning is to compute the Laplace Beltrami operator on a continuous manifold. The problem is very challenging, since only a sparse cloud of high dimensional points is available. However, the Laplace Beltrami operator can be discretely approximated using heat diffusion equations on the Laplacian graph as it is presented in [Belkin and Niyogi, 2002, Belkin and Niyogi, 2003, Zheng, 2008]. Based on spectral graph theory, such graph can then be represented as a local similarity matrix which reflects the degree to which points are near to one another. Spectral decomposition of this Laplacian matrix reveals the low dimensional structure of the underlying manifold. The convergence of eigenvectors of graph Laplacian associated to a point cloud dataset to eigenfunctions of the Laplace-Beltrami operator on continuous manifold has been proved mathematically [Belkin and Niyogi, 2007].

4.4.1.2. Algorithm

The structure of Laplacian Eigenmaps is presented in the Figure 4.3 and was briefly described in section 2.2.2.2.2.2. Here, we investigate the algorithm in more details.

Initially, the adjacency graph is constructed by putting an edge between nodes i and j if high dimensional points y_i and y_j are 'close' based on the K-nearest neighbours or hyper sphere neighbourhood procedure (see section 2.2.2.2.2.2.1 for details). Then, weights are assigned to the edges of the graph to express the geometrical relationship between corresponding points using the following heat kernel:

$$w_{ij} = \begin{cases} \exp(-\left\| y_i - y_j \right\|^2) & \text{if i and j connected} \\ 0 & \text{otherwise} \end{cases} \tag{4.15}$$

The optimal embedding is discovered by minimizing the following objective function (see also section 2.2.2.2.2.2.3):

$$\varepsilon = \frac{1}{2}\sum_{i,j=1}^{\mathbf{N}} \left| x_i - x_j \right|^2 w_{ij} = X^T LX \tag{4.16}$$

subject to constraint to remove an arbitrary scaling factor in the embedding:

$$x^T Mx = 1 \tag{4.17}$$

and constraint to remove translation invariance (to centre on the origin):

$$x^T M1 = 0 \tag{4.18}$$

Matrix M is a diagonal weight matrix with elements $m_{ii} = \sum_{j=1}^{\mathbf{N}} w_{ij}$. This matrix is interpreted as a measure of the empirical density of points around y_i (the degree of vertex importance). In turn, the Laplacian graph is given by a sparse semi definite positive matrix:

$$L = M - W \tag{4.19}$$

The square matrix L is symmetric and real, hence it is Hermitian, i.e. $L = L^*$, where $*$ denotes the conjugate transpose of the matrix. This is a sufficient condition to apply a version of the Rayleigh-Ritz theorem [Ledermann and Vajda, 1961], which characterises eigenvalues of Hermitian matrices as the solutions of a series of optimisation problems [Horn and Johnson, 1985]:

$$\begin{aligned}
&\lambda_1 = \underset{X^T MX = I}{\arg\min}\, X^T LX \\
&\lambda_2 = \underset{\substack{X^T MX = I \\ X \perp X_1}}{\arg\min}\, X^T LX \\
&\lambda_3 = \underset{\substack{X^T MX = I \\ X \perp X_1 \perp X_2}}{\arg\min}\, X^T LX \\
&\ldots \\
&\lambda_{\mathbf{D}} = \underset{\substack{X^T MX = I \\ X \perp X_1 \perp \ldots \perp X_D}}{\arg\min}\, X^T LX
\end{aligned} \tag{4.20}$$

where $0 = \boldsymbol{\lambda}_{\mathbf{1}} \le \boldsymbol{\lambda}_{\mathbf{2}} \le .... \le \boldsymbol{\lambda}_{\mathbf{D}}$ are the eigenvalues of L , whereas $x_1, x_2, ..., x_{\mathbf{D}}$ the corresponding eigenvectors. Minimisation of equations (4.20) subject to $X^T MX = I$ (from (4.17) and (4.18)) is equivalent to the solution of generalised eigenvalue problem in the form (see also section 2.2.2.2.2.2.4) [Belkin and Niyogi, 2002, Belkin and Niyogi, 2003]:

$$LX = \lambda MX \tag{4.21}$$

where the obtained eigenvectors ordered according to their eigenvalues satisfy:

$$\begin{aligned}
&Lx_1 = \boldsymbol{\lambda}_{\mathbf{1}} Mx_1 \\
&Lx_2 = \boldsymbol{\lambda}_{\mathbf{2}} Mx_2 \\
&\ldots \\
&Lx_D = \boldsymbol{\lambda}_{\mathbf{D}} Mx_D
\end{aligned} \tag{4.22}$$

The final low dimensional coordinates X are given by $\mathbf{d}$ eigenvectors which correspond to the $\mathbf{d}$ smallest nonzero eigenvalues. Note the bottom $\mathbf{d}+1$ eigenvectors of L can be determined without performing a full matrix decomposition [Bai et al., 2000]. Moreover, the matrix L is extremely sparse, which results in substantial computational savings for large training sets.

In the cost function (4.16), a large weight w_{ij} corresponds to small distance between the data pairs y_i and y_j according to equation (4.15). Hence, the difference between their low dimensional representations x_i and x_j highly contributes to the objective function (4.16). As a consequence, nearby points in the high dimensional space are brought closer together in the low dimensional representation, so that local neighbourhood relations are correctly preserved by LE.

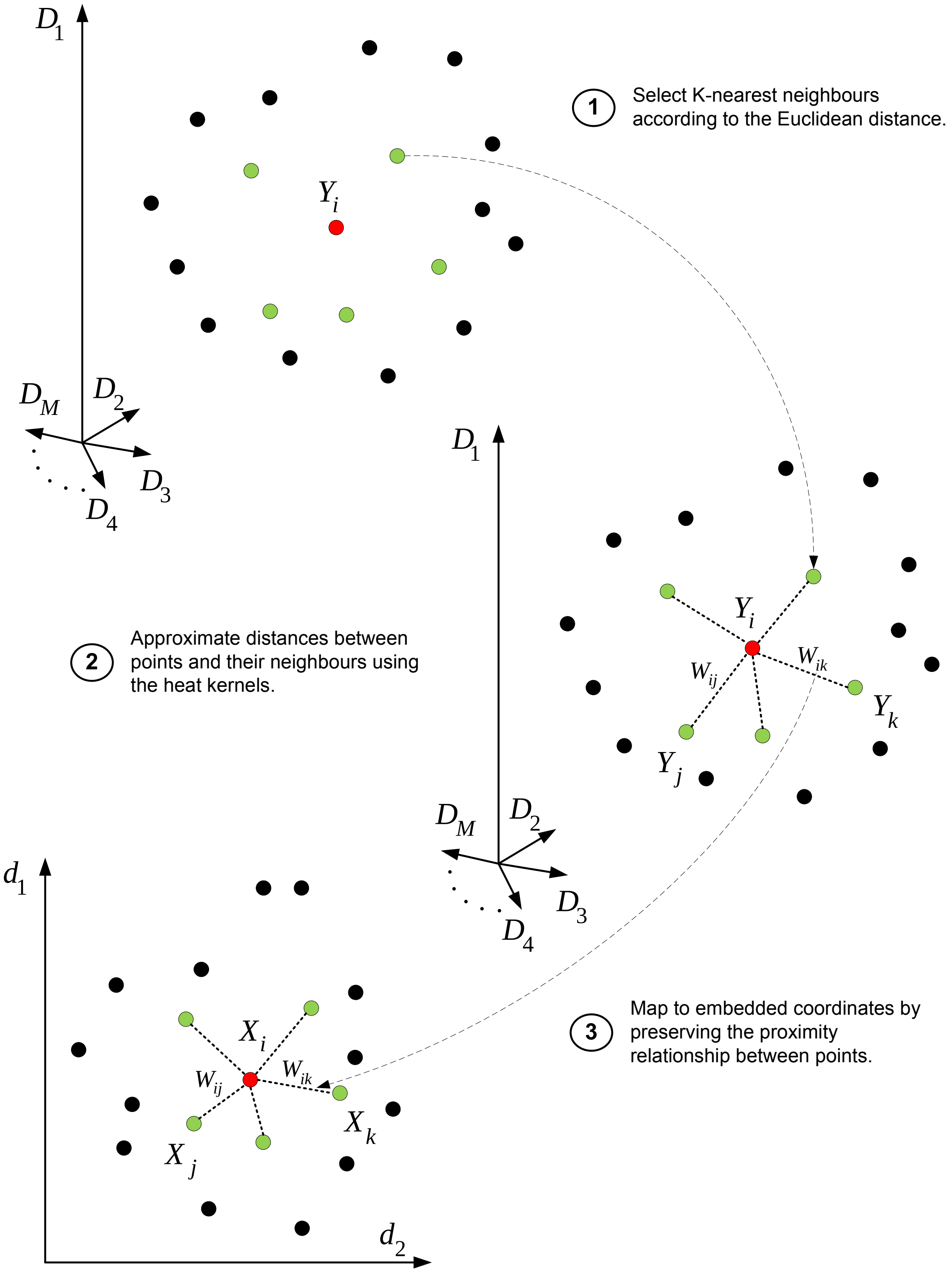

D_1
D_M
D_2
D_3
D_4
Y_i
1
Select K-nearest neighbours according to the Euclidean distance.
2
Approximate distances between points and their neighbours using the heat kernels.
Y_i
W_{ij}
W_{ik}
Y_k
Y_j
3
Map to embedded coordinates by preserving the proximity relationship between points.
d_1
d_2
X_i
W_{ij}
W_{ik}
X_k
X_j

Figure 4.3. Successive steps of standard Laplacian Eigenmaps.

4.4.2. Temporal Laplacian Eigenmaps

The proposed Temporal Laplacian Eigenmaps algorithm shares the processing steps with the embedding based approaches (section 2.2.2.2.2.2 and Figure 2.12). However, there are fundamental differences in the foundations of the algorithm. First, a neighbourhood for each data point is obtained automatically using the novel concept of temporal neighbours. Then, two sparse complementary graphs are assembled to encapsulate temporal constraints and employed in an extended optimisation process to discover embedding of high dimensional time series.

4.4.2.1. Construction of Temporal Neighbourhoods

The temporal similarity between data points is maintained implicitly during dimensionality reduction by building new types of neighbourhoods which express temporal dependencies. Since, temporal neighbours are placed nearby in the embedded space; there is no need to enforce any artificial constraints as in the ST-Isomap framework. Two types of temporal neighbourhoods are proposed for each data point P_i:

- Temporal neighbours (T): the $2\mathbf{m}$ closest points in the sequential order of input (Figure 4.4a):

$$T_i = \{P_{i-m},...,P_{i-1},P_i,P_{i+1},...,P_{i+m}\} \tag{4.23}$$

- Spatio-temporal repetition neighbours (S): let's associate to each point, P_i, $2\mathbf{s}$ temporal neighbours which define a time series fragment F_i. The repetition neighbours, S_i, of P_i are the centres of the q_i time series fragments, $F_{i,k}$, which are similar to F_i (Figure 4.4b):

$$S_i = \{F_{i,1}(C),...,F_{i,q_i}(C)\} \tag{4.24}$$

  where $F_{i,k}(C)$ returns the centre point of $F_{i,k}$. Note that by design repetition neighbours are assumed to be extracted from different repetitions of the current

MTS fragment. These repetitions of the same phenomenon are generated by either the same source or different sources.

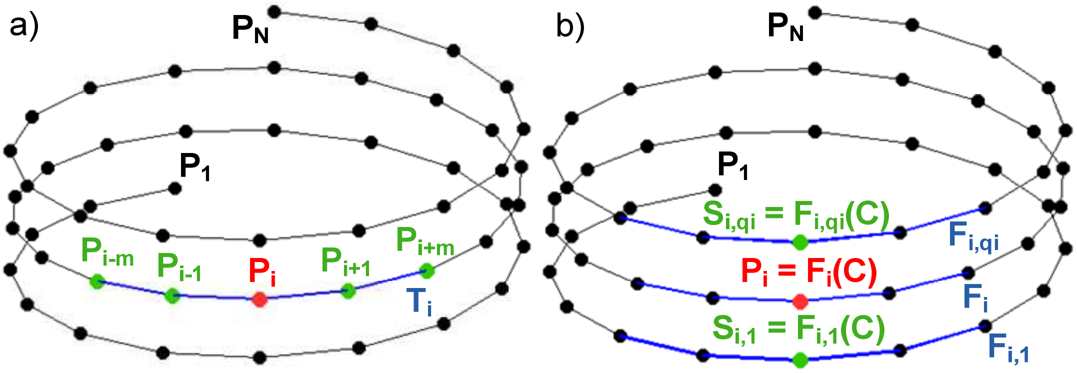


Figure 4.4. Temporal (a) and spatio-temporal repetition (b) neighbours (green dots) of a given data point, Pi, (red dots).

The selection of the $2m$ adjacent neighbours is straightforward since it is based on the data temporal order (equation (4.23)). In practise, this parameter is set to one ($\mathbf{m} = 1$) to model the first-order Markov dependency between consecutive points in time series. The size of the repetition neighbourhood q_i corresponds to the number of times a state is repeated in the training set. The optimal repetition neighbourhood size as well as a selection of these neighbours is automatically determined using the following procedure (Figure 4.5 and Figure 4.6):

1. Associate to each data point, P_i, $2\mathbf{s}$ adjacent temporal neighbours to create the local fragment, F_i, centred on P_i.
2. Calculate similarity between the local fragment F_i and fragments created by sliding a warping window through the entire training set. The similarity between fragments is measured with the DTW metric (see appendix A.1) and stored in a neighbourhood similarity matrix M of size $\mathbf{N} \times \mathbf{N}$. The pair wise similarity of points during computation of DTW can be measured using any distance, in particular Manhattan, Euclidean or Hausdorff (see appendix A.3) metric. If it is not explicitly

stated otherwise, the standard Euclidean distance is used in an evaluation process.

3. Perform temporal windowing of the similarity matrix M by applying a moving average filter on distances between fragments using a history window of size $2\mathbf{s}$:

$$a_{i,j} = \frac{1}{2\mathbf{s}} \sum_{b=0}^{2\mathbf{s}-1} m_{i-b,j-b} \qquad (4.25)$$

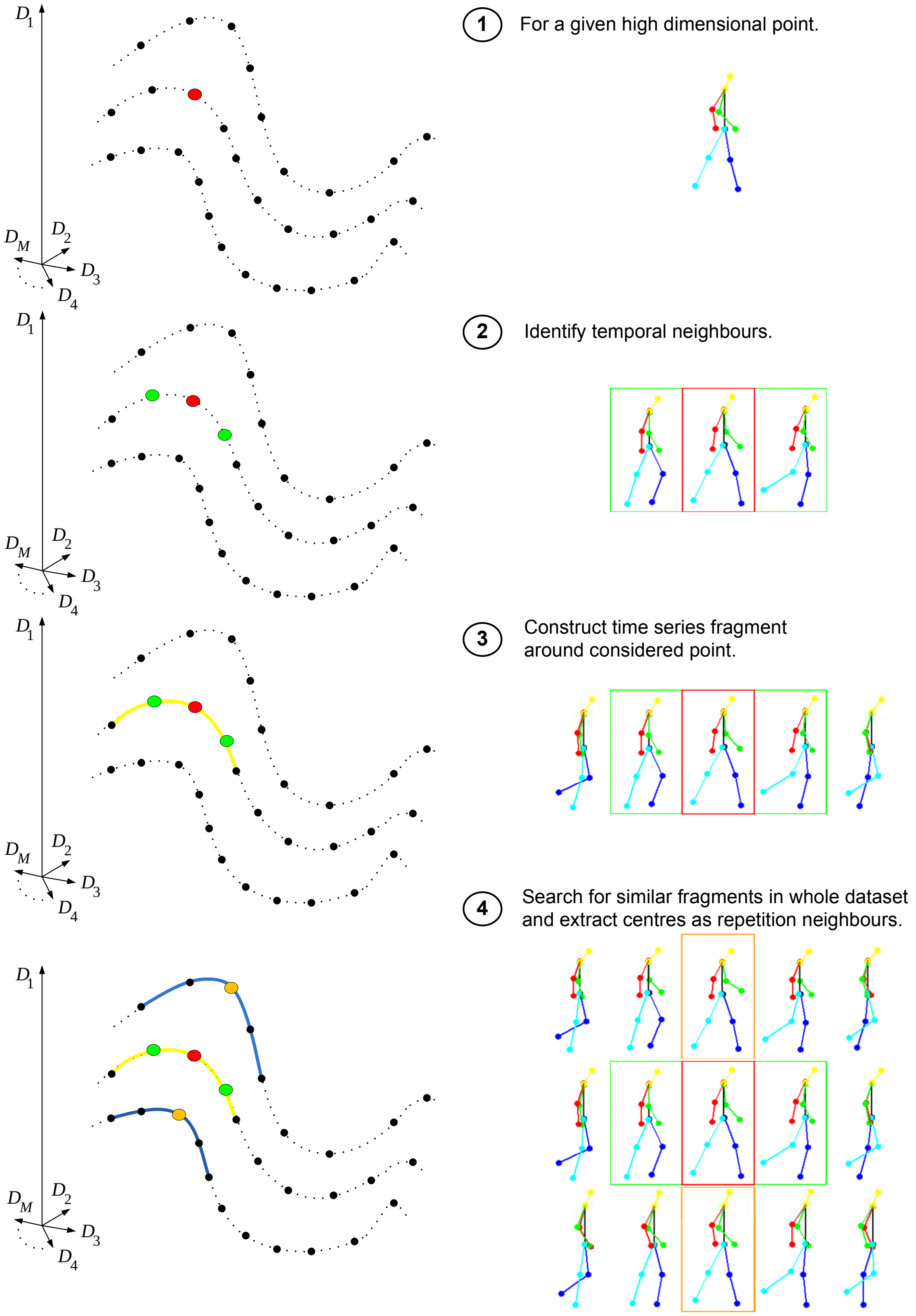
4. For each data point P_i, search for similar fragments, $F_{i,k}$, defined by a similarity greater than $\mathbf{b} = 0.75$ standard deviations σ_i from the mean μ_i in the row i of neighbourhood similarity matrix A:

$$k = \{ j \mid a_{i,j} < \mu_i - \mathbf{b}\sigma_i \} \qquad (4.26)$$

   The value of $\mathbf{b}$ was set a priori in all conducted experiments to represent a difference of 1.5 standard deviations between consecutive local extremas. Vast range of performed experiments using different datasets (see section 4.5) suggests that TLE is not sensitive to this value.

5. Extract from each similar fragment, $F_{i,k}$, the data point which corresponds to P_i, i.e. the centre of $F_{i,k}$. The extracted points define P_i's temporal repetition neighbourhood.

This procedure takes into account a spatio-temporal similarity between different time series to identify the optimal repetition neighbours. To facilitate this process and obtain the best possible results, a high level representation of data should change smoothly in local regions with the order of the input. This assumption is valid for time series, since successive data points in time series tend to be very similar, and likewise in the corresponding high level data representations. As a consequence, TLE can take advantage of any sequentially ordered input data to identify spatio-temporal dependencies by just pre-processing raw data into an appropriate domain specific feature representation.



For a given high dimensional point.

Identify temporal neighbours.

Construct time series fragment around considered point.

Search for similar fragments in whole dataset and extract centres as repetition neighbours.

Figure 4.5. Selection example of temporal (green) and repetition (orange) neighbours in the temporal Laplacian Eigenmaps using MoCap data (D=52).

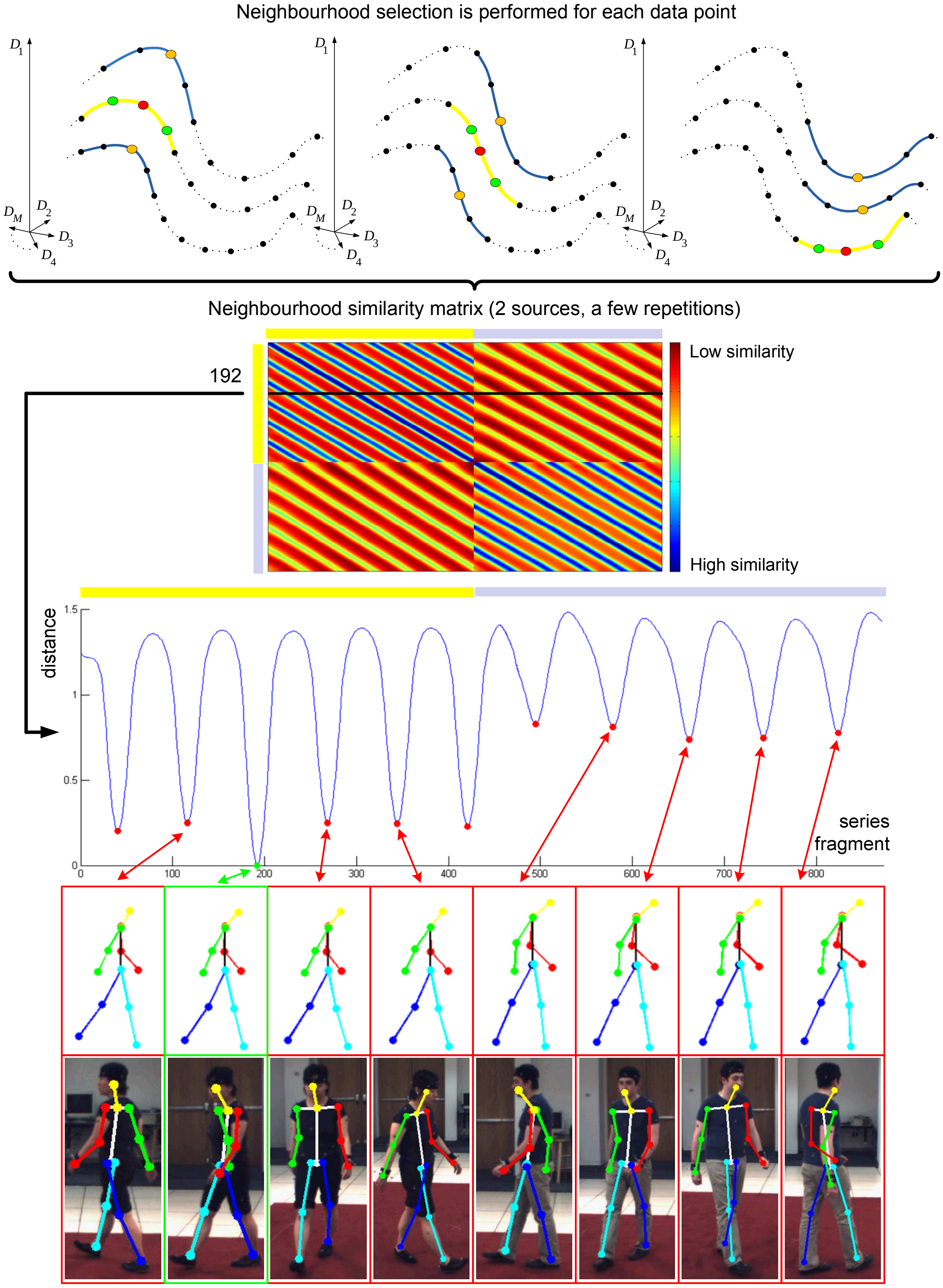


Figure 4.6. Example of neighbourhood similarity matrix created by the TLE using two sources of MoCap data with a few repetitions each. Each local minimum corresponds to the most similar repetition neighbour in relation to the reference pose (green) extracted from different repetitions of the time series.

The introduced procedure has one free parameter *s* which defines the length of the time series fragment used during DTW comparisons. However, the choice of this parameter is extremely simple and it is not critical, since the neighbourhood selection schema is quite insensitive to its value. First, assuming some basic prior knowledge about the dataset of interest, the upper value bound of this parameter could be estimated since it should not exceed the lengths of time series repetitions in the dataset. Secondly, the parameter value should be sufficient large to express 'satisfactory' local curvature of time series fragments. Although, the determination of the lower bound depends on the dataset and may be problematic, in practice, it does not have to be performed.

In Figure 4.7, we present the percentage of correctly indentified repetition neighbours in relation to the values of the parameter **s** for human MoCap data. The depicted graph shows clearly a plateau for a very large range of parameter values. Performance in the right area of graph converges towards zero when **s** reaches the lengths of the time series repetitions in this dataset. As seen in Figure 4.7 any value between 9 and 72 is equally appropriate in terms of accuracy. Thus, in practice, by taking a conservative approach, the value can be set rougly as a half of any repetition length. However, when exact knowledge about lengths of repetitions is not available, the value can be intuitively suggested by a user. Note that due to computional cost of DTW alignment, the smaller values of parameter **s** are advantageous when the processing time is an issue. In this research, we advise a default value of 10 for this parameter, which proves to work satisfactory for all conducted experiments whatever the dataset.

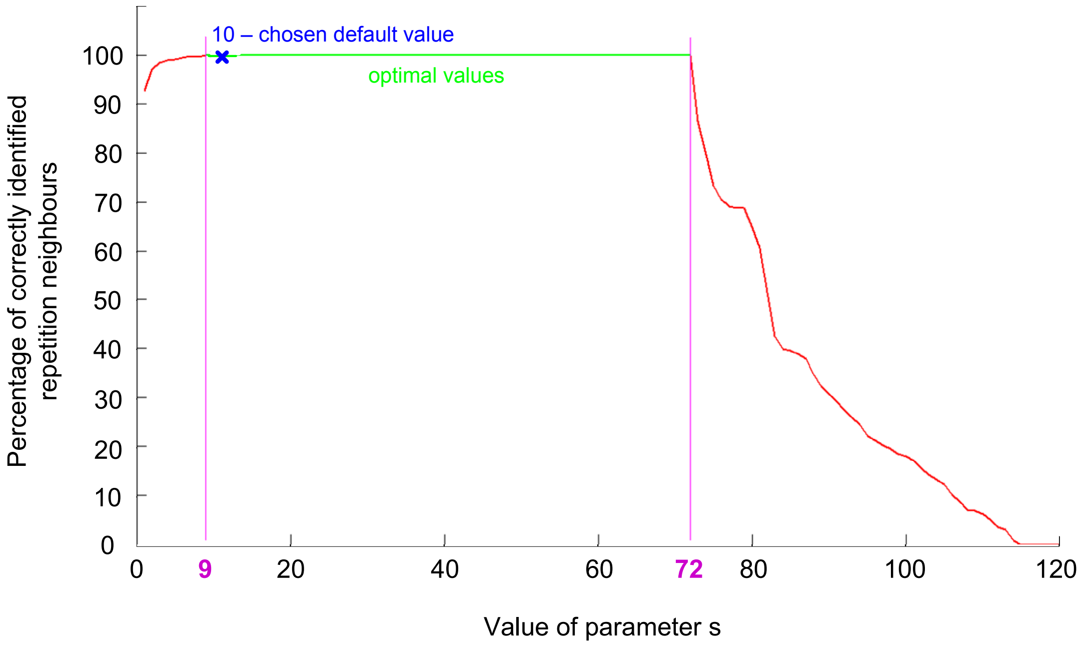


Figure 4.7. The relation between the percentage of correctly identified repetition neighbours and values of parameter s in the proposed neibhourhood selection procedure.

4.4.2.2. Graphs Assembling

The temporal neighbour relations are used in the construction of two temporal graphs $G = \{T, S\}$, where any two vertices are connected only when some temporal relation exists between these points. Weights W are assigned to edges of each graph separately using the standard LE formulation (4.15):

$$w_{ij}^{G} = \begin{cases} \exp(-\left\|y_i - y_j\right\|^2) & \text{if i and j are temporally correlated} \\ 0 & \text{otherwise} \end{cases} \qquad (4.27)$$

The difference between the standard LE graph, which was obtained with the K-nearest neighbour procedure, and our temporal graphs is illustrated in Figure 4.8. In the case of standard LE, the temporal structure is not reflected in the graph when time series are generated from a single source (Figure 4.8a). Similarly, for time series issued from different sources, not only the temporal structure is lost but each time series is modelled almost separately, which may result in creating disjoints graphs when variations between time series are large (Figure 4.8c). In contrast, our combined temporal graphs are capable to represent appropriately the temporal structure of the data in both cases (Figure 4.8b and Figure 4.8d), thus, they encode more powerful constraints for the dimensionality reduction process.

Neighbourhood connections defined in the Laplacian graphs implicitly impose points closeness in the embedded space. Consequently, the temporal neighbours allow modelling a first-order Markov dependency of time series into the resulting embedding, whereas repetitions neighbours remove style variability by aligning time series in the embedded space.

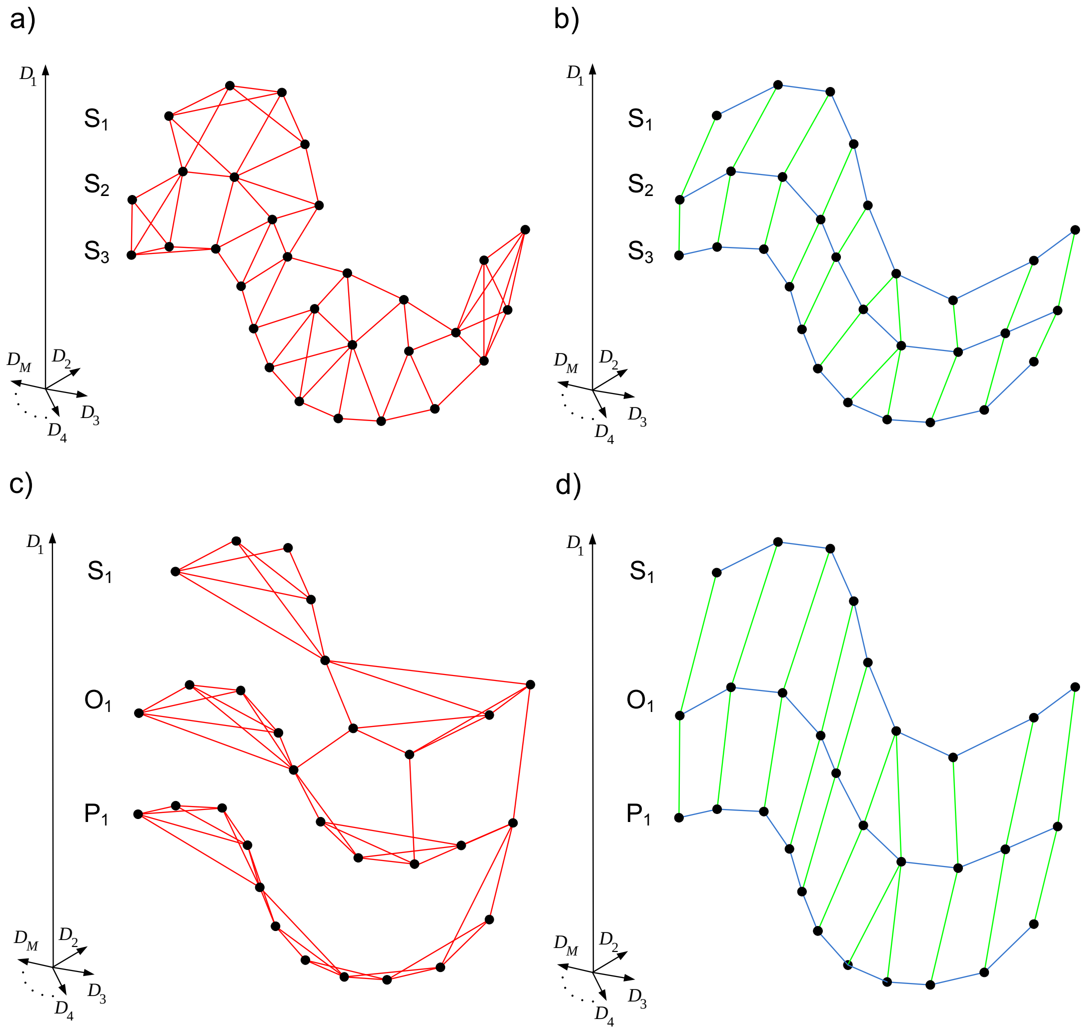


Figure 4.8. Graphs constructed by standard Laplacian Eigenmaps (left – red colour denotes K-nearest neigbhours) and proposed Temporal Laplacian Eigenmaps (right – blue colour denotes temporal neighbours, whereas green colour illustrates repetition neighbourhoods) for: a,b) time series issued from a single source, c,d) time series issued from different sources.

4.4.2.3. Optimisation Process

Following the standard LE formulation, we introduce an extended cost function to combine information from both graphs:

$$\varepsilon = \frac{1}{2}\sum_{i,j=1}^{N}\left|x_i - x_j\right|^2 w_T^{ij} + \frac{1}{2}\sum_{i,j=1}^{N}\left|x_i - x_j\right|^2 w_S^{ij} = X^T L_T X + X^T L_S X \qquad (4.28)$$

The objective of dimensionality reduction process is to minimise the above equation with respect to the embedded coordinates X subject to constraints:

$$\operatorname{argmin}_X \quad X^T L_T X + X^T L_S X \tag{4.29}$$

$$\text{subject to } X^T M_T X + X^T M_S X = I \tag{4.30}$$

where $M_G = diag\{m_{11}^G, m_{22}^G, ..., m_{nn}^G\}$ is a diagonal matrix with entries: $m_{ii}^G = \sum_{j=1}^{\mathbf{N}} w_G^{ij}$, and $L_G = M_G - W_G$ is the Laplacian matrix. The minimum of the objective function is found by applying Lagrange multipliers [Mizrahi and Sullivan, 1990] to equation (4.29) subject to the constraint expressed by equation (4.30):

$$\wedge(X,\lambda) = X^T(L_T + L_S)X + \lambda(I - X^T(M_T + M_S)X) \tag{4.31}$$

$$\frac{\partial \wedge (X,\lambda)}{\partial X} = (L_T + L_S)X - \lambda(M_T + M_S)X = 0 \tag{4.32}$$

$$(L_T + L_S)X = \lambda(M_T + M_S)X \tag{4.33}$$

The solution of minimisation problem is given by the embedded space X which is spanned by the eigenvectors which correspond to the **d** smallest nonzero eigenvalues λ obtained by the solution of the sparse generalised eigenvalue problem (4.33) [Arnoldi, 1951, Fokkema et al., 1999, Knyazev, 2002] based on the generalisation of the Rayleigh-Ritz theorem [Horn and Johnson, 1985] (see section 4.4.1.2).

4.4.2.4. Summary

Since temporal relationship is a local property of data, TLE can be conceptually classified as a local nonlinear dimensionality reduction method similarly to the standard LE framework. However, whereas the latter focuses at preserving only local geometry, the former aims at maintaining the local temporal structure of high dimensional data, which implicitly tends to the preservation of the global temporal topology of the data manifold. As a result, our approach is able to extract a common temporal pattern and generate a distinctive data-driven representation of multivariate time series regardless of stylistic variations. Therefore, it is particularly suitable for time series data which include data repetition; otherwise, when applied on a single time series, it often behaves similarly to the standard LE.

Note that, style variability is actually not completely removed from the low dimensional representation but only considerably marginalised during the dimensionality reduction process. This is expected since according to equation (4.27), small distances between successive temporal neighbours result in corresponding large weights in a cost matrix and thus high contributions to the objective function (4.28). In contrast, weights between relatively distant spatio-temporal repetition neighbours, which are responsible for expressing stylistic variability, are significantly smaller and thus proportionally less important. As a consequence, the temporal constraints dominate over spatio-temporal ones and thus allow the discovery of the unique low dimensional pattern of time series. Finally, thank to the sparse generalised eigenvalue problem, our method is computationally very efficient and guarantees globally optimal analytical solution in a non iterative manner.

As we will show in the evaluation section (4.5), our method is superior to ST-Isomap and other time-series-oriented dimensionality reduction methods including BC-GPLVM and GPDM in the terms of efficiency and quality of produced embedded spaces.

4.5. Evaluation

The proposed method is validated with both artificial and real datasets. Different types of multidimensional series are chosen to extensively evaluate the performance and robustness of the proposed methodology in different scenarios.

While we have implied that MTS are ordered along the time dimension, in practice TLE is a versatile framework and without modifying the algorithm core, it can be used for any MTS, no matter how they are ordered as long as the markovian property is preserved. This is achieved by simply pre-processing raw data to a feature representation which changes smoothly with the order of input. Such representation facilitates the selection procedure of repetition neighbourhoods using the DTW metric (see section 4.4.2.1).

First, all datasets used in this evaluation are introduced in section 4.5.1. Then, the setup of experiments is explained in section 4.5.2.1 followed by a description of performed experiments in section 4.5.2.2. Subsequently, the flexibility of TLE framework is demonstrated by modelling the temporal structure of motion capture data (section 4.5.5), raw videos (sections 4.5.7 and 6.4) and even the sequential change of the camera perspective in images (section 4.5.4). A practical application of the proposed methodology, i.e. view dependent action recognition, is presented in section 4.5.7. Finally, a broad discussion about obtained results is provided in section 4.5.8.

4.5.1. Datasets

Figure 2.28, Figure 3.5 and Figure 4.9 illustrate the datasets used for evaluation.

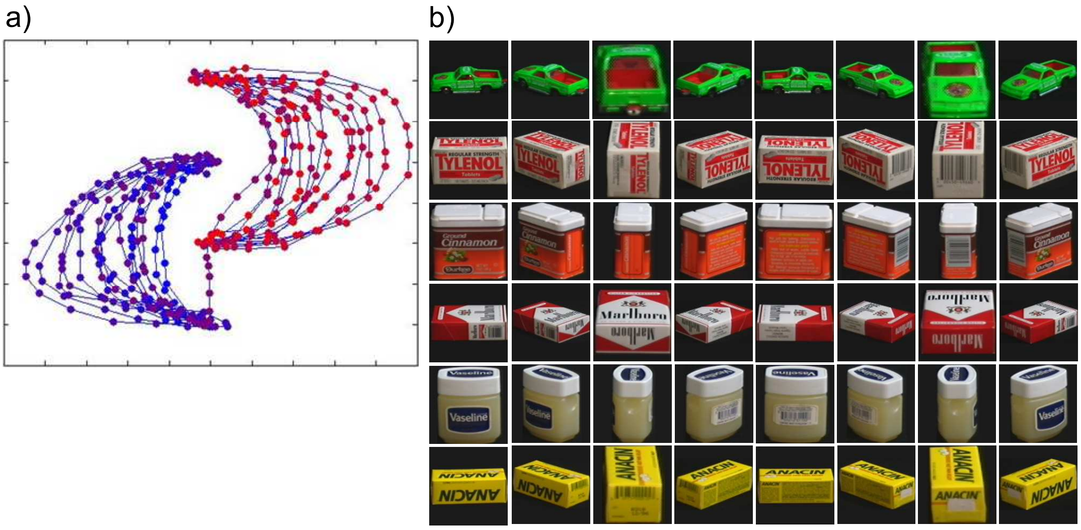


Figure 4.9. Datasets used in the qualitative experiments: a) the mouse movement dataset and b) the 6 selected representative objects seen from different view angles (every 45 degrees) from the Columbia Image Library.

The “two moons” dataset [Zhou et al., 2003] is a record of two successive sets of vertical mouse movements each forming a moon shape with a transition between them (Figure 4.9a). This toy problem was introduced to evaluate spatio-temporal properties of dimensionality reduction methods [Zhou et al., 2003, Jenkins and Mataric, 2004]. Although the input dimensionality of data is 2, the dataset is intrinsically a 3-dimensional [Jenkins and Mataric, 2004], since two distinct spatial motions are expected to be modelled separately with respect to the temporal cycle of the movement. This dataset consists of a single MTS source with a number of motion repetitions.

The Columbia Object Image Library (COIL) is a database of colour images of 100 objects. These objects were captured from 72 views against a uniform black background by a fixed camera (every 5 degrees) [Nene et al., 1996]. All images were normalised to the size 128×128. In our experiment, 6 representative objects were chosen according to a rough visual similarity of the global shape, although they significantly differ in appearance (objects: 27.car, 31.box_1, 39.container, 46.cigarette_packet, 55.jar, 79.box_2, see Figure 4.9b). The sequential change of the object shape along the view circle can be considered as a multidimensional series, which we call a multidimensional view series (MVS). In such case, each object is considered to be a different source of MVS. In this dataset, two repetitions of each source are available in the ranges $<0°,180°)$ and $<180°,360°)$ respectively. From a geometrical point of view, the second range is the repetition of the first one with the front-back inversion of the appearance (Figure 4.12c). All these images are expected to reside on a 1-dimensional manifold, since there is only one intrinsic dimension, i.e. the object orientation. However, in our experiment, this intrinsic structure is embedded into a 2-dimensional space to take into account the cyclic nature of the view change. As a result, an ideal visual low dimensional representation of this dataset is a unique circle pattern for all objects, since the view dimension is shared between all of them. This circle is expected to be parameterised by only one parameter, i.e. the view change (see Figure 2.9a).

The HumanEva (HE) dataset has been introduced in section 2.3.2.3. In this evaluation only sequences of “walking and jogging in a circle” are processed using the actors depicted in Figure 3.5. Training was performed using the longest available continuous sequence of valid MoCap poses for each subject as presented in Table 4.1. The walking and jogging actions were chosen, since their intrinsic dimensionality as well as their underlying manifold structure is well known and conceptually easy to justify. Both actions are cyclic, since the intrinsic joint configuration of human body recurs every two steps with both legs. Therefore, two successive steps are considered to be a single MTS which is repeated several of times in the action. Intuitively, any two steps correspond to a continuous curve in a human motion space, since there is only one degree of freedom, i.e. the innate state/configuration of the motion over time. As a consequence, the intrinsic structure of both actions is a 1-dimensional manifold embedded into a 2-dimensional space to model the nonlinearity and cyclic nature of the action. A natural visual low dimensional representation of this struture is a smooth closed 2-dimensional curve. It is worth to point out that both steps are usually highly symmetric, since the intrinsic configuration of joints is roughly the same for opposite limbs. For that reason, in the ideal case, the 2-dimensional curve representation is expected to have an axis of symmetry, such as an ellipse, where each half represents one step in the cycle. Finally, all subjects (i.e. sources of MTS) are expected to be modelled jointly along a unique ellipse like pattern, since the intrinsic content/configuration of the motion is the same, despite of the style variability in the execution. This ellipse is supposed to be parameterised by only one parameter, i.e. the intrinsic change of the body configuration (see Figure 2.9a). This analysis is in agreement with the value of intrinsic dimensionality determined by EE (see section 3.2.1.1) and consistent with other research on modelling walking and jogging actions [Grochow et al., 2004, Elgammal and Lee, 2004a, Urtasun et al., 2006a, Darby et al., 2010].

Table 4.1. The summary of frames which are used in a learning process.

Action	Name	Frames	Number of frames
Walking	Subject 1 in trial 3	16-1484	1468
	Subject 2 in trial 3	55-1498	1443
	Subject 3 in trial 3	52-1172	1120
Jogging	Subject 1 in trial 1	6-252	246
	Subject 2 in trial 1	6-795	789
	Subject 3 in trial 1	5-775	770

The Weizmann dataset has been introduced in section 2.3.3.4.1. All 9 subjects and 10 actions are used for the evaluation (a few examples are illustrated in Figure 2.28). Actions were manually segmented into 240 instances of primitive motions. Each atomic action is a single MTS, whereas each actor is regarded as a different MTS source. The intrinsic dimensionality of the dataset is 2 as evaluated by [Blackburn and Ribeiro, 2007].

4.5.2. Experimental Framework

The proposed algorithm is evaluated through qualitative and quantitative analyses of performance. Results are compared with those produced by standard dimension reduction methods, i.e. LE, Isomap and BC-GPLVM, and their respective improved temporal versions, i.e. TLE, ST-Isomap and GPDM.

4.5.2.1. Setup

In order to evaluate embedding-based methods quantitatively a mapping function is required which allows projecting data between high and low dimensional spaces. The RBFN mapping was trained from a low to high dimensional space and then inverted as described in section 2.2.2.4.4. The Gaussian basis function (equation (2.60)) is exploited in the learning process, because of its excellent approximation properties [Poggio and Girosi, 1990]. Note that our graph based extension of RBFN, which has been proposed in section 3.3.2, is not compatible with TLE. In contrast to standard spectral methods (see Figure 4.8a,c), the structure of TLE graph is very regular and uniformly spread across training data (see Figure 4.8b,d). As a consequence, it is not feasible to simulate neither strong nor weak flows in such graph to determine desired sub graphs, which become the cluster centres. For that reason, the standard RBF network is learned in all experiments.

Unlike TLE and mapping-based approaches, all other embedding-based methods require manual parameter tuning. In this study, we used the default parameters provided with the Matlab implementations of BC-GPLVM [Lawrence and Quinonero-Candela, 2006] and GPDM [Wang et al., 2006]. In the case of spectral methods, extensive testing was conducted to determine the optimal settings for each experiment. In addition, the number of nontrivial neighbours required for ST-Isomap [Jenkins and Mataric, 2004] was calculated using the TLE estimation procedure when appropriate.

4.5.2.2. Experiments

First, we evaluate qualitatively our novel algorithm using two datasets for which the underlying structure is known so that the quality of the embedded space can be judged visually. Initially, we compare TLE against the most similar approach, i.e. ST-Isomap using the synthetic dataset of mouse motion (section 4.5.3). Afterwards, TLE and all baseline methods are evaluated in a more demanding experiment using a very high dimensional image dataset. Here, we show that TLE can take advantage of any sequential series of observations as long as the selection procedure of repetition neighbourhoods is feasible. The objective of this experiment is to discover a compact 1-dimensional joint view manifold of the 6 different objects, where multivariate series corresponds to a sequential change of camera perspective for an object (section 4.5.4).

Secondly, the quantitative and further qualitative comparison of TLE against state of the art approaches is provided in section 4.5.5. Here, time series are human motions which are represented by motion capture data as described in section 2.3.2.3. In order to make the quantitative comparison possible, the 3D pose refinement framework presented in section 3.4.5.2 is simplified. In this experiment, we consider three different subjects performing two actions (walking and jogging). In order to provide an exhaustive evaluation, around 6000 pose estimates in total are simulated by introducing a Gaussian noise to ground truth poses with an average error per joint of 80mm. This error corresponds to the average error of 3D pose estimates generated by the 3D pose recovery framework described in section 3.4.5.1. In addition, we also provide results of the real 3D pose estimates produced by our algorithm as a reference (see section 3.4.5.1).

Finally, we demonstrate practicality the generalisation potential of TLE in a challenging computer vision applications, i.e. pose recovery from multiple cameras (4.5.6) and view dependent action recognition from monocular videos (section 4.5.7).

4.5.3. Qualitative Evaluation on Artificial Dataset

Figure 4.10 shows the 3D spaces of the “two moons” dataset produced by ST-Isomap, LE and TLE. Unlike standard LE, the other two methods successfully represent the activity as two periodic motions connected by a transition motion since both aims at preserving spatio-temporal properties of the data. Comparison of computation times (Figure 4.11) illustrates the superiority of TLE, which is almost 9 times faster than ST-Isomap when the whole dataset is used.

This toy problem was introduced to evaluate the global spatio-temporal properties of dimensionality reduction methods [Zhou et al., 2003, Jenkins and Mataric, 2004]. Since neither BC-GPLVM nor GPDM can model any global spatio-temporal relationships between time series, they are not taken into account in this experiment.

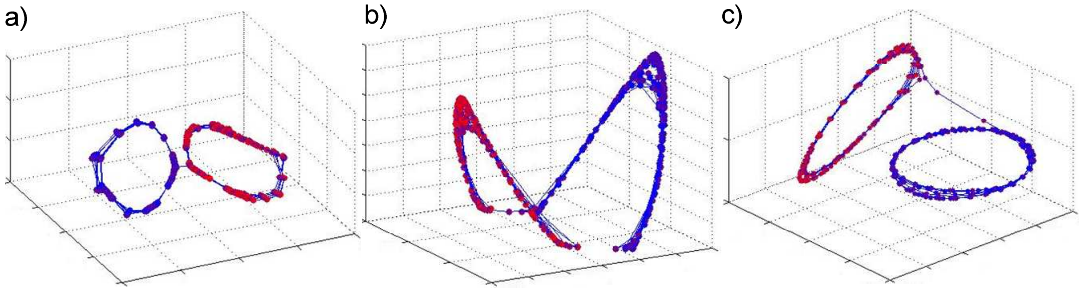


Figure 4.10. The intrinsic representations of the "two moons" dataset which were discovered by: a) ST-Isomap, b) LE and c) TLE.

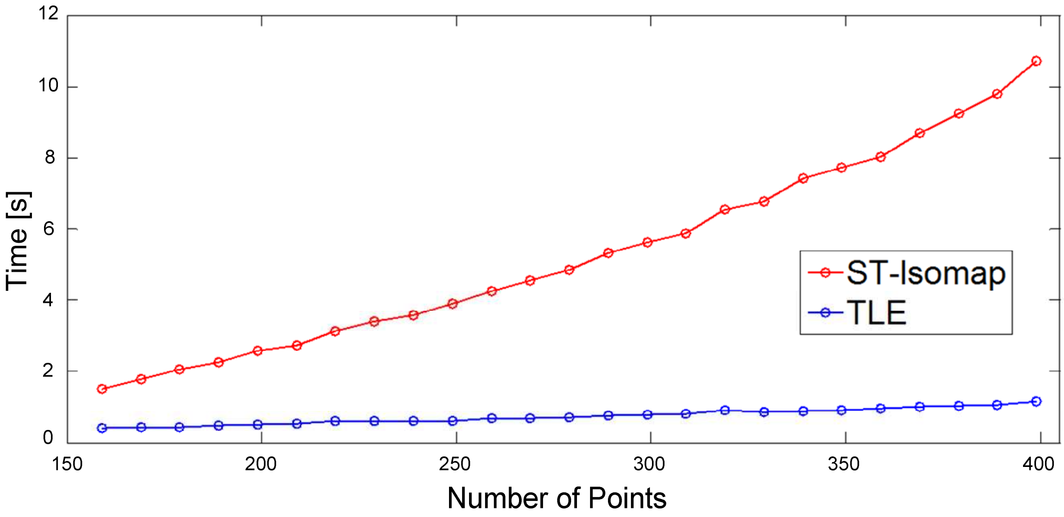


Figure 4.11. Computation time comparison between ST-Isomap and TLE.

4.5.4. Qualitative Evaluation on Image Dataset

In the first step, each image from COIL dataset is represented by a 16384-dimensional vector in the grey level scale. Any series of observations ordered along a single dimension, such as time, may be thought of as a time series. In this experiment, the multidimensional series is defined as a sequential change of camera perspective. Such type of series is refered to as a multidimensianal view series (MVS).

The current temporal methods, such as ST-Isomap, BC-GPLVM and GPDM, consider such MVS as ordinary temporally correlated sequence of points. In contrast, the proposed TLE is more flexible and can take advantage of the sequential view change information to produce better low dimensional model by pre-processing the input data to facilitate the repetition neighbourhood selection procedure.

Although the appearance of objects usually differs significantly, the global shape of many objects is similar and change smoothly along the view circle (Figure 4.12a). As a consequence, MVS can be seen as the alteration of the global object geometry across different views. The global geometry of an object in an image can be represented as a contour of the object shape. Therefore, initially, the shape is extracted by thresholding pixel values of the grey level image to obtain a binary shape of the object (Figure 4.12b). The quality of shapes is improved by applying a combination of morphological operations, i.e. *close*, *shrink*, *thicken* and *majority*. The final contour of the object is generated by removing interior pixels using the morphological *remove* operation and tracing the obtained boundary (Figure 4.12c). As a result, each image is represented as a sequence of internal contour coordinates. Such representation of image changes smoothly accross different views (Figure 4.12c) and, therefore, helps to select accurate and representative repetition neighbours for TLE. In order to deal with different proportions of shapes, a rigid point registration procedure [Myronenko et al., 2007] is employed, whereas a final comparison between pairs of shapes in the DTW alignment is performed using the median Hausdorff distance (see appendix A.3). An example of the neighbourhood similarity matrix constructed during dimensionality reduction using TLE is depicted in Figure 4.13.

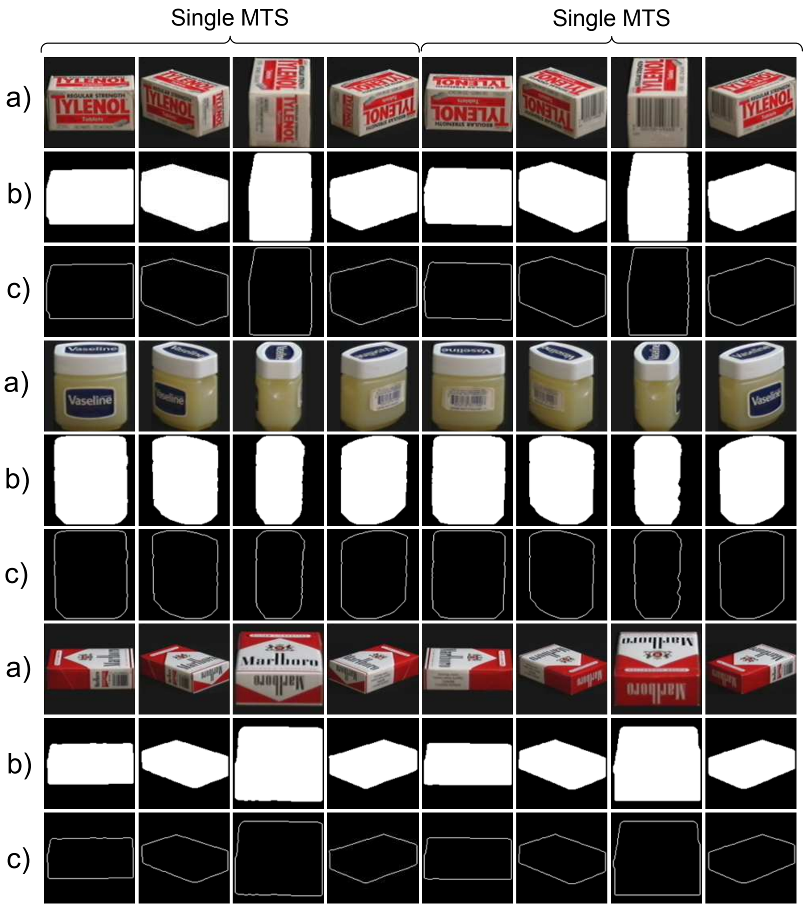


Figure 4.12. Extraction of object contours in different views (every 45 degrees): a) original image; b) foreground mask and c) contour representation. The full cycle of view consists of two time series repetitions.

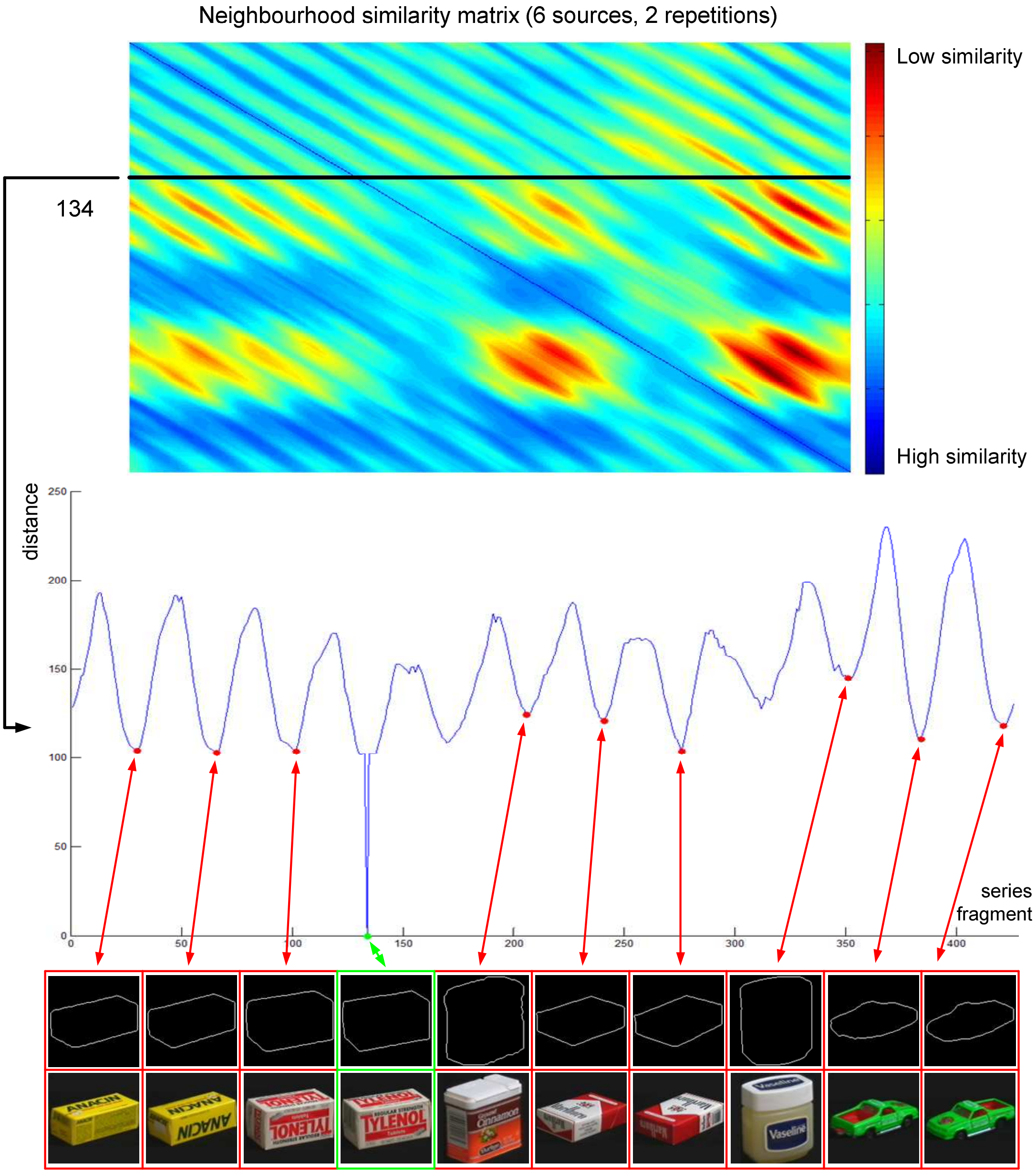


Figure 4.13. Example of neighbourhood similarity matrix created by the TLE for view series using the 6 objects with two repetitions each. Each local minima corresponds to the most similar repetition neighbour in relation to the reference object (green) extracted from different repetitions of the time series.

Figure 4.14 presents a group of 1-dimensional joint view manifolds of 6 original image objects embedded in the 2-dimensional spaces discovered by: LE, BC-GPLVM, ST-Isomap and TLE. Embedded spaces which are produced by Isomap and GPDM are similar to those obtained with LE and BC-GPLVM respectively.

Figure 4.15, Figure 4.16, Figure 4.17 and Figure 4.18 provide the detailed visualisation of the view manifold structures for the related spaces illustrated in Figure 4.14. The geometrically motivated LE as well as the locality preserving BC-GPLVM fail to discover the structure of the view series (Figure 4.14a,b). Both embeddings are dominated by the inter-data variations of series issued from single sources. In order to also model intra-data variations of series between different sources, the spatio-temporal constraints are essential as seen in embedded spaces generated by ST-Isomap and TLE (Figure 4.14c,d). However, although parameters of ST-Isomap are set to optimal values using prior knowledge about the available series, the obtained low dimensional representation is still highly distorted because of object appearance variations. As a consequence, it is difficult to identify any global pattern in that space (Figure 4.17). In contrast, TLE produces a compact and consistent ellipse-like representation which meets our expectations (see section 4.5.1). Figure 4.18 clearly shows that all the objects are arranged according to the view point in this representation, which is invariant to object appearance.

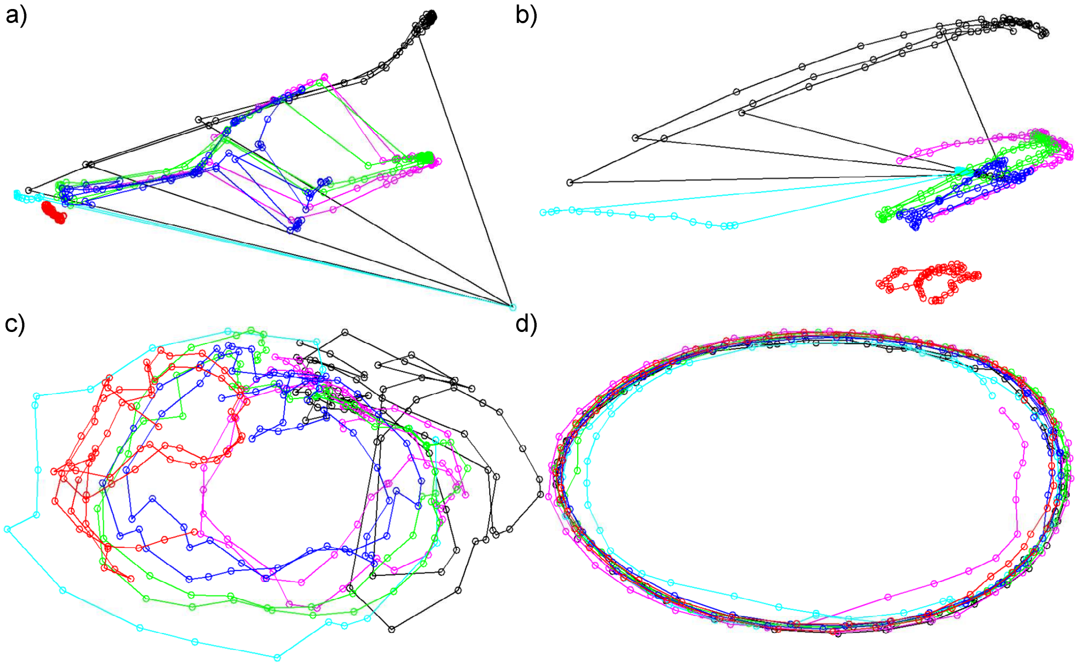


Figure 4.14. The 1-dimensional joint view manifold of 6 image objects from the COIL dataset embedded in the 2-dimensional space discovered by: a) LE; b) BC-GPLVM; c) ST-Isomap and d) TLE. Different colours correspond to series associated with different objects.

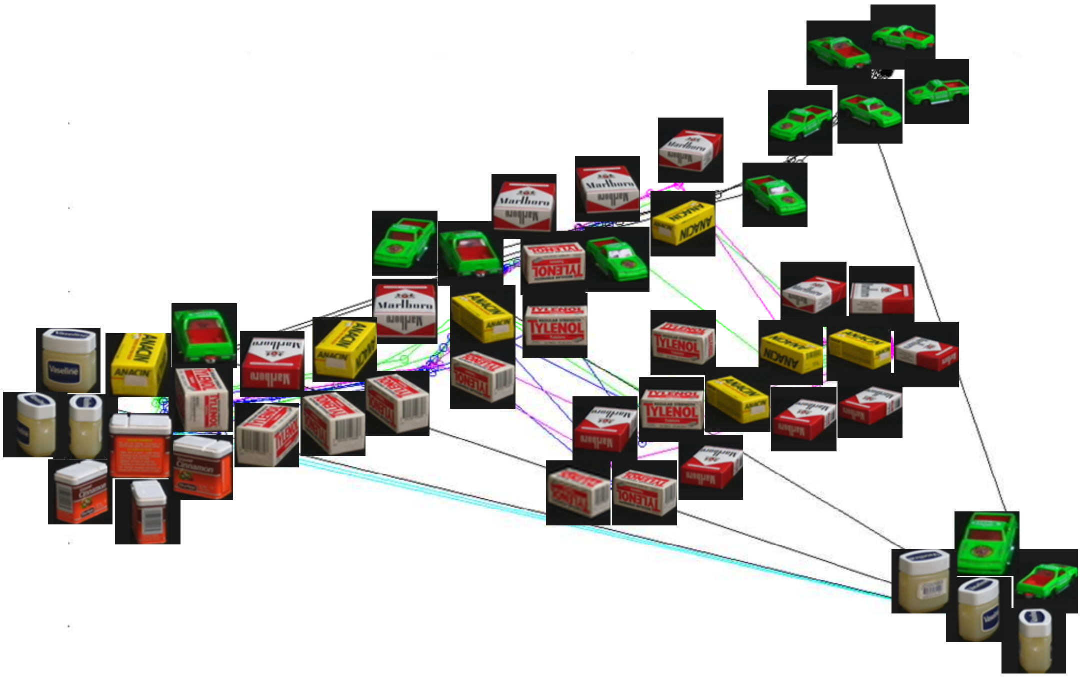


Figure 4.15. The 1-dimensional joint view manifold embedded in the 2-dimensional space discovered by LE with visualisation of corresponding objects.

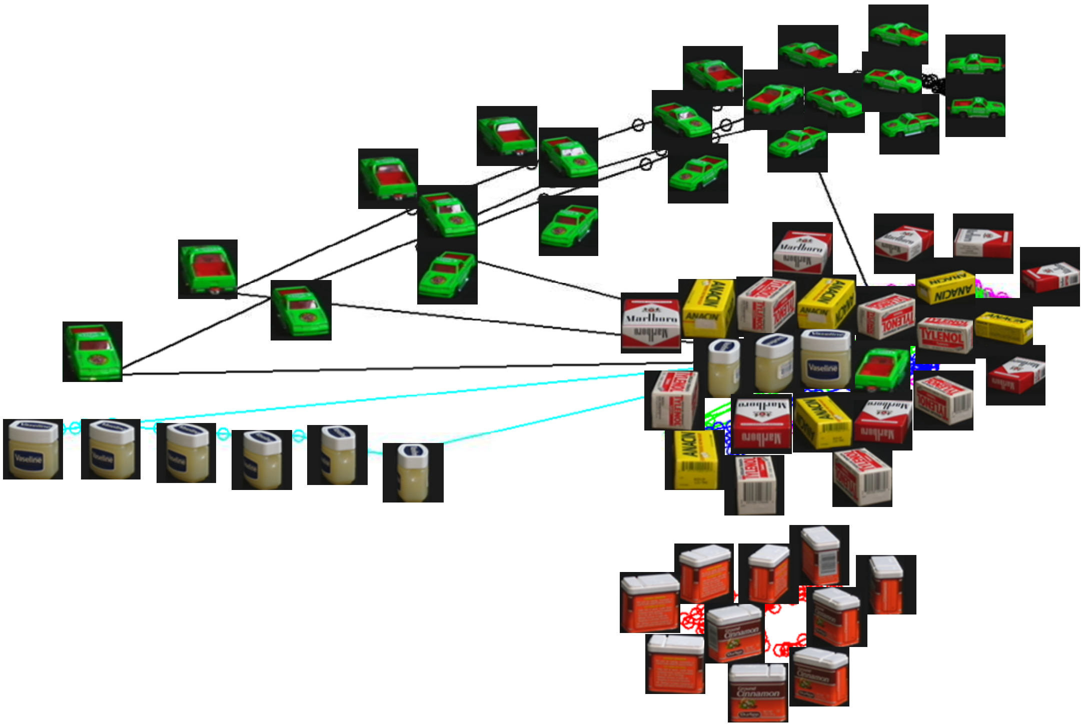


Figure 4.16. The 1-dimensional joint view manifold embedded in the 2-dimensional space discovered by BC-GPLVM with visualisation of corresponding objects.

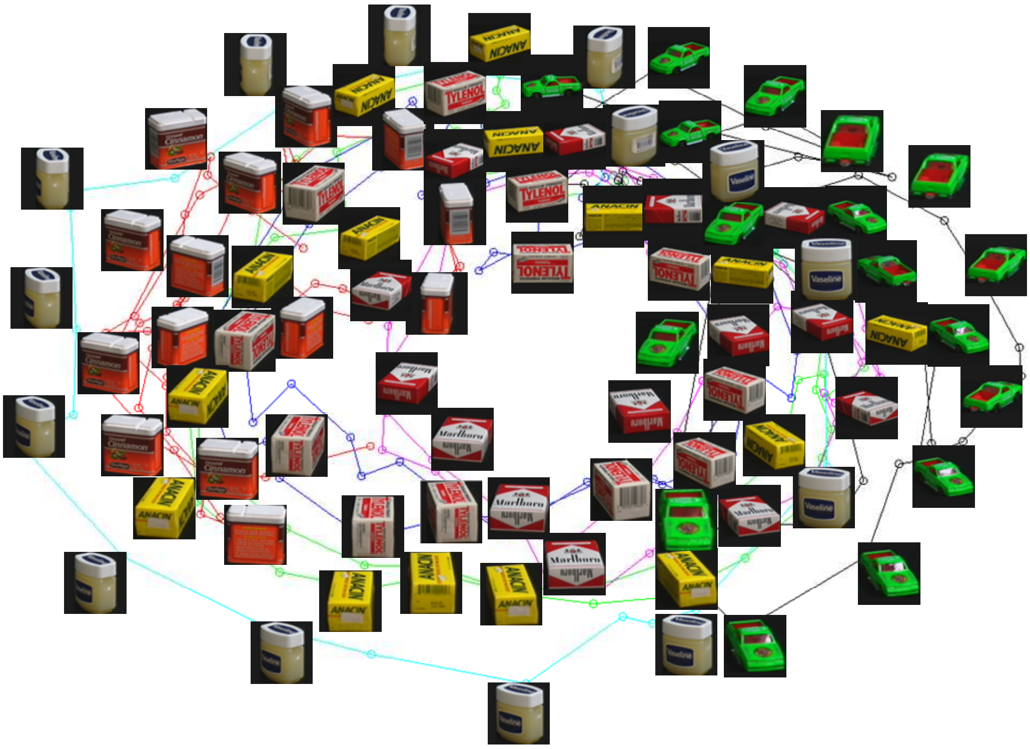


Figure 4.17. The 1-dimensional joint view manifold embedded in the 2-dimensional space discovered by ST-Isomap with visualisation of corresponding objects.

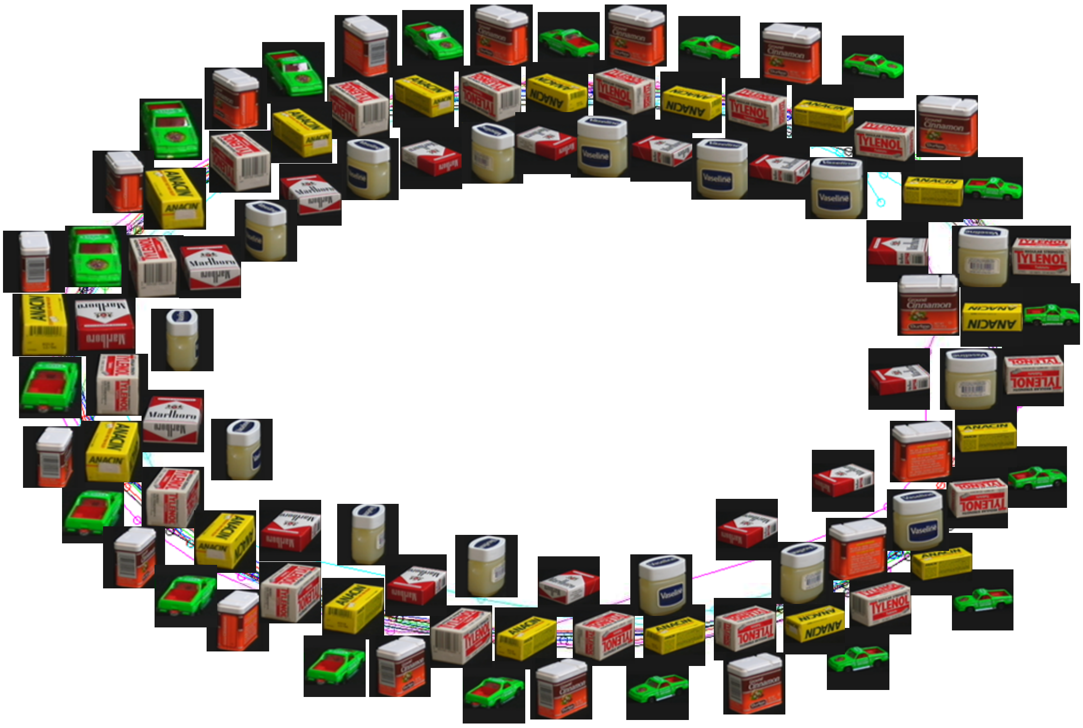


Figure 4.18. The 1-dimensional joint view manifold embedded in the 2-dimensional space discovered by TLE with visualisation of corresponding objects.

4.5.5. Quantitative Evaluation using 3D Pose Refinement Framework

The quantitative comparison between the proposed TLE and other state of the art methods is performed using the 3D pose refinement framework, which has been introduced in section 3.4.5.2. Here, human motion is represented by a sequence of 52-dimensional feature vectors extracted from motion capture data as described in section 2.3.2.3. To measure performances, experiments are conducted using cross-validation taking either one or two subjects for training leaving respectively two or one subjects for testing. Initial test pose estimates are simulated by introducing a Gaussian noise to ground truth poses (see section 3.4.1) thus, final quantitative results are calculated by averaging over 5 test sequences. In addition, the quantitative results are supported with the visual evaluation of the generated low dimensional spaces, since the ideal visual representation is known (see section 4.5.1).

In order to provide a fair comparison, first, the full offline learning pipeline presented in section 3.4.5.2 is used for calculating the average errors for LE and Isomap according to the MI metric. Subsequently, the framework is simplified by removing the quantitative measure block from the offline processing (see Figure 3.9). Then, such simplified offline learning procedure is employed for the exhaustive search of parameter K in LE and Isomap to identify the optimal solution which is used as a reference. The evaluation of TLE, ST-Isomap, BC-GPLVM and GPDM is performed on the simplified framework as well, since there is no need of estimating the K-nearest neighbour parameter. An example of neighbourhood similarity matrix generated during dimensionality reduction using TLE is depicted in Figure 4.6. The online refinement pipeline does not change (Figure 3.9), thus, 3D pose estimates are projected to the embedded space and the nearest neighbour is projected back to the posture space.

Figure 4.19a,b,c and Figure 4.20a,b,c show that Isomap and LE are unable to recover the expected unified ellipse (see section 4.5.1) to represent the 2-subject walking/jogging cycle in the embedded space. In both cases, the obtained spaces are dominated by intra variations between the subject dependent series in agreement with previous experiment (section 4.5.4). Among temporal methods, BC-GPLVM and GPDM discover the closed 2-dimensional curve representations for each subject separately without space generalisation (Figure 4.19b,e and Figure 4.20b,e). Moreover, the symmetrical feature of the motion is not well preserved between succeeding steps (Figure 4.19b,e and Figure 4.20b,e). This implies that the simple constraint of temporal continuity is insufficient to model intra variations between series of different sources. In contrast, the incorporation of some spatio-temporal constraints using either ST-Isomap or TLE, allow generalising the space of the different MTS (Figure 4.19d,f and Figure 4.20d,f). However, the spaces discovered by ST-Isomap are distorted and not smooth even when the optimal parameters are provided, hence, accuracy results are unsatisfactory (Figure 4.19d and Figure 4.20d). On the other hand, TLE produces the expected unique ellipse representation (see section 4.5.1) by embedding nonlinearly the common intrinsic dimension of motion and discarding style variability between different sources as well as different repetitions of the same source (Figure 4.19f and Figure 4.20f).

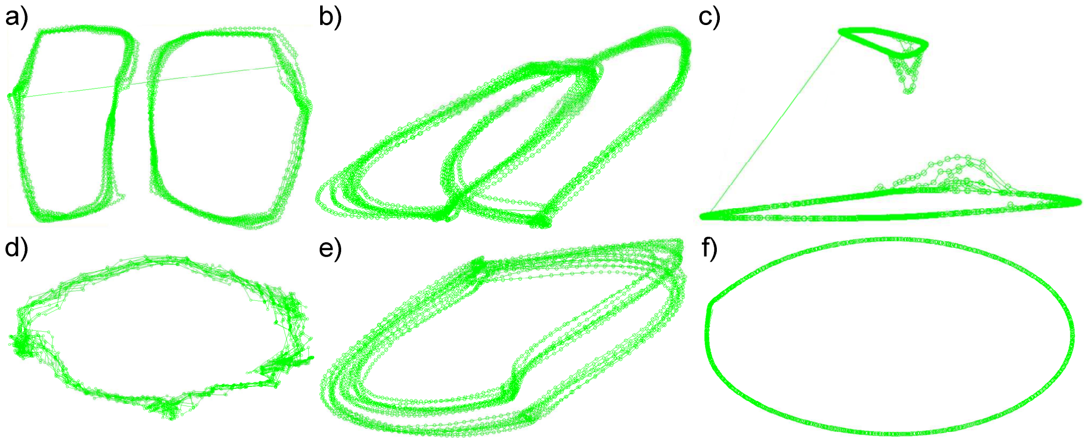


Figure 4.19. Embedded spaces for walking (2 subjects) using a) Isomap, b) BC-GPLVM, c) LE, d) ST-Isomap, e) GPDM and f) TLE.

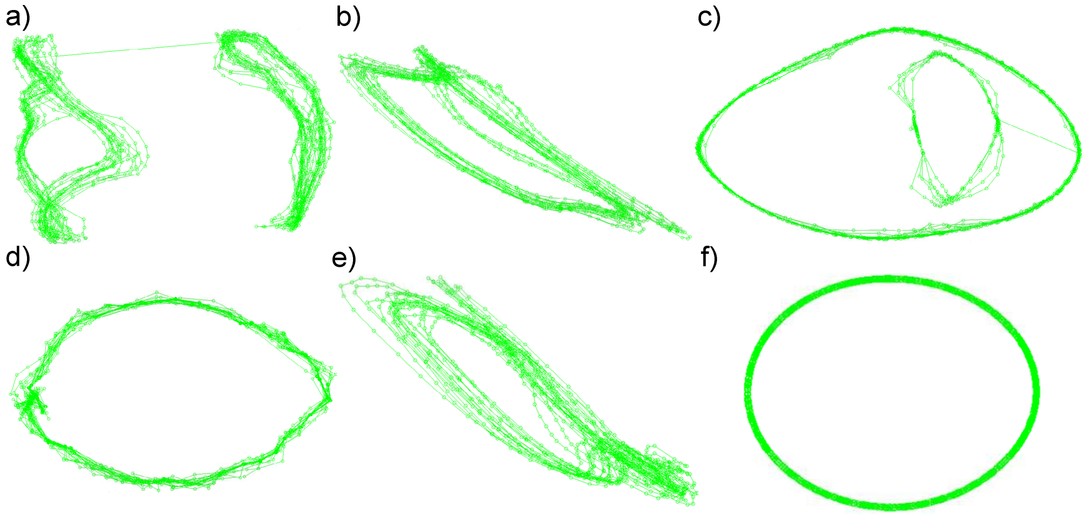


Figure 4.20. Embedded spaces for jogging (2 subjects) using a) Isomap, b) BC-GPLVM, c) LE, d) ST-Isomap, e) GPDM and f) TLE.

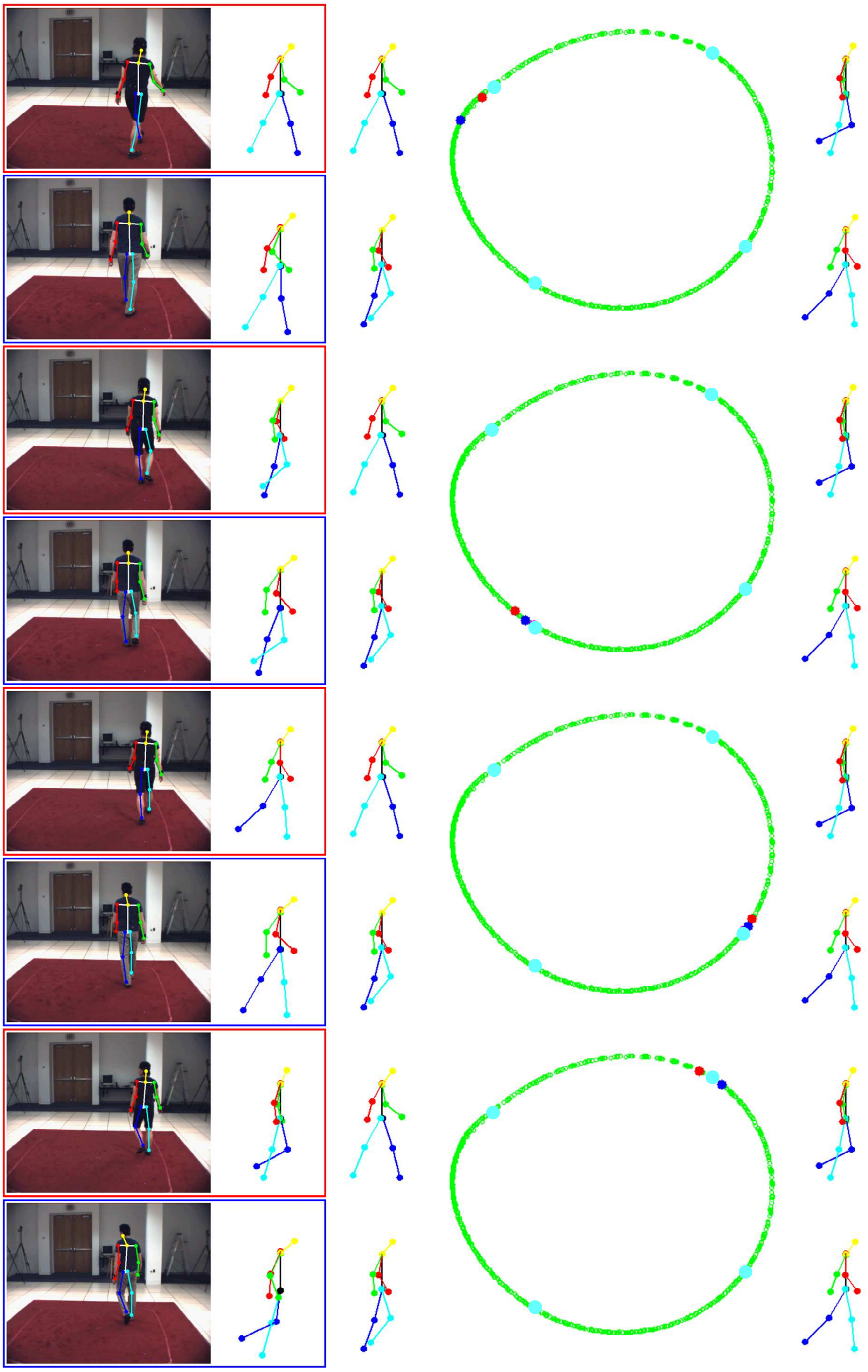


Figure 4.21. Embedded space for walking (2 subjects) using TLE with visualisation of corresponding key poses. The red and blue dots correspond to the poses of 2 subjects depicted on the left, whereas 4 magenta dots represent the reference poses selected from training set for visualisation purpose.

These findings are supported by a quantitative comparison of the obtained accuracy (Figure 4.22). First, performance analysis confirms the generalisation abilities of the methods integrating temporal constraints since data from a second subject improves their accuracy (Figure 4.22). Conversely, performances of Isomap and LE worsen. Among the temporal approaches, BC-GPLVM and TLE benefit the most from additional training samples (accuracy +12%). On the other hand, GPDM's dynamic model seems to be able to optimise most of its parameters from a single subject. Consequently, TLE and BC-GPLVM are the most successful approaches. However, TLE not only displays the best performances and produces better quality embedded spaces (Figure 4.19b,f and Figure 4.20b,f), but it is also significantly faster by an order of magnitude, even when the cost of the proposed automatic parameter estimation procedure is added (Figure 4.23 last column). This is very important because this shows that, unlike BC-GPLVM, TLE has the ability to learn models from much larger training sets which should conduce to even better results.

Note that the results reported in Figure 4.22 for Isomap and LE using a single subject for training are worse than those presented in the last experiment of Table 3.4 (second and third experiments in Table 3.4 are not comparable, because they were performed using perfect motion capture data without any noise, thus the better performance is expected). The reason of worse results in this experiment is that the more challenging and exhaustive evaluation is carried out according to the leave-one-subject-out procedure; for instance training is performed with the male subjects S2 and S3, while testing is done with the female subject S1 (see Figure 3.5), or training with the short subjects S1 and S2 and testing using the tall subject S3 (see Figure 3.5).

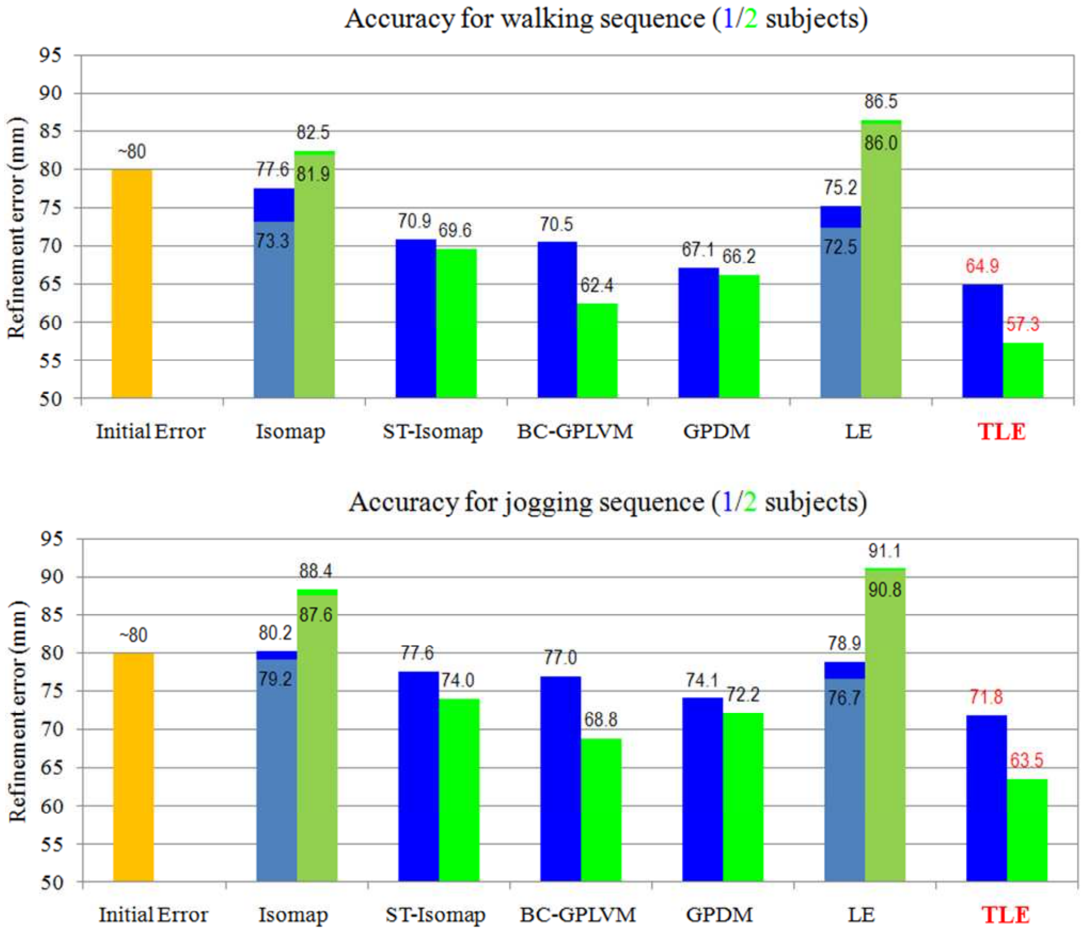


Figure 4.22. Average refinement RMS error of cross validation for walking and jogging sequences using either one (blue) or two (green) subjects for training. Error for the optimal neighbourhood size for Isomap and LE is depicted within corresponding bars.

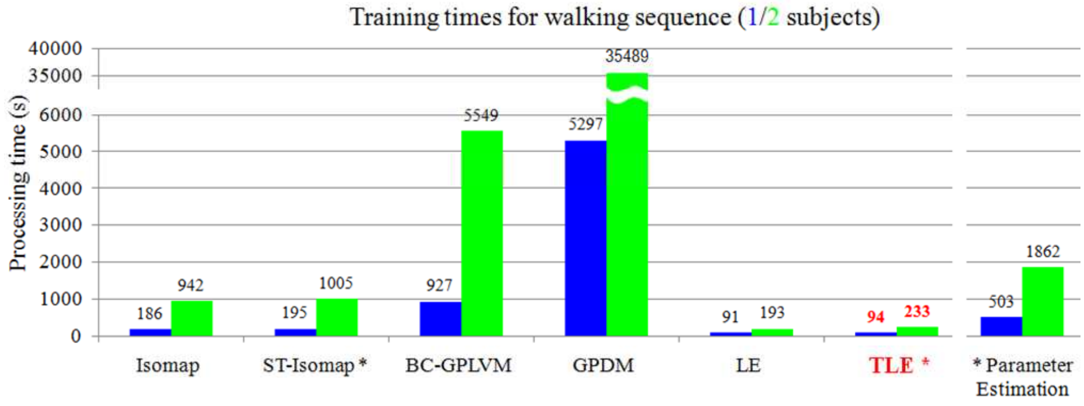


Figure 4.23. Training times based on either 1 (blue) or 2 subject (green) walking sequences (parameter estimation is manual for all embedding-based methods).

Finally, the error of the real 3D pose estimates obtained using our 3D pose recovery framework (section 3.4.5.1), when the action model is trained by TLE using 2 subjects, is equal to $\sim 48mm$ (an improvement of 9% in relation to LE and 4% in comparison to the best Isomap, see Table 3.4). An overall improvement of the refinement framework using the real noise generated by our 3D pose recovery framework is $\sim 40\%$; whereas it is $\sim 28\%$ for artificially simulated noise in this experiment.

The main motivation of this experiment was to provide a comprehensive evaluation platform for the quantitative comparison of the proposed TLE and the state of the art methods. In order to do that, a large testing dataset was simulated. Although we did not intend to design a state of the art pose recovery framework, we believe that our 3D refinement framework has the potential to produce even better pose estimates than those presented in Figure 4.22, when applied on real noise data.

4.5.6. Application to Pose Recovery

In the previous section, the proposed methodology was incorporated into the 3D human pose recovery framework (section 3.4.5.1) as the post processing step to refine pose estimates. Here, we demonstrate the another exemplary application by integrating TLE directly into a 3D pose recovery pipeline from multiple cameras.

4.5.6.1. 3D Pose Recovery Framework

The proposed 3D pose recovery framework aims at estimating a 3D human skeleton from a visual hull using multiple calibrated cameras [Moutzouris et al., 2011]. The entire process is summarised in Figure 4.24.

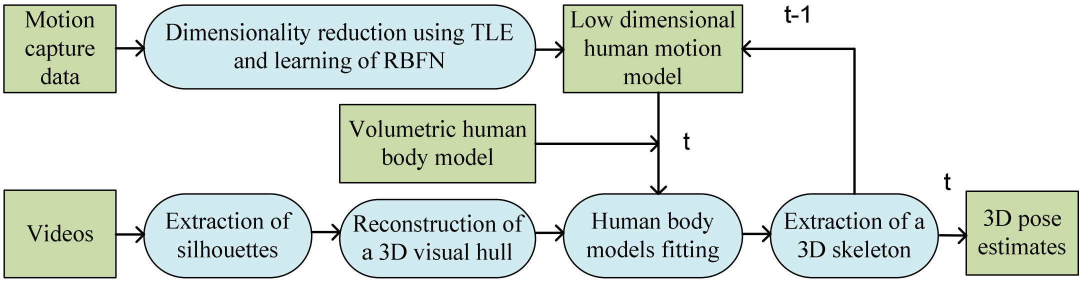

Motion capture data
Dimensionality reduction using TLE and learning of RBFN
Low dimensional human motion model
t-1
Volumetric human body model
t
t
Videos
Extraction of silhouettes
Reconstruction of a 3D visual hull
Human body models fitting
Extraction of a 3D skeleton
3D pose estimates

Figure 4.24. 3D pose recovery framework with a prior model of human motion, which is learned from MoCap data using Temporal Laplacian Eigenmaps.

First, during the learning stage, the space of human motion is reduced by applying TLE as described in previous section 4.5.5. Subjects 1 and 2 are used for the training. Then, the RBFN mapping is learned to provide a bidirectional projection mechanism between spaces as explained in section 4.5.2.1.

Let's assume that human motion is observed by M fixed and calibrated cameras located around a scene of interest. In addition, since we do not deal with the problem of global tracking, the global rotation and translation are assumed to be provided for every frame. The introduced framework exploits two 3D articulated human body models, i.e. skeleton (Figure 2.24) and volumetric (Figure 4.25a) representations. Since relations between body parts are known and both models are expected to satisfy the human body proportions, which were defined by Leonardo da Vinci (Figure 2.24), the transformation between them is straightforward.

The actual process of 3D pose estimation (Figure 4.24) starts with the extraction of silhouettes in each camera using the standard threshold-based background subtraction technique. Then, the 3D visual hull is created from silhouettes shape (Figure 4.25b) according to the bounding edge method [Cheung et al., 2005]. This is achieved by computing the intersection of the M visual cones, which are formed by projecting the contour of image silhouette into 3D space through a pin hole camera centre [Tsai, 1987] (Figure 3.8). For the current frame, the 3D skeleton is estimated by maximising the overlap between the current visual hull representation and predicted volumetric human body models. These predictions are estimated by, first, projecting the 3D skeleton from the previous frame on the low dimensional space using trained RBFN. Then, this projection is associated to its closest low dimensional neighbour in the manifold. Finally, for the obtained low dimensional point, a set of neighbouring samples is selected based on K-nearest neighbours procedure ($K \in \{2, 30\}$) and projected back to the human motion space as the predicted pose candidates for the current frame.

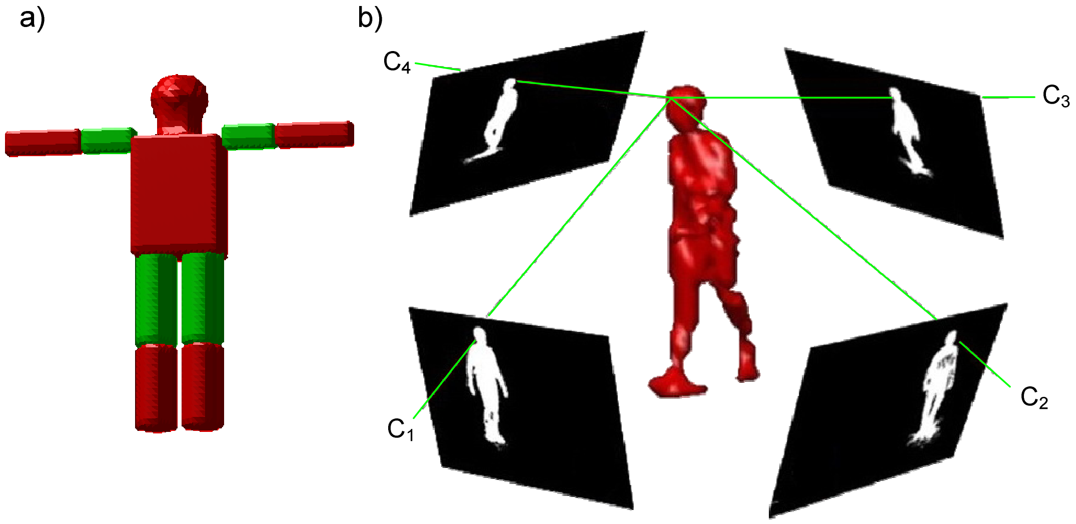


Figure 4.25. a) A volumetric human body model; b) a visual hull, which is extracted from silhouettes using the centre C_M of the pin hole camera model.

4.5.6.2. Results

For testing, the first 100 frames of walking action are used from the Image & MOCAP Synchronized Dataset [HumanEvaI, 2010]. Note that the testing subject is completely different than those used for training. Figure 4.26 presents the mean average error between estimated positions of body joints and the ground truth for each frame. The error is reported using either two (green) or thirty predicted candidates (blue). We also provide the estimation error (red) without using low dimensional human motion model in order to demonstrate the practical advantage of applying TLE. Here, since pose candidates are not available, the human body volumetric model is fitted into the visual hull using an exhaustive and expensive optimisation process, which aims at maximisation of the limbs overlap according to the hierarchical model structure.

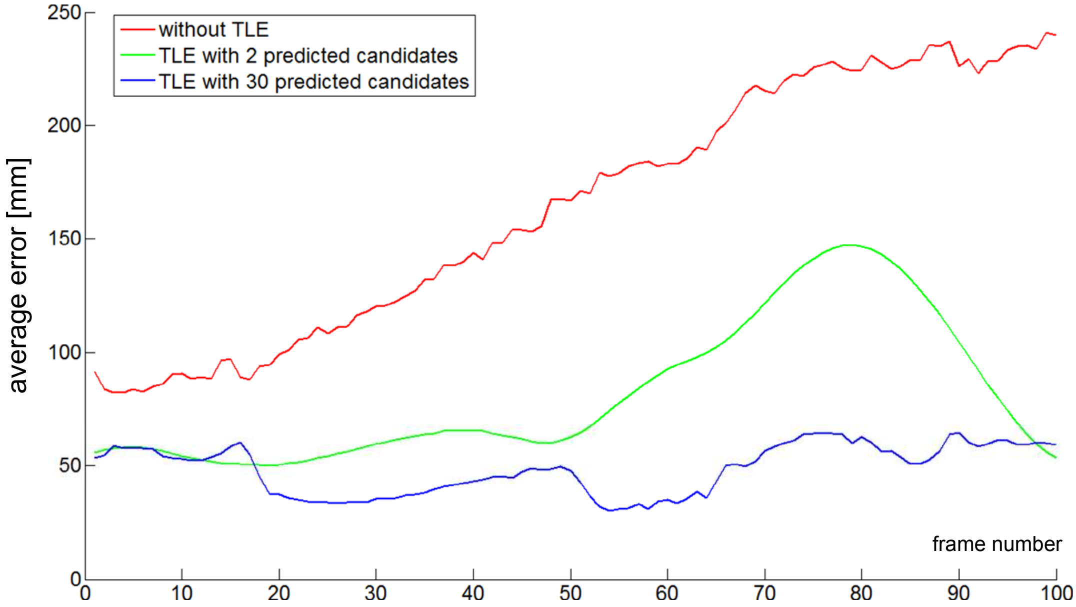


Figure 4.26. The obtained average error of 3D pose recovery for each frame: the framework without TLE (red – 164mm); with TLE using 2 pose candidates (green – 81mm) and using TLE and 30 pose candidates (blue – 48mm).

Figure 4.26 shows clearly that when the process of pose estimation is not constrained, the average error drastically diverges over time. The incorporation of TLE into the pipeline prevents the accumulation of error and allows significantly improving the performance from initial 164*mm* to 48*mm* in the case of thirty predicted pose candidates (accuracy +71%).

4.5.7. Application to Action Recognition

In the previous sections, we have demonstrated the superior performance of TLE. Here, we integrate our technique within a standard human action recognition framework [Blackburn and Ribeiro, 2007] to perform video annotation and demonstrate its generalisation potential in a challenging computer vision task, i.e. view dependent action recognition. Our action recognition framework consists of two processes: offline generation of action descriptors (Figure 4.27) and online classification of new instances of actions (Figure 4.28).

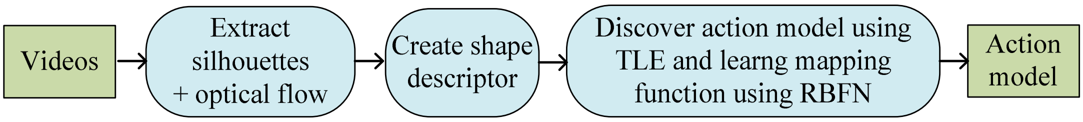

Videos
Extract silhouettes + optical flow
Create shape descriptor
Discover action model using TLE and learng mapping function using RBFN
Action model

Figure 4.27. Process of learning action descriptor.

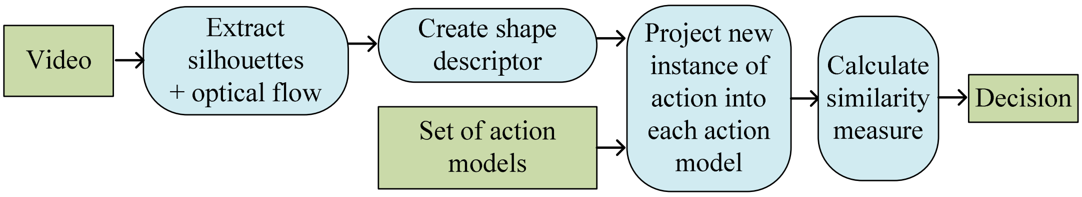

Video
Extract silhouettes + optical flow
Create shape descriptor
Set of action models
Project new instance of action into each action model
Calculate similarity measure
Decision

Figure 4.28. Classification process of a new video.

Let Y denotes the set of $\mathbf{N}$ videos defining an action primitive performed by different people. More formally, Y is defined as $Y = \{Y^s \mid s = 1..\mathbf{N_s}\}$, where s denotes the style index. Each frame y of the video is represented by $\mathbf{D}$ pixels of interest region (see section 2.3.3.1.2): $Y^s = \{y_i^s \mid y_i^s \in \mathbb{R}^{\mathbf{D}}, i = 1..\mathbf{T^s}\}$, where $\mathbf{T^s}$ is the number of frames in the sequence. A unified and compact action model, X , of dimension $\mathbf{d} \ll \mathbf{D}$, is defined by $X = \{X^s \mid s = 1..\mathbf{N_s}\}$, where $X^s = \{x_i^s \mid x_i^s \in \mathbb{R}^{\mathbf{d}}, i = 1..\mathbf{T^s}\}$.

4.5.7.1. Pre-processing and Shape Representation

In the first step, video pre-processing is performed to generate informative and discriminative features of observed human motion. The process starts with isolating foreground pixels in each frame using a simple background subtraction operation (the background of a scene is provided in the dataset). Then, in each frame, the moving foreground object is converted to a binary silhouette, whose quality is improved by applying the morphological *open* operation (i.e. holes are filled) (Figure 4.29b). All silhouettes are normalised to deal with translation and scale variations by using the largest silhouette square bounding box available within the entire action dataset. As a result of this normalisation, any motion becomes relative to the internal deformation of the shape.

Afterwards, silhouettes are converted to a grey level gradient using a signed distance function at each pixel [Elgammal and Lee, 2004a, Blackburn and Ribeiro, 2007]:

$$y(p)=\begin{cases} dist_c(p) & \text{pixel inside contour} \\ 0 & \text{pixel on contour} \\ -dist_c(p) & \text{pixel outside contour} \end{cases} \tag{4.34}$$

where the $dist_c(p)$ is the distance to the closest point on the contour with a positive sign inside the contour and a negative sign outside the contour. Such representation assigns highest values to the silhouette’s most medial axis points. The obtained shape representation is illustrated in Figure 4.29d,f. Grey scale images are used as high dimensional features in our framework (Figure 4.29f), whereas, the colour versions illustrate the effect on the silhouette’s medial axis (Figure 4.29e). The smoothing decreases the variance between subtle differences of similar shapes, such as those caused by clothing and hair variability, by emphasizing medial axis. Once the smoothing is completed, the intensity range in all images is re-scaled to a pre-defined maximum value (e.g., 255).

As a result of the pre-processing stage a static 2916-dimension feature vector is extracted for each frame y. In addition, to increase the discriminative power of each frame, an average optical flow computed by Lucas and Kanade method (see appendix A.2) (Figure 4.29c,e) is included as a dynamic characteristic.

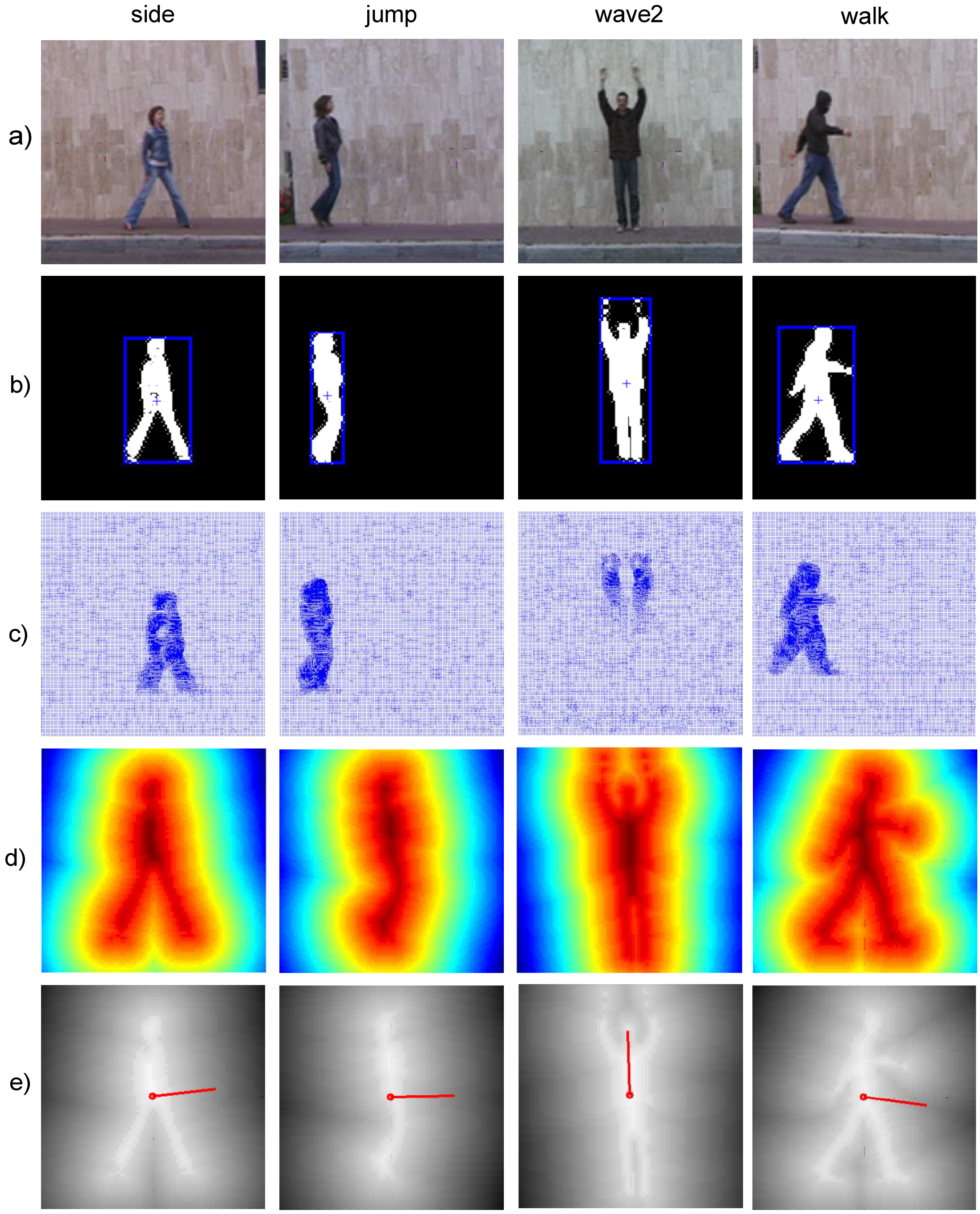


Figure 4.29. Extraction of shape representation: a) original video; b) binary silhouette with the bounding box; c) optical flow; d) implicit distance function representation (colour scale) and e) implicit distance function representation (gray scale) with a dominant motion direction.

4.5.7.2. Learning of Action Descriptors

A unique template model X of observed motions for a specific action is discovered automatically by reducing the dimensionality of static features to 2 dimensions using TLE. A 4-dimensional action descriptor consists of the 2-dimensional template model of the action plus the 2-dimensional dominant direction of the motion obtained using optical flow. Thanks to the generalisation power of TLE, we produce a single unified descriptor per action instead of the action and subject dependent descriptors required by the standard framework [Blackburn and Ribeiro, 2007].

4.5.7.3. Manifold Mapping Function

Our low dimensional action descriptor requires a mapping procedure between the original space Y of motions and low dimensional space X of the action model in order to generalise to unseen examples. Since a unique embedding of action is discovered, the difference of observed shape among different people with the same body configuration is reflected in a nonlinear mapping. This is addressed by learning the advanced RBFN, called generative decomposable model [Elgammal and Lee, 2004b], from the low to high dimensional space for each action model, which is inverted for the projection in the opposite direction as described in section 2.2.2.4.4. This model explicitly decomposes the intrinsic body configuration as a function of time from other conceptually orthogonal aspects which affects observation such as shape and appearance variability. Following the approach of [Elgammal and Lee, 2004b], the generative mapping function is modelled using two factors:

- Content B : a representation of the intrinsic body configuration which characterises motion as a function of time and it is invariant to person shape and appearance.

- Style S : a time-invariant person parameter which describes the person appearance, shape and execution style.

In our framework, content is a continuous domain, while style is represented by the discrete classes present in the training data, thus intermediate styles can be linearly interpolated. As a result, the style continuity is approximated. The procedure of fitting the decomposable generative model to the data consists of two steps. First, a set of style-dependent functions are trained. Then, all functions are combined into a single style-independent projection function.

Since mapping between the embedded action manifold and the observed space is highly nonlinear, generalised RBFN (see section 2.2.2.4.4) is applied to provide the style dependent nonlinear mapping function for each person s in the training data following equation (2.62):

$$Y^s = \psi\left(X^s\right) A^s \tag{4.35}$$

where A^s is a $(\mathbf{Z}+\mathbf{d}+1)\times\mathbf{D}$ matrix of mapping coefficients, which encodes action content and style variability. The interpolation matrix $\psi\left(\bullet\right)$ is defined according to (2.64) by:

$$\psi(X^s) = \{[\varphi(\left\|X^s - c_1\right\|), \varphi(\left\|X^s - c_2\right\|), \ldots, \varphi(\left\|X^s - c_Z\right\|), 1, X^s]\} \tag{4.36}$$

where $C = \{c_j \mid j = 1..\mathbf{Z}\}$ is a set of distinctive representative points along the embedded space and $\varphi(\bullet)$ is a radial basis function (see section 2.2.2.4.4). A^s is calculated by applying the Moore-Penrose pseudo-inverse on matrix $\psi(X^s)$ and solving a linear system of equations: $A^s = \psi(X^s)^+ Y^s$, like in section 2.2.2.4.4. On the contrary to [Elgammal and Lee, 2004b], the manifold representation C is computed directly as a mean style manifold due to the unified representation obtained using TLE. Next, it is transformed by a non-rigid point registration procedure [Myronenko et al., 2007] to better fit the data.

Given the learned nonlinear mapping coefficients A^{s} ($s = 1 \ldots \mathbf{N_s}$) the shape style parameters S are decomposed by fitting an asymmetric bilinear model [Tenenbaum and Freeman, 2000] in the space of nonlinear mapping coefficients [Elgammal and Lee, 2004b]:

$$\mathrm{A} = B \times_3 S \tag{4.37}$$

where all coefficients A^{s} are arranged in an order three coefficient tensor A whose dimensionality is $(\mathbf{Z}+\mathbf{d}+1)\times\mathbf{D}\times\mathbf{N_s}$. S ($\mathbf{N_s}\times\mathbf{N_s}$) denotes the mode-3 basis of A, which represents the orthogonal basis for the style space, whereas B contains the content bases for the mapping coefficient space ($(\mathbf{Z}+\mathbf{d}+1)\times\mathbf{D}\times\mathbf{N_s}$). Mode-i $\times_i$ is a tensor multiplication as defined in [Lathauwer et al., 2000].

This decomposition is performed by representing the tensor A in a matrix form A, where first each coefficient matrix A^{s} is converted to a coefficient vector a^{s} of dimensionality $\mathbf{N_a} = \mathbf{D}*(\mathbf{Z}+\mathbf{d}+1)$ by column wise stacking (columns of the matrix are concatenated to form a vector). Afterwards, all coefficient vectors a^{sv} are arranged in the matrix A of dimensionality $\mathbf{N_a}\times\mathbf{N_s}$. The style orthogonal factors are decomposed from the assembled matrix A using Singular Value Decomposition:

$$\begin{aligned} A &= USV^{T} \\ B &= unstack(US) \\ S &= V^{T} \end{aligned} \tag{4.38}$$

To avoid over-fitting, the dimensionality of style orthogonal space S is reduced to retain a subspace representation by preserving 99% of the original information. The reduced dimensionality for tensors B, S are $(\mathbf{Z}+\mathbf{d}+1)\times D\times\mathbf{n_s}$, $\mathbf{n_s}\times\mathbf{N_s}$ respectively, where n_s denote the number of basis maintained for the style factor.

As a result, the style-independent projection function, which generalise the space of the action descriptor (Figure 6.9d), is expressed by equation:

$$y^{s} = \psi(x) * (B \times_3 s) \tag{4.39}$$

where any image observation y^{s} is synthesized from the body configuration represented by an embedding coordinate x using the estimated style vector s of dimensionality $\mathbf{n}_{\mathrm{s}}$ and the learned content tensor B.

4.5.7.4.Action Classification Process

Action recognition is accomplished by a nearest-neighbour classification scheme. First, a new instance of action is pre-processed and then projected into each action model using the corresponding generative decomposable model presented in the previous section 4.5.7.3. The similarity between the projection and the model is calculated using the sum rule of the following three metrics: the modified Hausdorff distance (see appendix A.3, equation (6.28)), curve dissimilarity function [Frenkel and Basri, 2003] and optical flow variation.

Given a new instance of action $\tilde{Y}$, the corresponding embedded coordinates $\tilde{X}$ on the manifold and the person style parameter $\tilde{s}$ are obtained by minimising the following reconstruction error:

$$\arg\min_{x,s} \left\| \tilde{y} - \psi(\tilde{x}) * (B \times_3 \tilde{s}) \right\|^2 \tag{4.40}$$

If the style vector, $\tilde{s}$ is known we can obtain a closed form solution for $\tilde{x}$ and vice versa. This leads to an iterative procedure for estimating $\tilde{s}$ and $\tilde{x}$ simultaneously until equation (4.40) converges [Elgammal and Lee, 2004b]. First, the style $\tilde{s}$ is initialised using a mean style vector, which is derived from S. Then, the embedded coordinates $\tilde{x}$ are computed by solving a linear system of equations using the Moore-Penrose pseudo-inverse (from equation (2.65)):

$$\psi(\tilde{x}) = \tilde{y}(B \times_3 \tilde{s})^{+} \tag{4.41}$$

Coordinates of $\tilde{x}$ are provided by the last $\mathbf{d}$ rows of the matrix $\psi(\tilde{x})$. The optimal style $\tilde{s}$ is assumed to be approximated as a weighed linear combination of style classes present in the training data:

$$\tilde{s} = \sum_{i=1}^{N_S} \tilde{w}_i s_i,\ \sum_{i=1}^{N_S} \tilde{w}_i = 1,\ \tilde{w}_i > 0 \tag{4.42}$$

In order to solve for the linear regression weights w, let's assume that the observation $\tilde{y}$ is drawn from a Gaussian mixture model centred at $\psi(\tilde{x}) * (B \times_3 s)$ for each style class s. Then, the observation probability given the content and style is expressed by:

$$p(\tilde{y} \mid \tilde{x}, s) \propto \exp(-\left\| \tilde{y} - \psi(\tilde{x}) * (B \times_3 s) \right\|^2 / 2\sigma^2) \tag{4.43}$$

and proportional style conditional class probabilities are defined by:

$$p(s \mid \tilde{x}, \tilde{y}) = \frac{p(\tilde{y} \mid \tilde{x}, s)\, p(s)}{\sum_s p(\tilde{y} \mid \tilde{x}, s)\, p(s)} \tag{4.44}$$

The new style $\tilde{s}$ is estimated using equation (4.42) where $\tilde{w}_s$ is set to $p(s \mid \tilde{x}, \tilde{y})$. Given these two steps, both parameters, i.e. embedding coordinates $\tilde{x}$ and style vector $\tilde{s}$, are optimised iteratively in the Expectation-Maximisation framework [Dempster et al., 1977]. In the E-step, the coordinates are computed given the style parameters, whereas in the M-step new style parameters are re-estimated given the content coordinates. The procedure is repeated until convergence of equation (4.40) [Elgammal and Lee, 2004b].

4.5.7.5. Results

Action recognition results are presented in Table 4.2 according to the leave-one-subject-out cross validation (see section 2.3.3.4.3). Usage of TLE improves accuracy of the standard framework [Blackburn and Ribeiro, 2007] to 100% which has been the state of the art for this dataset since 2007. Other methods which generate low dimensional action representations are [Chin et al., 2007, Wang and Suter, 2007b, Wang and Suter, 2008, Fang et al., 2009]. However, all these methods do not model the temporal structure of actions during dimensionality reduction. Moreover, all methods require the exhaustive search of the optimal number of nearest neighbours in order to obtain satisfactory accuracy. Since TLE's generalisation property handles stylistic variations displayed by different people, this scheme is scalable to a larger subject population. An example of the neighbourhood similarity matrix constructed during dimensionality reduction using TLE is depicted in Figure 4.30, whereas a 3D visualisation of action manifolds is given in Figure 4.31.

Table 4.2. Action recognition results in comparison to previous results on the Weizmann dataset.

Name	Accuracy	Comments
Our TLE + [Blackburn and Ribeiro, 2007]	100.0%	Model per action
Blackburn [Blackburn and Ribeiro, 2007]	95.0%	Model per action per subject
Blank [Gorelick et al., 2007]	100.0%	No action model
Yeffet [Yeffet and Wolf, 2009]	100.0%	Model per all actions
Schindler [Schindler and van Gool, 2008]	100.0%	Model per action
Wang [Wang and Suter, 2008]	100.0%	Model per all actions
Ta [Ta et al., 2010a]	100.0%	Model per action per subject
Weinland [Weinland et al., 2010b]	100.0%	Model per all actions
Jhuang [Jhuang et al., 2007]	98.8%	Model per all actions
Wang [Wang and Suter, 2007b]	97.8%	Model per action
Roth [Roth et al., 2009]	97.0%	Model per all actions
Kellokumpu [Kellokumpu et al., 2008]	95.6%	Model per action
Junejo [Junejo et al., 2008]	95.3%	No action model
Brendel [Brendel and Todorovic, 2010]	95.0%	No action model
Ta [Ta et al., 2010b]	94.5%	Model per all actions
Chin [Chin et al., 2007]	93.0%	Model per action
Liu [Liu et al., 2008]	90.4%	No action model

Fang [Fang et al., 2009]	89.5%	Model per all actions
Zhang [Zhang and Gong, 2010]	89.3%	Model per all actions
Vezzani [Vezzani et al., 2010]	86.7%	Model per action
Klaser [Kläser et al., 2008]	84.3%	Model per all actions
Dollar [Dollar et al., 2005]	80.0%	No action model

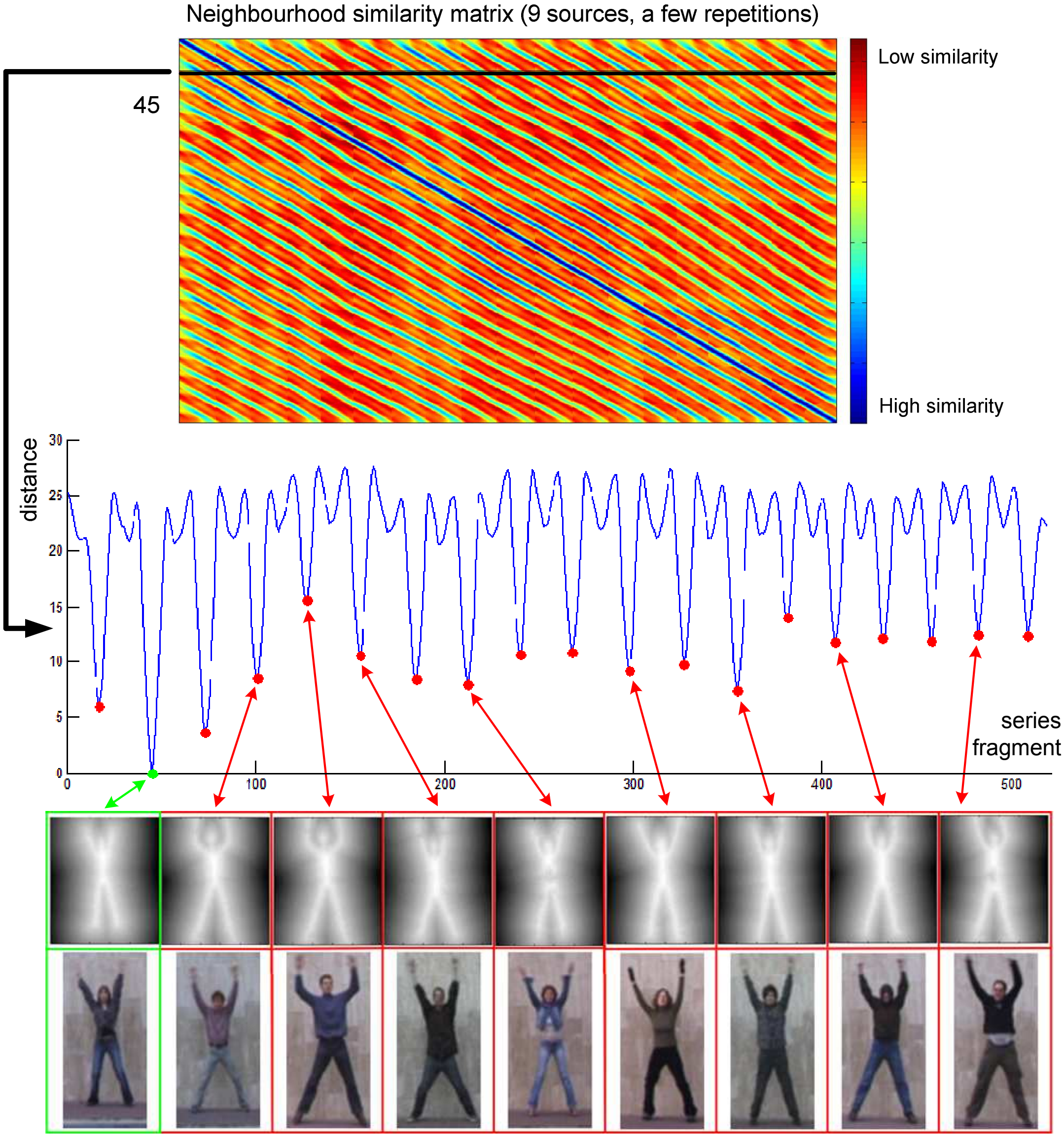


Figure 4.30. Example of neighbourhood similarity matrix created by the TLE using the action 'jack' and 9 subjects with a few repetitions each. Each local minima corresponds to the most similar repetition neighbour in relation to the reference pose (green) extracted from different repetitions of the time series.

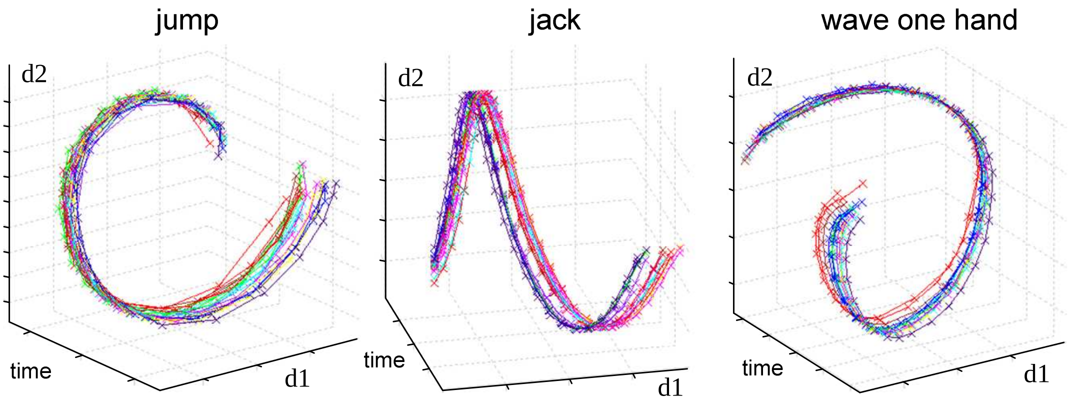


Figure 4.31. Action manifolds generated by TLE with reintroduced time dimension (for visualisation purpose only) to visualise the temporal development of actions. Different colours correspond to different subjects used for learning.

4.5.8. Discussion

In all experiments, the proposed TLE discovers consistently more informative and intuitive low dimensional representations of MTS in comparison to the other state of the art methods. This is achieved by the innovative formulation of temporal and spatio-temporal constraints, which are incorporated into the LE framework.

Analysis of results produced by BC-GPLVM and GPDM (sections 4.5.4 and 4.5.5) confirms that both methods cannot cope with large stylistic variations of data during dimensionality reduction. The same conclusion was drawn by [Urtasun et al., 2008]. Our evaluation gives evidence that the simple temporal correlation between successive points is an insufficient constraint to preserve the global relationships between series during a dimensionality reduction. As a consequence, spatio-temporal constraints are essential to recover the meaningful global pattern of multidimensional series, especially with the increase of input dimensionality and data-inter/intra variations of MTS. For instance, although BC-GPVLM and GPDM produce reasonable low dimensional spaces in section 4.5.4 for motion capture data, they fail completely when applied on a much higher dimensional image dataset (section 4.5.5). This can be explained by the fact that the number of training samples becomes insufficient to overcome the curse of dimensionality (see section 2.2), when the problem complexity increases, due to more sources and larger stylistic variations between them. In contrast, thanks to defined spatio-temporal constraints, ST-Isomap and, in particular, TLE are capable to recover both the local and global pattern of multivariate series in both experiments.

Although ST-Isomap may seem to be an alternative to our methodology, we show that TLE is superior in terms of performance and practicality (sections 4.5.3, 4.5.4, 4.5.5). This implies that our fundamentally different concept of modelling spatio-temporal constraints is more advanced than what was proposed by Jenkins et al. Whereas, they use a naive spatio-temporal approach for neighbourhood selection using distance based correspondence to alter the geometrically motivated cost matrix (according to section 4.3.1), we compose two temporal graphs directly from factual spatio-temporal relationships between neighbours in an automatic manner. Consequently, local temporal neighbours are placed nearby in the embedded space without enforcing any additional artificial constraints. Moreover, assuming that the intrinsic dimensionality is known (the standard assumption for all dimensionality reduction methods), our method is fully automatic and does not require any manual tuning of parameters. On the contrary ST-Isomap is sensitive to a set of parameters which has to be provided in advance: the crucial number of nontrivial neighbours for each point, two similarity factors, the size of temporal window and the size of temporal block for the pre-processing (see section 4.3.1).

Assuming that the optimal parameters are provided, Table 4.3 provides insight into the computational complexities of the most time consuming algorithmic components for all considered temporal techniques. Table 4.3 as well as Figure 4.11

and Figure 4.23 confirm that TLE is much more efficient. Thus, it can be applied to much larger datasets of MTS than the other presented methods.

Table 4.3. Computational complexity of temporal dimensionality reduction methods, where N denotes the number of points in a dataset, I denotes the number of iterations in an optimisation process and p denotes the ratio of nonzero elements in a sparse matrix to the total number of elements N .

Method	Computational complexity
ST-Isomap	$O(N^3)$
BC-GPLVM	$O(I*N^3)$
GPDM	$O(I*(N^3+N^3))$
TLE	$O(p*N^2)$

In the last experiment, TLE was applied successfully to modelling realistic MTS extracted from videos to perform view dependent action recognition (section 4.5.7). The difficulty of this experiment is derived from a high dimensionality of feature vector, a large number of available sources and repetitions, significant stylistic variability between them and finally the considerable size of the whole dataset. Unfortunately, it was impractical to apply ST-Isomap, BC-GPLVM and GPDM in this application, because of their inherent limitations, which has been confirmed by the previous experiments, in particular:

- the prohibitive computational complexity (Figure 4.23, Table 4.3),
- the number of parameters to be set empirically (especially ST-Isomap – section 4.3.1),
- the poor generalisation properties which may suggest a very low recognition rate (for example, see ‘bend’ action in Figure 4.32 discovered by ST-Isomap and BC-GPLVM, or Figure 4.14).

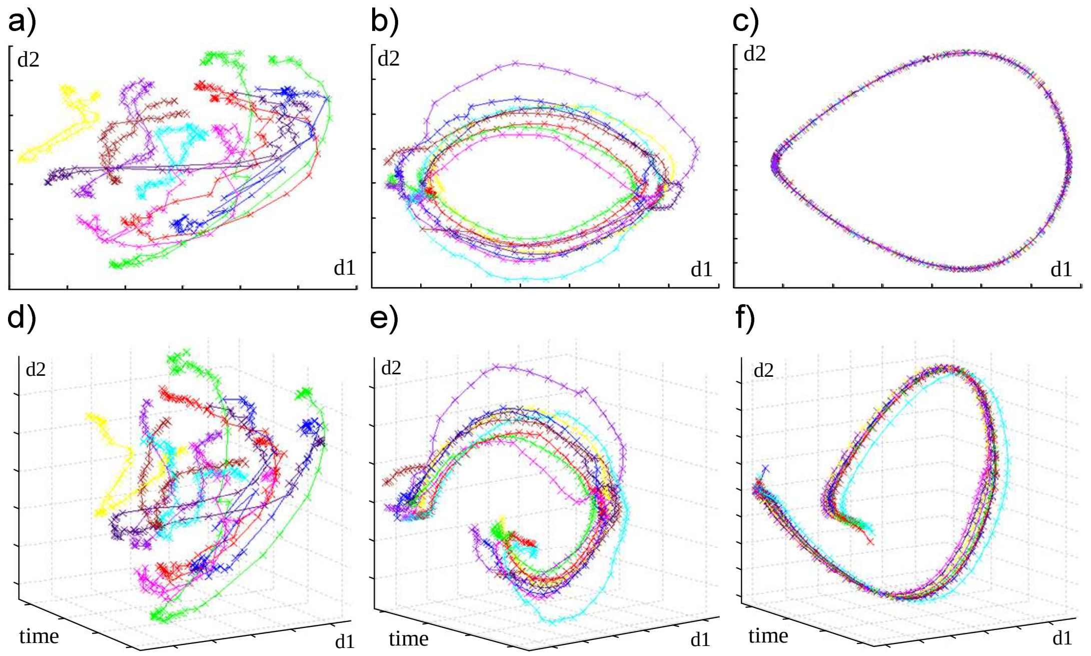


Figure 4.32. Action 'bend' manifold discovered by a,d) BC-GPLVM; b,e) ST-Isomap and c,f) TLE. In second row the time dimension is reintroduced into the space to visualise the temporal development of the action. The poor generalisation properties of a,d,b,e may suggest a very low recognition rate in comparison to c,f.

4.6. Summary

In this chapter, a novel embedded-based dimensionality reduction approach, called Temporal Laplacian Eigenmaps was proposed. It automatically discovers embedded spaces tailored to multidimensional time series, in particular, when data are generated from different sources.

The main motivation of the algorithm is to exploit temporal coherence as a valuable clue in the dimensionality reduction process. This is achieved by inclusion of time series constraints in the form of temporal graphs, in the LE framework without requiring the manual tuning of parameters. Two types of constraints were proposed: temporal within time series and spatio-temporal between different time series. As a result, TLE is able to preserve implicitly the local and global temporal topology of the data instead of the local geometry. This means that TLE maintains the temporal continuity of time series during dimensionality reduction process and suppress stylistic variations displayed by different sources of time series by aligning them in the low dimensional space.

Qualitative and quantitative experiments in different domains proved the high quality of the generated low dimensional spaces. Moreover, the practicality of the algorithm was demonstrated in two important computer vision applications: 3D pose recovery and action recognition. These experiments demonstrated that the method is computationally efficient and has excellent generalisation properties.

5. Spatio-Temporal Gaussian Process Latent Variable Model

5.1. Introduction

Advances in data acquisition and storage capabilities during the past decades have led to more and more high dimensional datasets emerging in most branches of science. However, at the same time, the amount of available data samples is severely insufficient in relation to the sample dimensionality to cover adequately the complexity and richness of measured phenomena. As a consequence, scientists very often face the problem of the generalisation of the known data samples to the entire distribution of possibilities to obtain a reliable model of the observed phenomenon. This issue can be tackled effectively by nonlinear probabilistic dimensionality reduction (section 2.2.2.3). In contrast to deterministic dimensionality reduction methods (section 2.2.2.2), it allows not only eliminating redundancies and irrelevant information present in data while ensuring the maximum possible preservation of information, but it also approximates the underlying distribution of the observed space using only a small number of corresponding hidden variables. As a consequence, a continuous and generative model is created which exhibits excellent generalisation properties to unseen data. In addition, it can be learned successfully without over-fitting using significantly less data samples than space dimensions [Lawrence, 2004, Lawrence, 2005], which is a desired property for many real-life problems. For instance, a probabilistic and generative model of high dimensional data may be used in such applications as tracking, animation, pose recovery, robots controlling and classification.

Obviously, the same generalisation problems arise when dealing with multidimensional time series (MTS), thus, the probabilistic exploration of MTS is an appealing concept with a lot of potential applications including MTS classification. To the best of our knowledge, the probabilistic analysis of time series using dimensionality reduction transformation, which is constrained explicitly by MTS structure, has never been addressed by the research community. In this chapter, we propose a methodology which takes advantage of the MTS temporal structure in order to learn the probabilistic generative model tailored to the MTS space.

On one hand, the previous chapter introduced a novel and powerful method, called Temporal Laplacian Eigenmaps (TLE). It allows the automatic recovery of low dimensional spaces tailored to multivariate time series (MTS), in particular generated from different sources. Although TLE proves its superiority in this challenging task in comparison to the other popular state of the art methods, it is a deterministic framework which does not model uncertainty of series space.

On the other hand, GPLVM is a very attractive probabilistic alternative for nonlinear dimensionality reduction. It emerged in 2004 [Lawrence, 2004] and instantly made a breakthrough in dimensionality reduction research (see section 2.2.2.3.2.2.2). The novelty of this approach is that in addition to the optimisation of low dimensional coordinates during the dimensionality reduction process as other methods do, it marginalises out parameters of a smooth and nonlinear mapping function from low to high dimensional space. As a consequence, GPLVM defines a continuous and generative low dimensional representation of high dimensional data, which is called latent space. Current GPLVM based approaches have proven to be effective in many tracking and animation applications (see section 5.2), when preservation of MTS variability is desired, assuming relatively small stylistic variations among MTS. However, extensive study of the GPLVM framework has revealed some essential limitations of the basic algorithm. As we have seen in the previous chapter 4, the GPLVM family is not suitable for discovering a unified low dimensional representation of MTS in the presence of stylistic variations because of the absence of global spatio-temporal constraints (section 4.5). Another key drawback of GPLVM is its computationally expensive learning process (see sections 2.2.2.3.2.2.2 and 4.5.5) which may converge towards local minima [Urtasun et al., 2008]. Although these methods have been applied successfully in tracking and animation, they are clearly inappropriate in a context of MTS recognition based applications where the discovery of a unique content pattern is more valuable than modelling stylistic variations and usually learning is performed on large datasets.

As we have seen in previous chapter 4, modelling of MTS is not a trivial problem because of the inherent complexity in terms of stylistic variations, redundancies and temporal correlations. In this chapter, we tackle this fundamental problem within a probabilistic framework by introducing a novel concept of spatio-temporal interpretation of GPLVM. The main innovation is a combination of the generalisation potential of TLE with the probabilistic generative model of GPLVM to formulate a probabilistic nonlinear dimensionality reduction algorithm. We call it Spatio-Temporal GPLVM [Lewandowski et al., 2011] (ST-GPLVM). ST-GPLVM is capable of producing an underlying probabilistic model of MTS in the presence of stylistic variations. Our main contribution is an integration of a spatio-temporal 'constraining' prior distribution over a latent space, which is inspired by TLE, within the likelihood optimisation process of GPLVM. As a result, a core pattern of multivariate time series is extracted with associated uncertainties of prediction, whereas style variability is marginalised. Qualitative and quantitative evaluations confirm the superiority of the concept for a classification of different types of MTS using the GPLVM framework.

The remainder of this chapter is organised as follows. The next section 5.2 provides a brief review of the main variations of GPLVM and their applications. Then, the theory behind the concept is introduced in section 5.3. Eventually, evaluation results are presented in section 5.4 followed by the chapter summary in section 5.5.

5.2. Related work

GPLVM is a very flexible approach and it has been successfully applied in a range of application domains including pose recovery [Tian et al., 2005, Ek et al., 2007], human tracking [Urtasun et al., 2005, Urtasun et al., 2006a, Urtasun et al., 2006b, Hou et al., 2007, Jing et al., 2008, Moon and Pavlovic, 2008, Gupta et al., 2008, Zhang et al., 2010], computer animation [Grochow et al., 2004, Urtasun et al., 2008, Deena and Galata, 2009], robotics [Shon et al., 2006, Bitzer and Vijayakumar, 2009], wireless telecommunication [Ferris et al., 2007], data visualisation [Lawrence, 2004], classification [Urtasun and Darrell, 2007] and modelling of deformable surfaces [Salzmann et al., 2008].

The standard formulation of GPLVM has been described in section 2.2.2.3.2.2.2, whereas back-constrained (BC-GPLVM) and dynamic (GPDM) extensions in sections 4.3.2 and 4.3.3 respectively. In this section, we summarise the main limitations of the GPLVM framework (section 5.2.1) and discuss another interesting variant of GPLVM, called Locally Linear GPLVM [Urtasun et al., 2008] (LL-GPLVM), in section 5.2.2.

5.2.1. Limitations of GPLVM methods

The whole family of GPLVM based approaches shares some major limitations. First, they cannot extract the global pattern of MTS during dimensionality reduction especially in the presence of stylistic variations (section 4.5). Moreover, they are computationally expensive [Lawrence, 2004, Lawrence, 2007, Urtasun et al., 2008] (see also section 4.5.5), with the processing time increasing cubically with the number of points in a dataset and linearly with the number of iterations in the optimisation process. Furthermore, the GP-LVM objective function is severely under-constrained in the general case [Ek et al., 2009] and, therefore, it is sensitive to local minima if the initialisation of the model is poor [Urtasun et al., 2008].

Although GPLVM frameworks have been applied successfully in a variety of applications (see section 5.2), the above drawbacks prevent successful utilisation of the GPLVM framework for many MTS classification applications such as speech, gesture and action recognition where latent spaces should be inferred from large amount of time series data generated by different subjects and used to classify data produced by unknown individuals.

5.2.2. Locally Linear GPLVM

Locally Linear GPLVM [Urtasun et al., 2007, Urtasun et al., 2008] (LL-GPLVM) extends the concept of imposing high dimensional constraints over a latent space during optimisation process, which was introduced by BC-GPLVM (see section 4.3.2). The main idea of LL-GPLVM is to exploit prior knowledge about the cyclic nature of human motion to enforce a cylindrical topology. This is achieved by two means. First, advanced similarity measures (i.e. kernels) are carefully designed to reflect prior knowledge in a back-constrained mapping function. In particular, two of the three latent dimensions are constrained by the extracted periodic phase of motion and compared during optimisation using a designed distance function on a unit circle. Similarly, the LLE based objective function (see section 2.2.2.2.2.2.3) is adjusted to consider the cyclic phase of motion and incorporated into the GPLVM framework to preserve a domain specific prior knowledge about observed data.

LL-GPLVM is very effective in a preserving style variability of cyclic motions, e.g. walking and running. However, in addition to the general disadvantages of GPLVM family (see previous section 5.2.1), LL-GPLVM comes with three additional important drawbacks. By design, it requires prior knowledge about the expected topology of an action, which is usually unknown, and the creation of constraints which support this topology. Moreover, the current implementation of constraints can deal with only cyclic types of human body motions. Finally, the LLE objective function is based on the empirical setting of the number of nearest neighbours.

5.3. Proposed Methodology

We propose a novel spatio-temporal formulation of GPLVM to extract the intrinsic structure and associated uncertainty of a MTS space. This is achieved by giving a Gaussian process prior to the generative mapping function from the latent variable space, X , to the observed space, Y , under constraints preserving the spatio-temporal MTS patterns of the underlying manifold.

A brief introduction of the methodology is given in section 5.3.1, whereas details are provided in section 5.3.2. Finally, section 5.3.3 summarises our contribution.

5.3.1. Approach Outline

The proposed methodology is summarised in Figure 5.1. Initially the spatio-temporal constraints L are constructed. These spatio-temporal constraints are founded on adaptation of temporal graphs (section 4.4.2.2), which have proved to be very powerful in modelling complexity and dependencies of MTS (chapter 4). They are exploited twofold. First they are used to better initialise the latent space by discovering a low dimensional embedded space which is close to the expected representation. Secondly, they constrain the GPLVM optimisation process, in the form of spatio-temporal constraining prior distribution over a latent space, so that it converges faster and maintains the spatio-temporal topology of MTS. The learning process is performed using two stage maximum a posteriori (MAP) estimation, which is standard for GPLVM (see section 2.2.2.3.2.2.2). The latent positions X, and the hyperparameters, Φ, are optimised iteratively until the optimal solution is reached under the introduced constraining prior $p(X \mid L)$. The key novelty of the proposed methodology is its style generalisation potential. ST-GPLVM approximates a compact and coherent probabilistic distribution of MTS in the observed space by conserving simultaneously the local temporal correlation within each MTS and global spatio-temporal relationships between different MTS. As a consequence, the method is capable to identify common spatio-temporal patterns of MTS by discarding style variability among all conceptually similar series.

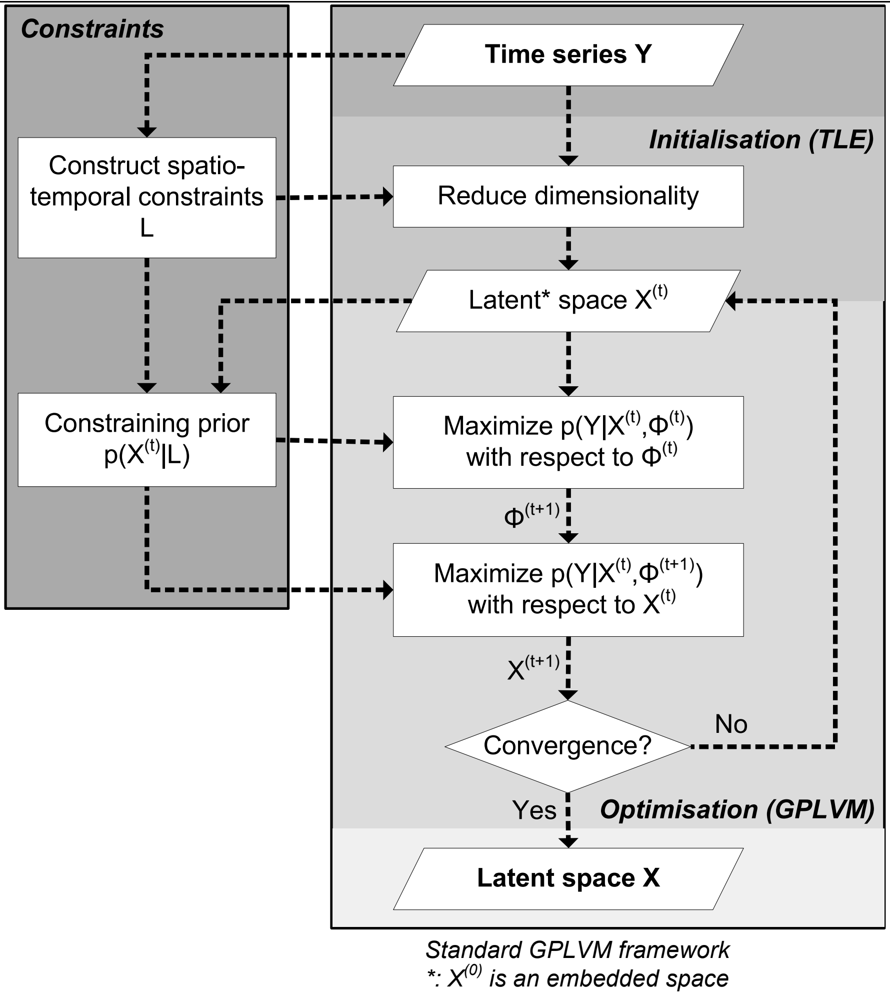

Constraints
Time series Y
Initialisation (TLE)
Construct spatio-temporal constraints L
Reduce dimensionality
Latent* space X^(t)
Constraining prior p(X^(t)|L)
Maximize p(Y|X^(t),Φ^(t)) with respect to Φ^(t)
Φ^(t+1)
Maximize p(Y|X^(t),Φ^(t+1)) with respect to X^(t)
X^(t+1)
Convergence?
No
Yes
Optimisation (GPLVM)
Latent space X
Standard GPLVM framework
*: X^(0) is an embedded space

Figure 5.1. Spatio-Temporal Gaussian Process Latent Variable Model pipeline.

5.3.2. Spatio-Temporal Extension of GPLVM

The proposed ST-GPLVM relies on a spatio-temporal constraining prior which is introduced into the standard GPLVM framework in order to maintain the temporal coherence and suppress the style variability of the MTS space.

First, since neighbourhood graphs have been powerful in designing nonlinear geometrical constraints for dimensionality reduction using spectral based approaches (see section 2.2.2.2.2.2), we use constraints derived from graph theory. Here, in order to model effectively MTS, the temporal graphs, which have been proposed in section 4.4.2.2, are adopted to form automatically a novel conditioned prior $p(X \mid L)$, where L denotes the spatio-temporal constraints. Neighbourhood connections in both graphs represent spatio-temporal dependencies of MTS (see section 4.4.2.1) and implicitly enforce point closeness in the latent space. Consequently, the temporal graph T allows modelling the temporal continuity of MTS, whereas the repetition graph (also referred as spatial graph) S marginalises style variability by aligning MTS in the latent space. The proposed prior probability of the latent variables, which forces each latent point to preserve the spatio-temporal topology of the observed data, is expressed by:

$$p(X \mid L) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{tr(X^T L X)}{2\sigma^2}) \tag{5.1}$$

where $L = L_T + L_S$ combines information from both graphs, and L_G is the Laplacian matrix given by equations (see sections 4.4.2.2 and 4.4.2.3):

$$w_{ij}^G = \begin{cases} \exp(-\left\| y_i - y_j \right\|^2) & \text{if i and j are temporally correlated} \\ 0 & \text{otherwise} \end{cases} \tag{5.2}$$

$$L_G = M_G - W_G \tag{5.3}$$

for each graph individually $G = \{T, S\}$ and $M_G = diag\{m_{11}^G, m_{22}^G, ..., m_{nn}^G\}$, $m_{ii}^G = \sum_{j=1}^{\mathbf{N}} w_G^{ij}$. σ denotes a global scaling of the prior and controls the 'strength' of the constraining prior. Note that, although $p(X \mid L)$ is not a proper prior, conceptually it can be seen as equivalent to a prior for a given set of constant weights L in agreement with the notation of [Urtasun et al., 2008].

The spatio-temporal formulation of GPLVM is introduced by designing an objective function, where the standard uninformative prior $p(x)$ is replaced by the proposed conditioned instructive prior $p(X \mid L)$ to form a new objective function:

$$p(X, \Phi \mid Y, L) \propto p(Y \mid X, \Phi) p(X \mid L) p(\Phi) \tag{5.4}$$

where graph-based spatio-temporal constraints L are imposed on the latent space. Although distance relation between neighbours (especially spatial ones) may be large in L according to equations (4.19) and (4.27), it is infinite between unconnected points. Therefore optimisation of the above objective function enforces implicitly closeness of temporally correlated points in the latent space. Maximisation of the new objective function (5.4) is equivalent to minimising the negative log posterior of the model:

$$
\begin{aligned}
&L(X,\Phi) = -\ln p(X,\Phi \mid Y,L) = \\
&= \frac{1}{2}((\mathbf{DN}+1)\ln 2\pi + \mathbf{D}\ln\left|\Sigma\right| + tr(\Sigma^{-1}YY^{T}) + \sigma^{-2}tr(X^{T}LX)) + \sum_{i}\Phi_{i}
\end{aligned}
\tag{5.5}
$$

Following the standard GPLVM approach, the learning process involves minimising equation (5.5) with respect to X and Φ iteratively using a numerical optimisation method until convergence (see section 2.2.2.3.2.2.2).

ST-GPLVM is initialised using TLE which is able to preserve the constraints L in a produced embedded space (see chapter 4). Consequently, compared to the standard usage of linear PPCA (see section 2.2.2.3.1), initialisation is likely to be closer to the global optimum. In addition, the enhancement of the objective function (2.51) with the prior (5.1) constrains the optimisation process and therefore further mitigates the problem of local minima. The topological structure in terms of spatio-temporal dependencies of MTS is implicitly preserved in the latent space without enforcing any domain specific prior knowledge.

The proposed methodology can be easily applied to other GPLVM based approaches, such as BC-GPLVM (section 4.3.2) and GPDM (section 4.3.3). The extension of BC-GPLVM results in a spatio-temporal model (ST-BC-GPLVM) which provides explicitly bidirectional mapping between latent and high dimensional spaces, where the objective function is designated by substituting (4.3) into (5.4):

$$
p(W,\Phi \mid Y,L) \propto p(Y \mid W,\Phi)\,p(W \mid L)\,p(\Phi) \tag{5.6}
$$

Alternatively, ST-GPDM produces a spatio-temporal model with an associated nonlinear dynamical process in a latent space, where the proposed prior (5.1) is integrated into objective function (4.13) resulting in:

$$p(X,\Phi_X,\Phi_Y,W\,|\,Y,L) \propto p(Y\,|\,X,\Phi_Y)\,p(X\,|\,\Phi_X)\,p(X\,|\,L)\,p(\Phi_Y)\,p(\Phi_X)\,p(W) \quad (5.7)$$

5.3.3. Summary

The proposed ST-GPLVM is a continuous latent variable model, which approximates an underlying probability distribution of MTS in a high dimensional space. This is achieved by learning a probability density function which gives a natural measure of plausibility, assigning higher probabilities to MTS that are similar to those used for training. The learning process is constrained through a novel prior distribution in a latent space which takes into account the local temporal correlation between successive points in MTS and the global spatio-temporal relationships between different MTS. Note that the accurate initialisation of the model using TLE and the incorporated constraints reduce significantly the risk of converging towards local minima. Moreover, since the new objective function is more constrained, the processing time is reduced. Finally, the proposed extension is compatible with a sparse approximation of the full Gaussian process (see section 2.2.2.3.2.2.2) which allows decreasing further processing complexity.

As we will demonstrate in the evaluation section 5.4, the integration of spatio-temporal extension addresses some of the limitations of GPLVM family. In particular, it allows producing generalised latent spaces of MTS in the presence of stylistic variations, which is extremely important for classification of MTS, e.g. in action, gesture and speech recognition applications.

5.4. Evaluation

The proposed approach was validated in terms of performance and robustness in two different domains, i.e. human body motion modelling and sign language recognition.

In this section, first, two datasets, which are used in the evaluation process, are introduced in section 5.4.1. Then, the setup of experiments is explained in section 5.4.2.1 followed by a description of performed experiments in section 5.4.2.2. Subsequently, sections 5.4.3 and 5.4.4 present experimental results. A broad discussion of the obtained results is provided in section 5.4.5. Finally, a practical application of the proposed methodology, i.e. view independent action recognition, is demonstrated in chapter 6.

5.4.1. Datasets

The HumanEva (HE) dataset has been introduced in section 2.3.2.3. In this evaluation, we consider three different subjects performing a “walking in a circle” action in trial 3. Each action comprises the 500 first frames of the longest available continuous sequence of valid MoCap poses, as seen in Table 4.1. We do not use all available frames because of the high computional complexity of the standard GPLVM optimisation proccces. Similarly to previous experiments in section 4.5.1, two successive steps are considered to be a single MTS, which is repeated a number of times in the action. Each subject corresponds to a different source of MTS. Since our goal is to model probabilistic distribution of human poses, as recommended by [Wang et al., 2006, Urtasun et al., 2006a, Wang et al., 2008], we reduce the dimensionality of a walking space to 3 dimensions to facilitate the learning process of the underlying probabilistic model.

Flock Sign Language Dataset [Kadous and Sammut, 2005] consists of 95 signs of the Auslan language which is used by the Australian Deaf and non-vocal communities. Auslan is a dialect of British Sign Language. The dataset was collected by a two-hand system using the two Fifth Dimension Technologies data gloves (left photo in Figure 5.2) and two Ascension Flock-of-Birds magnetic position trackers (right photo in Figure 5.2). Each position tracker provides 6 degrees of freedom (i.e. roll, pitch and yaw as well as x, y and z), whereas the gloves also provide information about all five fingers. As a consequence, each sign is represented by 11 channels of information per hand, which result in a sequence of 22-dimensional feature vectors. All signs were collected from a single native signer in a longitudinal study over a period of nine weeks, thus nine sources of gestures are available. Each sign is considered to be a single MTS and is repeated three times, which results in 27 samples per sign in total. Due to the high flexibility of human hand, gestures exhibit large intra variations between repetitions, as well as various moving speeds. Intuitively, any gesture corresponds to a continuous curve in a hand gesture space, since there is only one degree of freedom, i.e. an innate configuration of hands over time. Gestures are embedded into a 2-dimensional space to model a nonlinearity of hand motion. Similarly to body motion capture data, the third dimension is added for a sign representation in a low dimensional space to facilitate a learning process of underylying probabilistic model.

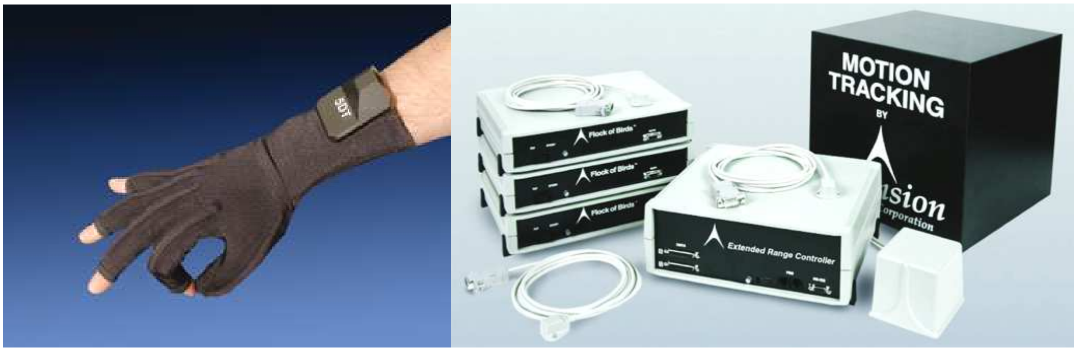


Figure 5.2. The Fifth Dimension Technologies data glove on the left [5DT, 2011] and the Ascension Flock-of-Birds magnetic position tracker on the right [Inition, 2011].

5.4.2. Experimental Framework

The proposed methodology is evaluated through qualitative and quantitative analyses of performance using the original and extended formulation of the three main representatives of the GPLVM family, i.e. GPLVM (section 2.2.2.3.2.2.2), BC-GPLVM (section 4.3.2) and GPDM (section 4.3.3). Note that it does not make sense to incorporate our extension into LL-GPLVM, since the main objective of LL-GPLVM is to model distinctly style variability in a latent space, whereas the goal of our extension is to suppress style variability for the sake of generalisation to extract the intrinsic content pattern.

5.4.2.1. Setup

In all experiments, the computational complexity of the learning process is reduced using the sparse FITC approximation of covariance matrix (see section 2.2.2.3.2.2.2). The back-constrained models use a RBF kernel (section 4.3.2). The global scaling of the constraining prior, σ, and the width of the back constrained kernel were set empirically for each experiment whenever appropriate. Values of all the other parameters of the models are estimated automatically using standard maximum likelihood optimisation during model training.

5.4.2.2. Experiments

First, our new approach is evaluated qualitatively through a comparative analysis of latent spaces discovered by standard non-linear probabilistic latent variable models, i.e. GPLVM, BC-GPLVM and GPDM and their spatio-temporal extensions, i.e., ST-GPLVM, ST-BC-GPLVM and ST-GPDM, where the proposed spatio-temporal constraints have been included (section 5.4.3). The evaluation is conducted using time series of MoCap data, i.e. repeated human motions, which are represented using quaternions as described in section 2.3.2.3.

Then, the superiority of spatio-temporal extension is demonstrated quantitatively in a multivariate stream data classification task in the presence of large stylistic variations. In this experiment, time series are sequences of language signs, i.e. hand configurations over time, which were collected by a set of sensors attached to each hand of native signer. The objective is to perform accurate and automatic sign language recognition [Starner, 1995]. The evaluation is performed using the standard GPLVM, its back-constrained extension BC-GPLVM and finally the proposed spatio-temporal formulation, i.e. ST-GPLVM. Since, ST-BC-GPLVM produces similar spaces similar to ST-GPLVM according to the previous experiment (see for example Figure 5.3b,d and Figure 5.4b,d), it is not considered in this experiment. In addition, it was not possible to use GPDM and its spatio-temporal extension in this experiment because of the prohibitive computational cost of the learning process (see Figure 4.23, Figure 5.3 and Table 4.3). In any case, it was not expected that a dynamical model would perform particularly well in a recognition based application, since it has never been used in this context by the research community.

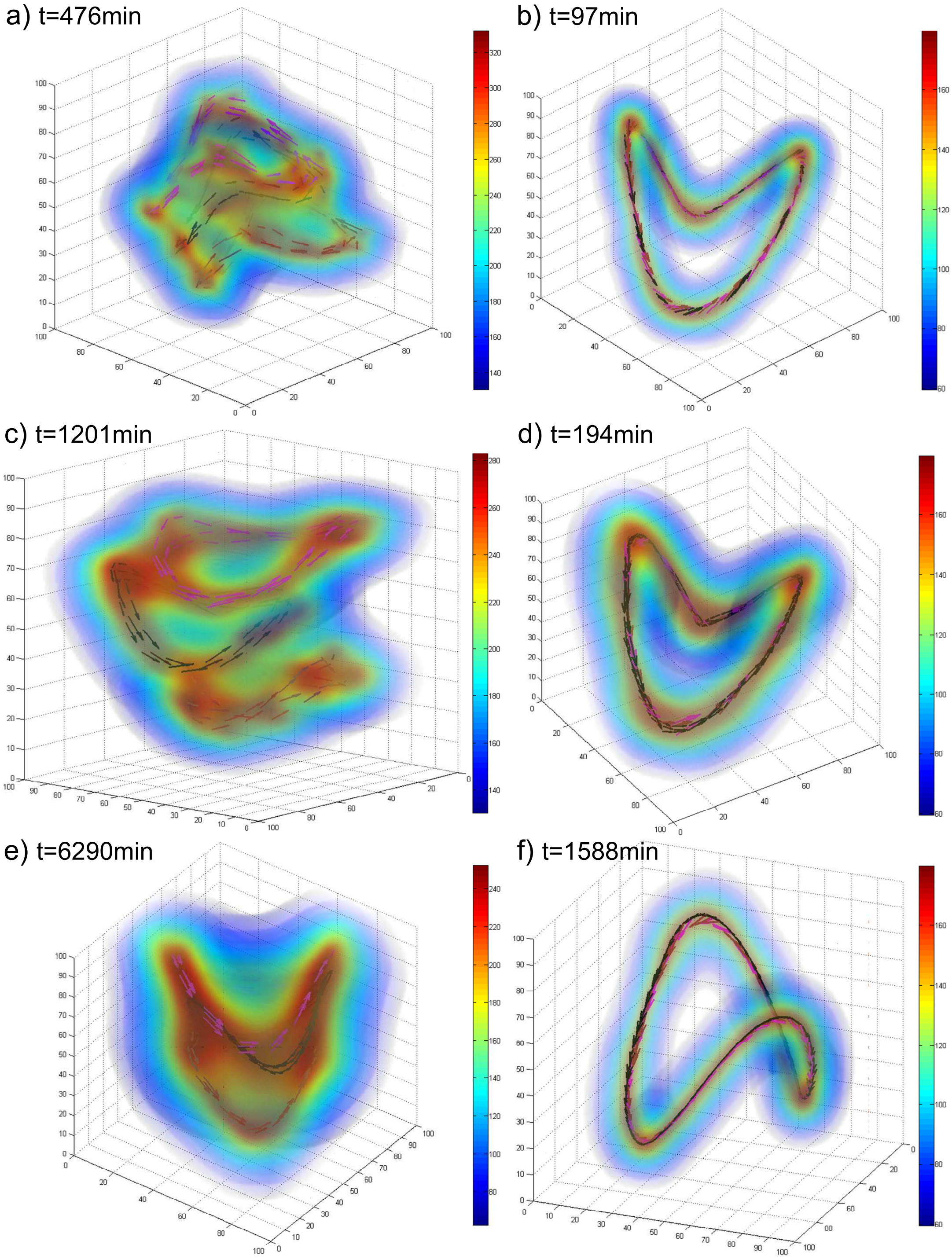
5.4.3. Qualitative Evaluation on Human Motion Dataset

Similarly to other experiments on the human motion dataset (sections 3.4.5 and 4.5.5), a human body movement is represented by a sequence of 52-dimensional feature vectors extracted from motion capture data as described in section 2.3.2.3. In this experiment, the number of inducing variables is set to 10% of the data for the FITC approximation (see section 2.2.2.3.2.2.2), whereas the global scaling of the constraining prior, σ, and the width of the back constrained RBF kernel were set empirically to 10^4 and 10^{-1} respectively.

The learned latent spaces for walking sequences with the corresponding first two dimensions and processing times are presented in Figure 5.3 and Figure

5.4. Qualitative analysis confirms the generalisation potential of the proposed extension. Standard GPLVM based approaches discriminate between subjects in the spatially distinct latent space regions (left column of Figure 5.3 and Figure 5.4). Moreover, action repetitions by a given subject are represented separately. In contrast, the introduction of our spatio-temporal constraint in objective functions allows producing consistent and smooth representation by discarding style variability in all considered models (right column of Figure 5.3 and Figure 5.4). In addition, the extended algorithms converge significantly faster than standard versions. Here, we achieve a speed-up of a factor 4 to 6.

As seen in Figure 5.3 and Figure 5.4, our spatio-temporal extension is adaptable to three established variants of the GPVLM family, i.e. GPLVM, BC-GPLVM and GPDM. There is no clear evidence about which spatio-temporal variant is best, since they are designed to tackle different type of applications. For instance, ST-GPDM may be superior when a dynamical model in a latent space is required, whereas ST-BC-GPLVM may be more valuable when a direct bidirectional mapping function between low and high dimensional spaces is needed. On the other hand, we will demonstrate that ST-GPLVM is a very attractive approach for MTS classification applications, such as hand gesture (section 5.4.4) and human action (chapter 6) recognition.



a) t=476min

b) t=97min

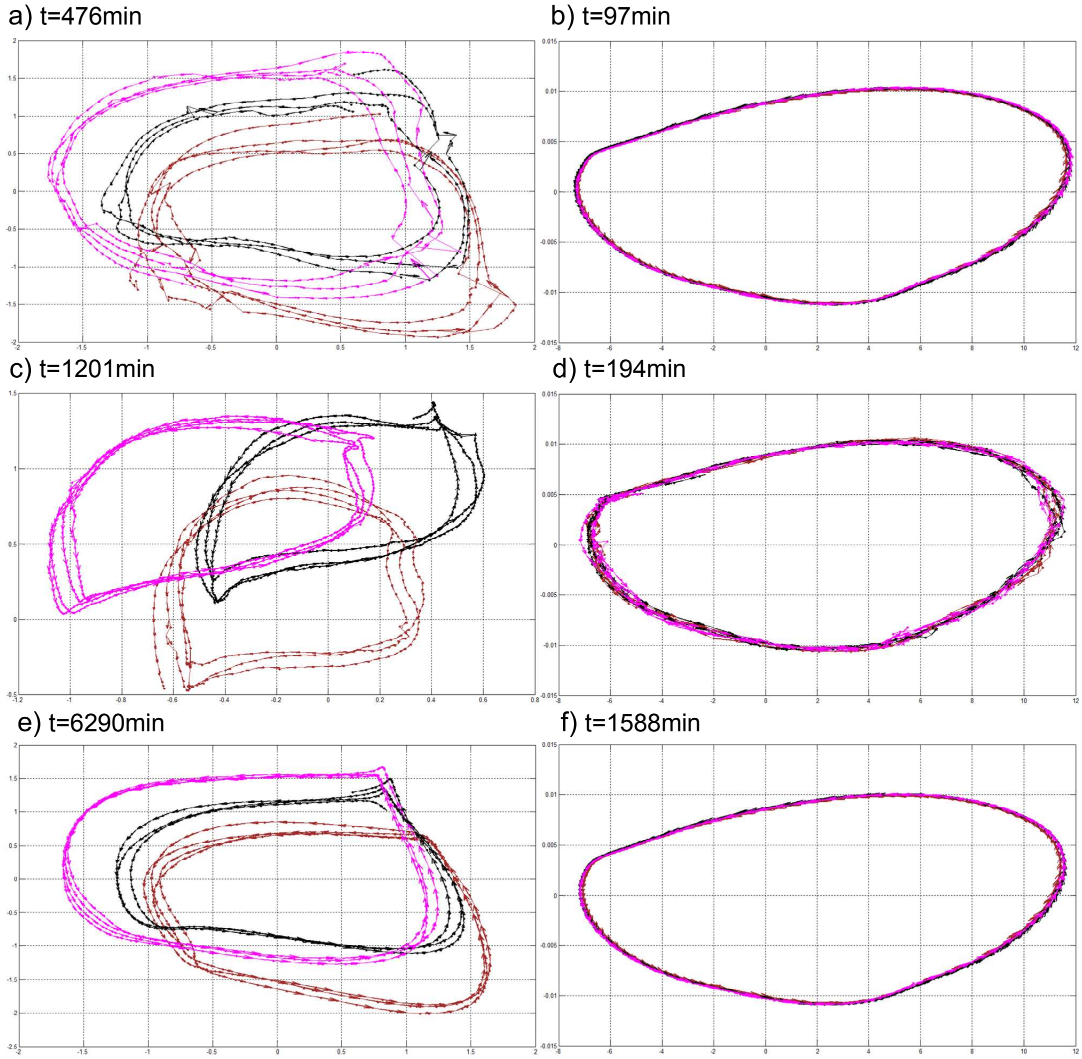
c) t=1201min

d) t=194min

e) t=6290min

f) t=1588min

Figure 5.3. 3D models learned from walking sequences of 3 different subjects with corresponding processing times: a) GPLVM; b)ST-GPLVM; c) BC-GPLVM; d) ST-BC-GPLVM; e) GPDM and f) ST-GPDM. Warm-coloured regions correspond to high reconstruction certainty.



a) t=476min

b) t=97min

c) t=1201min

d) t=194min

e) t=6290min

f) t=1588min

Figure 5.4. Projection of first 2 dimensions from 3D walking models of 3 different subjects in Figure 5.3 with corresponding processing times: a) GPLVM; b) ST-GPLVM; c) BC-GPLVM; d) ST-BC-GPLVM; e) GPDM and f) ST-GPDM.

5.4.4. Quantitative Evaluation on Sign Language Dataset

Here, each sign is represented as a sequence of 22-dimensional feature vectors. Experiments are carried out according to the leave-one-source-out cross validation strategy (see section 2.3.3.4.3), where sources correspond to rounds of perfomed gestures, which were captured in a longitudinal study. A final error is estimated by the average error rate over all experiments. We use one sign repetition of each source for training, whereas testing is performed with all gesture repetitions. The number of inducing variables is set to the length of shortest repetition of the considered sign for the FITC approximation (see section 2.2.2.3.2.2.2), whereas the global scaling of the constraining prior, σ, and the width of the back constrained RBF kernel were set empirically to 10^4 and 10^{-1} respectively.

Given a trained model, a gesture $\tilde{Y}$ of unknown source is recognised by maximising the following introduced estimation likelihood:

$$p(\tilde{Y} \mid \tilde{X}, Y, X, \Phi) = p_S(\tilde{Y} \mid \tilde{X}, Y, X, \Phi)\, p_{DTW}(\tilde{Y} \mid \tilde{X}, Y, X, \Phi) \qquad (5.8)$$

where p_S is the joint likelihood of frames in $\tilde{Y}$, which is derived from the standard equation (2.55):

$$p_S(\tilde{Y} \mid \tilde{X}, Y, X, \Phi) = \prod_{i=1}^{\mathbf{N}_{\tilde{\mathbf{Y}}}} \frac{1}{(2\pi\sigma^2(\tilde{x}_i))^{\mathbf{D}/2}} \exp(-\frac{\left\|\tilde{y}_i - \mu(\tilde{x}_i)\right\|^2}{2\sigma^2(\tilde{x}_i)}) \qquad (5.9)$$

whereas p_{DTW} is the probability of predicting the entire sequence $\tilde{Y}$:

$$p_{DTW}(\tilde{Y} \mid \tilde{X}, Y, X, \Phi) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{DTW(\tilde{Y}, \mu(\tilde{X}))}{2\sigma^2}) \qquad (5.10)$$

The means $\mu(\tilde{X})$ are the sequence of frames that the model would predict for a given $\tilde{X}$, whereas the variances $\sigma^2(\tilde{X})$ indicate the uncertainty of this prediction. The means $\mu(\tilde{X})$ and variances $\sigma^2(\tilde{X})$ are expressed by equations (2.56). DTW denotes the dynamic time warping distance (see appendix A.1), whereas σ^2 is the 'strength' of the constraining prior as defined in equation (5.1). The maximisation of the above posterior (5.8) is equivalent to minimisation of the following sum of negative log likelihoods:

$$L = L_S + L_{DTW} \qquad (5.11)$$

where

$$L_S = \sum_{\tilde{x} \in \tilde{X}} (\frac{\left\|\tilde{y} - \mu(\tilde{x})\right\|^2}{2\sigma^2(\tilde{x})} + \frac{\mathbf{D}}{2} \ln \sigma^2(\tilde{x}) + \frac{\mathbf{D}}{2} \ln 2\pi) \qquad (5.12)$$

$$L_{DTW} = \frac{1}{2} \sum_{i=1}^{\mathbf{N}_{\tilde{\mathbf{Y}}}} \sum_{j=1}^{\mathbf{D}} (\frac{DTW(\tilde{Y}, \mu(\tilde{X}))}{\sigma^2} + \ln \sigma^2 + \ln 2\pi) \qquad (5.13)$$

Table 5.1 gathers the obtained results of sign language recognition for GPLVM, BC-GPLVM, ST-GPLVM and the current state of the art results for this dataset. First, the imposition of high dimensional constraints improves the recognition performance in comparison to the standard under-constrained GPLVM. Among extended methods, the increase of performance is especially noticeable for our proposed spatio-temporal formulation of GPLVM, which confirms the generalisation abilities of the proposed methodology. Unfortunately, it is not straightforward to compare our best results with the current state of the art, since not all approaches follow the same evaluation methodology. For instance, some of them use only a subset of the provided signs for evaluation, which makes a direct comparison very difficult. Nevertheless, our framework achieves the best performance when aiming at 25 signs, whereas in the case of the most exhaustive evaluations using all available signs, our ST-GPLVM still produces very competitive results. While all considered methods are tailored to hand gesture recognition and some of them rely dramatically on a set of parameters provided by the user, our methodology is general for MTS modelling, where all critical parameters are estimated automatically. In particular, the approach proposed by [Kadous and Sammut, 2005] is intrinsically of a highly supervised nature, since it is based on a set of pre-defined metafeatures designed for a specific application domain. Not only, this design process is extremely challenging [Kadous and Sammut, 2005], but Kadous' algorithm is not deterministic because of embedded randomness and thus exhibits very high variation of results [Böhm et al., 2009]. In contrast, the nature of ST-GPLVM is fundamentally different, since training only relies on training set labels and when applied for classification task decisions are stable and repeatable. Note that the best results reported by [Kadous and Sammut, 2005] are obtained when the powerful boosting classification is integrated [Schapire, 1999]. Therefore further improvement of our performance will be possible, if a more advanced classification method is applied. Since our methodology is a general concept for MTS modelling, ST-GPLVM can be adaptable to a wide array of problems beyond hand gesture recognition, for instance video based action recognition (chapter 6).

Table 5.1. Percentage accuracy of sign language recognition, where C – customised data models, S – large sensitivity to parameter choice, V – variation of results, P – probabilistic framework, D – deterministic framework, R – uncertainty regarding signs used for evaluation.

Method	Number of signs	Accuracy	Comment
ST-GPLVM	95	91%	P
TLE	95	78%	D
BC-GPLVM	95	52%	P
GPLVM	95	44%	P
Kadous [Kadous and Sammut, 2005]	95	~93%	C, V
Kadous + Boosting [Kadous and Sammut, 2005]	95	98%	C, V
Rozado [Rozado et al., 2010]	95	90%	C, D
Yang [Yang and Shahabi, 2007]	95	~90%	D
Böhm [Böhm et al., 2009]	95	75%	D
ST-GPLVM	25	97%	P
Weng [Weng and Shen, 2008a]	25	~95%	D, S
Liu [Liu and Kavakli, 2010]	25	94%	D, S
Li [Li et al., 2006] as evaluated by [Weng and Shen, 2008b]	25	89%	D
Weng [Weng and Shen, 2008b]	25	89%	D, S

Li [Li et al., 2007a] as evaluated by [Weng and Shen, 2008b]	25	88%	D
ST-GPLVM	10	97%	P
Seo [Seo et al., 2009]	10	~98%	P
Siddiqi [Siddiqi et al., 2007]	10	96%	P, R
Bicego [Bicego et al., 2009]	10	~87%~	P

5.4.5. Discussion

In all conducted experiments, the proposed spatio-temporal extension of GPLVM discovers a compact underlying probabilistic model of MTS space, where intra and inter variations between MTS are suppressed to improve generalisation properties. This is achieved by the innovative spatio-temporal constraining prior, which is imposed over a latent space within the optimisation process of GPLVM framework.

In agreement with previous experiments (section 4.5), analysis of results produced by GPVLM, BC-GPLVM and GPDM (sections 5.4.3 and 5.4.4) confirms that these methods cannot handle stylistic variations of MTS during dimensionality reduction. The introduced spatio-temporal enhancement effectively overcomes this limitation and allows learning a unique generative model of MTS (Figure 5.3). In line with other research [Lawrence and Quinonero-Candela, 2006, Urtasun et al., 2008], we have shown that incorporation of high dimensional constraints within the GPLVM framework is extremely important for a successful dimensionality reduction. However, our temporally motivated constraints are not only conceptually different, but significantly more powerful in modelling MTS as it has been shown qualitatively in section 5.4.3 and quantitatively in section 5.4.4. In particular, our constraints, which are derived from the temporal graphs (section 4.4.2.2), encapsulate both the local temporal and global spatio-temporal relations between

MTS, whereas BC-GPLVM tries to encourage only a temporal coherence of successive points using back-constrained mapping from a high to low dimensional space. In contrast to LL-GPLVM, we aim at suppressing style variability from MTS data in order to generate a unique low dimensional representation which is crucial in recognition based scenarios. Moreover, our constraints are data-driven, whereas constraints of LL-GPLVM are highly supervised, designated for a specific domain of cyclic activities, and prior knowledge about intrinsic structure of MTS space is required in order to perform dimensionality reduction.

Among other limitations of GPLVM family (section 5.2.1), ST-GPLVM is computationally more attractive and more robust against local minima (see processing times in Figure 5.3). This relies, first, on an accurate initialisation of a latent space using TLE, which is more likely to be closer to the global optimum. Secondly, the ST-GPLVM objective function is more constrained, which further mitigates the problem of local minima.

In order to further show the value of the proposed methodology in a real and challenging computer vision application, ST-GPLVM has been incoporated within a view independent action recognition framework which will be presented in chapter 6.

5.5. Summary

In this chapter, a novel spatio-temporal extension of GPLVM framework was proposed, which allows discovery of a smooth and unique low dimensional representation tailored to MTS with an associated uncertainty. As a consequence, ST-GPLVM can be deployed in various MTS classification applications such as speech, gesture and action recognition, which has not been possible until now.

This is achieved by formulating a concept of spatio-temporal conditioned prior which is placed over a latent space and constrains the optimisation process of

GPLVM. The prior is derived automatically from two complementary temporal graphs which express the local temporal and global spatio-temporal dependencies between MTS. As a result, a unique and consistent probability distribution over the space of MTS is learned in the form of generative and continuous mapping function from a low to high dimensional space.

In conclusion, qualitative and quantitative experiments proved the high quality and generalisation power of the generated low dimensional spaces. In particular, our proposed methodology has been successfully applied in a context of two real-life MTS classification applications, i.e. hand gesture (section 5.4.4) and human action (chapter 6) recognition, where the marginalisation of style variability is crucial. The very competitive results produced by the proposed methodology demonstrate its strength and potential.

6. Action Manifolds for View-Independent Action Recognition

6.1. Introduction

Since video recording devices have become ubiquitous and have increasing impact on various aspects of our lives, the automated analysis of human action from video is now one of the most active areas of research in computer vision. This growing attention is driven by a broad spectrum of promising applications such as security and visual surveillance, content-based video analysis, behavioural biometrics, human-computer interactive applications and environments, robotics, indexing of film archives and animation in the entertainment industry (e.g. games and movies).

However, action recognition is an extremely challenging problem due to large variability in a physical appearance and individual motion style, camera viewpoint, perspective and scene environment. Following the work of [Sheikh et al., 2005], we have identified three major sources that give rise to variation in observed features:

- anthropometry - morphological and biomechanical differences between individuals induced by body size, body shape, gender, mood, etc. as well as motion execution variability [Easterby et al., 1982]. It has been shown that the same action performed multiple times by the same person, or by different people, exhibits significant inter and intra disparity [Parameswaran, 2004]. All these anthropometric factors are referred to as ‘style’ in the rest of the chapter. More formally, ‘style’ is defined as a variation of a given activity or movement which does not affect its intrinsic nature.

- viewpoint – an external factor not related to the observed type of action. It is the global position in a scene from which an action is recorded by a camera. If the camera is sufficiently far from the object of interest, its position can be defined on a sphere centred on the object (Figure 6.1a). However, in practice, in the context of action recognition within the application of visual surveillance and sport analysis, the static viewpoint is located within a height range which allows defining its position within a cylinder (Figure 6.1b). We assume that perspective effects are negligible. The variation of viewpoint leads to highly different image evidence of the same action. Note that the viewpoint and camera configuration are usually not available to an action recognition system.
- execution rate – speed of movement while performing an action as well as a rate at which the action is recorded. They both have an important effect on the recorded temporal extent of an action.

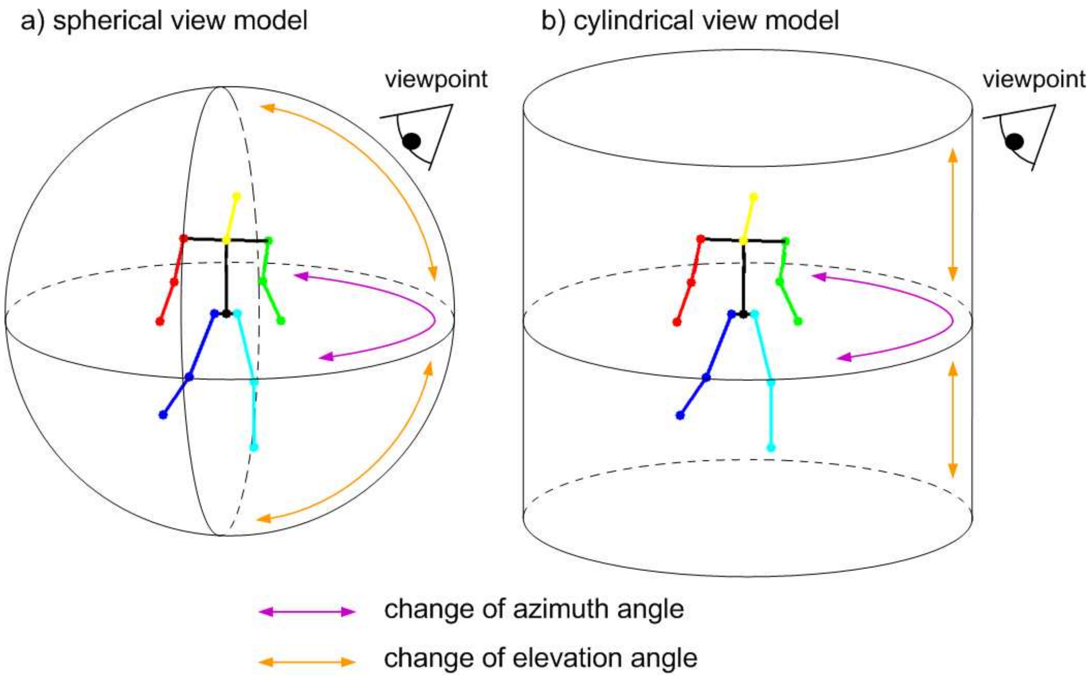


Figure 6.1. Spherical and cylindrical view models.

Any robust action recognition system should be invariant to these factors, i.e. it should be able to generalise over variations of style, view and speed within one class and distinguish between actions of different classes. This can be achieved by learning so called action models from pre-acquired training datasets. Subsequently, these action representations are used for classification of unseen action instances. However, the learning process is not trivial, since models have to be obtained from sparse training data in relation to the diversity of naturally plausible motions while avoiding over-fitting. Moreover, variability in human shape, appearance, posture, speed and individual style in a motion performance makes the unified description of a given action exceptionally difficult. In the case of a single uncalibrated camera, the lack of depth information and perspective effects make the problem of recognition even more demanding. Consequently, the task of action recognition from a single video is immensely challenging.

In this research, an innovative action descriptor is introduced, which addresses all these fundamental problems and allows accurate classification of unseen actions recorded by a single uncalibrated camera [Lewandowski et al., 2010b, Lewandowski et al., 2011]. The space of human motion is highly dimensional since the human body is a deformable object with no less than 244 degrees of freedom [Zatsiorsky, 2002], anthropometric variability [Easterby et al., 1982] and nonlinearity of human dynamics [Farnell, 1999]; however different instances of a given action reside only in a subspace of the entire feature space. Our innovative descriptor is learned by eliminating implicitly irrelevant factors, such as style and speed variability, to extract the intrinsic motion pattern of action during a dimensionality reduction process. Since temporal information is essential to characterise an action, the dimensionality reduction transformation takes into account local temporal and global spatio-temporal constraints to ensure uniqueness of the extracted motion pattern. This pattern is then generalised across different views to provide a compact and discriminative model of an action. As a consequence, we propose an intuitive and compact descriptor of human body motion which has the form of the temporally constrained Action Manifold. It is learned automatically from labelled training data and encapsulates style, view and speed variability in a coherent torus-like two-dimensional manifold (e.g. Figure 2.9c). The two intrinsic dimensions of each action correspond to style-invariant body configurations over time and view variability. The novel procedure which is used for generating this torus-like descriptor takes advantage of our contributions, which have been presented in chapters 4 and 5, and several other advanced techniques which have never been used in the context of view-independent action recognition.

First, we propose a variant of Temporal Laplacian Eigenmaps (TLE) which is tailored to human action videos. Then, our proposed action descriptor is produced by applying this natural extension of TLE to view-dependent videos in order to produce a stylistic invariant embedded manifold for each view separately. Implicitly, during dimensionality reduction, the action execution rate is normalised. Then, all view-dependent manifolds are automatically combined to discover a unified representation which models the action independently from style, speed and viewpoint in a single 3-dimensional space. In order to project actions between original and low dimensional descriptor space, the manifold continuity is approximated by either a bidirectional nonlinear mapping function [Lewandowski et al., 2010b] or an underlying probabilistic model of action (chapter 5). The proposed descriptors are validated in a challenging real-life scenario of view-independent action recognition using the IXMAS dataset (see sections 2.3.3.4.2 and 6.3), which is composed of a variety of actions seen from arbitrary camera viewpoints. Experimental results demonstrate robustness of the descriptor against style, speed and view diversity during action recognition and match the performance of most accurate action recognition methods, while overcoming their limitations.

The structure of this chapter is organised as follows. First, action recognition frameworks, which have been introduced in section 2.3.3, are put into a context of modelling view variability. Then, using the IXMAS dataset as an example (section 2.3.3.4.2), we describe in section 6.3 the main properties of an action video dataset which are required for training our action models. Without any loss of approach generality, the IXMAS dataset is used for clarity of explanation in numerous figures to illustrate key aspects of the proposed methodology (section 6.4). Furthermore, it is also used for the evaluation of our action descriptors in section 6.5. Subsequently, two procedures for automatic generation of either deterministic or probabilistic action manifolds are explained in sections 6.4.1 and 6.4.2 respectively. Next, the proposed descriptors are incorporated in an action recognition framework, which is validated quantitatively on a real dataset of human actions in section 6.5. Finally, section 6.6 concludes the chapter.

6.2. Related work

A general overview of action recognition frameworks has been provided in section 2.3.3 with a special focus on feature and action descriptors, classification methods and popular evaluation protocols. Here, we complete this earlier presentation by discussing previous work in terms of view-dependent and view-independent approaches.

6.2.1.1. View-Dependent Frameworks

View-dependent methods assume that all actions are recorded from a fixed viewpoint. The standard approach uses temporal templates such as Motion History Image [Bobick and Davis, 2001, Martinez-Contreras et al., 2009] or Motion History Histogram [Meng and Pears, 2009]. Actions have also been described in the space-time domain. Local space-time features are extracted from the volumetric space-time action shape derived from sequence silhouettes by solving the Poisson equation [Gorelick et al., 2007]. Alternatively, the structure of local 3D patches is analysed by detecting interest points in the spatio-temporal domain and extracting local descriptors, such as cuboids [Dollar et al., 2005, Zhao and Elgammal, 2008, Ta et al., 2010a, Ta et al., 2010b] or histograms of oriented gradients [Kaâniche and Brémond, 2009, Roth et al., 2009]. Moreover, by taking into account dynamics, action descriptors can be defined in terms of chaotic invariant features from joint tracking [Ali et al., 2007]. This can also be achieved by modelling the temporal development of a view-dependent action using Hidden Markov Model [Kellokumpu et al., 2008, Vezzani et al., 2010] or Conditional Random Fields [Wang and Suter, 2007b, Zhang and Gong, 2010]. Eventually, view-dependent action is represented in a low dimensional space [Wang and Suter, 2007a, Chin et al., 2007, Blackburn and Ribeiro, 2007, Wang and Suter, 2008, Jia and Yeung, 2008]. Although all these approaches have proved very accurate, the fact they rely on videos captured from a specific view limits their practicality in real world scenarios.

6.2.1.2. View-Independent Frameworks

View-independence has been addressed by two contrary approaches. In the first one, the view-independence is not directly modelled, since it is assumed that enough training data is available to adequately cover the entire space of plausible solutions. In particular, bag of words has proved to be effective in this category when applied on histograms of oriented gradients [Laptev et al., 2008, Kläser et al., 2008, Brendel and Todorovic, 2010, Ikizler-Cinbis and Sclaroff, 2010, Matikainen et al., 2010, Satkin and Hebert, 2010]. However, these approaches do not model any intrinsic structure of action and their learning processes rely only on image evidence. As a consequence, the general robustness of such action models is limited by image variability and, therefore, in practice training and testing data are expected to have a common origin.

On the other hand, viewpoints can be modelled explicitly to allow view-independency in an action representation. Many researchers focused on multiple camera systems to achieve view-invariant action recognition. For instance, 2D temporal templates are extended into 3D motion history volumes [Weinland et al., 2007]. If point correspondences between actions are assumed to be known, then either epipolar geometry [Yilmaz and Shah, 2005] or projective invariants of coplanar landmark points are exploited [Parameswaran and Chellappa, 2006]. Alternatively, an action and its view variability are represented using Stiefel and Grassmann manifolds [Turaga et al., 2008b] or a circular representation of volumetric data [Pehlivan and Duygulu, 2010]. The main drawback of these methods is that, since they all require multiple cameras setups, they can only be applied in a controlled environment.

More recently, research has tackled the task of action recognition from an arbitrary view, i.e. from a single video, where multi camera data are used for training. Typically, a database of exemplars from different views is created to recognise actions based on the best matching score. Normally silhouettes are used to represent an action. However their intrinsic ambiguity leads to a high density sampling of the view space [Ogale et al., 2005] or the requirement of supervised learning of a distance metric [Tran and Sorokin, 2008] to obtain accurate results. In contrast, richer action descriptors based on 3D exemplars represented by visual hulls and Hidden Markov Model allow reducing significantly the size of action templates [Weinland et al., 2007]. In this case, consequently, matching between observation and exemplars has to be performed in 2D by projecting visual hulls. Since such projection from high dimensional space to low dimensional maps to several possibilities, it impacts on the quality of the recognition rate [Weinland et al., 2007]. Junejo et al. [Junejo et al., 2008] propose to represent image sequences using self-similarity based descriptors which are fairly stable under view variation and characterises well the dynamics of a scene. However, this approach relies on coarse localisation and tracking of people in the video [Junejo et al., 2008]. In [Yan et al., 2008], a video is represented by the combination of 3D visual hulls with spatio-temporal volumes to build 4-dimensional action feature models. Alternatively, a video can be described as a bag of spatio-temporal features called video-words (see section 2.3.3.2.3) by quantising extracted 3D points of interest [Dollar et al., 2005]. For instance, bag of cuboids is used to train a support vector machine [Liu and Shah, 2008] or a Feature-tree [Reddy et al., 2010], alternatively the support vector machine is trained on multiple features, i.e. cuboids and spin-images [Liu et al., 2008]. In a similar vein, histograms of oriented gradients have been deployed [Kaâniche and Brémond, 2010, Weinland et al., 2010b]. Although these schemes perform accurate action recognition, the absence of generative action models limits their applicability and scalibility. Another interesting approach is to represent actions and view change as graphs. For instance, [Lv and Nevatia, 2007] introduce Action Nets that uses keyposes of actions rendered from multiple viewpoints for view-invariant action recogntion, where transitions between views and poses are encoded explicitly. In contast, in [Natarajan and Nevatia, 2008], a two layer graph model of an action is proposed where Conditional Random Fields are used to encode the action and the viewpoint-specific pose observation. Unfortunately, a large amount of motion capture data is required for training both approaches.

The methods most closely related to our approach model actions by reducing dimensionality of each sequence to obtain view-invariant manifold representations. [Richard and Kyle, 2009] uses R-transform as a descriptor and Isomap for dimensionality reduction (see section 2.2.2.2.2.2), whereas [Elgammal and Lee, 2004b, Elgammal and Lee, 2009] choose an implicit distance function representation and locally linear embedding (see section 2.2.2.2.2.2). In these approaches [Richard and Kyle, 2009, Elgammal and Lee, 2009], generative view-independent functions are designed to interpolate between intermediate views. This generative function is extended to handle also stylistic variation of data [Elgammal and Lee, 2004b, Elgammal and Lee, 2009]. However, due to the limitations of the chosen dimensionality reduction methods, none of these approaches managed to produce consistent style invariant representations, i.e. representations which are valid for a variety of individuals. Consequently, the accuracy of their systems is limited. This problem can be addressed by applying non-rigid transformation [Myronenko et al., 2007] to artificially unify manifold representations of different people [Elgammal and Lee, 2004b, Richard and Kyle, 2009]. However, since such transformation affects manifold geometry, they may no longer reflect relationships between points in the high dimensional space. Alternatively, in [Elgammal and Lee, 2009] the topological structure of a torus is artificially constrained on the manifold to explicitly deal with stylistic variation instead of being learned from the data. The problem within this approach is that it artificially enforced embedded representation may not adequately reflect relationships and intrinsic properties of high dimensional features.

6.3. Dataset Characteristics

Although, the proposed action manifolds can be used for classification of any monocular action video, the framework requires a specific type of video data for learning. First, the ideal dataset should provide a ‘satisfactory’ amount of training data for all actions, which may appear during the recognition process. Moreover, each action should be repeated a number of times by different subjects, thus providing ‘sufficient’ information for the framework to extrapolate stylistic variability to unknown subjects. Then, each action is expected to be recorded by synchronous multiple cameras in order to generalise view variability in action descriptors. In practice, only a few viewpoints are required to allow reconstruction of 3D visual hulls [Cheung et al., 2005] for each action. They can latter be employed for synthesising more dense and evenly spread training data across different azimuths and elevations of views (see section 6.5.1) according to the cylindrical model (Figure 6.1b).

In this research, without any loss of generality, the publicly available multi-view IXMAS dataset is used (section 2.3.3.4.2), since it is the only available currently dataset providing sufficient training data for view-independent action recognition applications. Moreover, it is considered as the well known benchmark for view-independent action recognition methods by the research community. This dataset comprises of 13 actions, performed 3 times by 12 different actors. Each of these 468 activity instances was recorded simultaneously by 5 calibrated cameras. This dataset is very challenging for two main reasons. First, it exhibits large style variability because of the number of available subjects, who perform action repetitions in a various ways. Secondly, since no specific instruction was given during acquisition, actors' chose freely their positions and orientations for each repetition. As a consequence, the action viewpoints are arbitrary and unknown.

6.4. Proposed Frameworks

Our framework is based on a novel compact and discriminative action descriptor, called Action Manifold, which accounts for variability that arises when cameras at arbitrary positions capture different people performing the same action. The descriptor is learned on a set of videos of action primitives performed by a variety of individuals, each of them captured on their own by a set of calibrated and synchronised cameras. In addition, for each action, one video can be labelled as a good representative, i.e. the most visually discriminative one,in order to speed up the dimensionality reduction process as we will explain in section 6.4.1.1.3. Usually the representative video is captured from a side view. Note that this is the optional step which simplifies processing requirements of the proposed prototype, however it can be easly automated (see section 6.4.1.1.3 for more details). We do not impose any restrictions regarding the number of video frames for a given action primitive. Moreover, an individual may perform an action several times in an arbitrary manner and at different speeds. Our action manifold is considered to be a high level semantic description of the action. Therefore, in agreement with current research in the field (see section 2.3.3.1.2), we assume that a person localisation and segmentation, as well as a temporal segmentation of action into primitives can be carried out sensibly by some low level pre-processing of video data. Testing is performed using examples of unknown action primitives performed by unknown people captured from an arbitrary and unknown view. Note that multi-camera setup is required only for the learning of action models, whereas the testing can be done using either single or multiple views.

Let Y denotes the set of $\mathbf{N}$ videos defining an action primitive performed by different people and captured from different views. For a given view, action repetitions and variability of people define action style. Therefore, Y is defined as $Y = \{Y^{sv} \mid s = 1..\mathbf{N_s}, v = 1..\mathbf{N_v}\}$, where s denotes the style index and v is the view class index. Each frame y of a video is represented by $\mathbf{D}$ pixels of region of interest (see section 2.3.3.1.2): $Y^{sv} = \{y_i^{sv} \mid y_i^{sv} \in \mathbb{R}^{\mathbf{D}}, i = 1..\mathbf{T^{sv}}\}$, where $\mathbf{T^{sv}}$ is the number of frames in the sequence. A unified and compact action model, X , of dimension $\mathbf{d} << \mathbf{D}$, is defined by $X = \{X^{sv} \mid s = 1..\mathbf{N_s}, v = 1..\mathbf{N_v}\}$, where $X^{sv} = \{x_i^{sv} \mid x_i^{sv} \in \mathbb{R}^{\mathbf{d}}, i = 1..\mathbf{T^{sv}}\}$.

The proposed descriptor is of either a deterministic or a probabilistic nature. First, the deterministic variant is introduced in section 6.4.1 [Lewandowski et al., 2010b] followed by the probabilistic formulation in section 6.4.2

[Lewandowski et al., 2011]. The summary of our contribution is given in section 6.4.3.

6.4.1. Deterministic Action Model

The descriptor learning procedure is divided into two parts. First, view-dependent analysis of action data generates a style invariant action model for each view. This is performed using Temporal Laplacian Eigenmaps (chapter 4), which is capable to generalise a space of multidimensional time series (e.g. actions) in the presence of stylistic variations (i.e. different people perform repetitions of the same action). Then, these models are combined to produce a compact and view invariant model of the action. Finally, continuity of the descriptor is approximated by learning a generative decomposable model [Lee and Elgammal, 2006a]. Figure 6.2 summarises the processing pipeline for the generation of a deterministic action model.

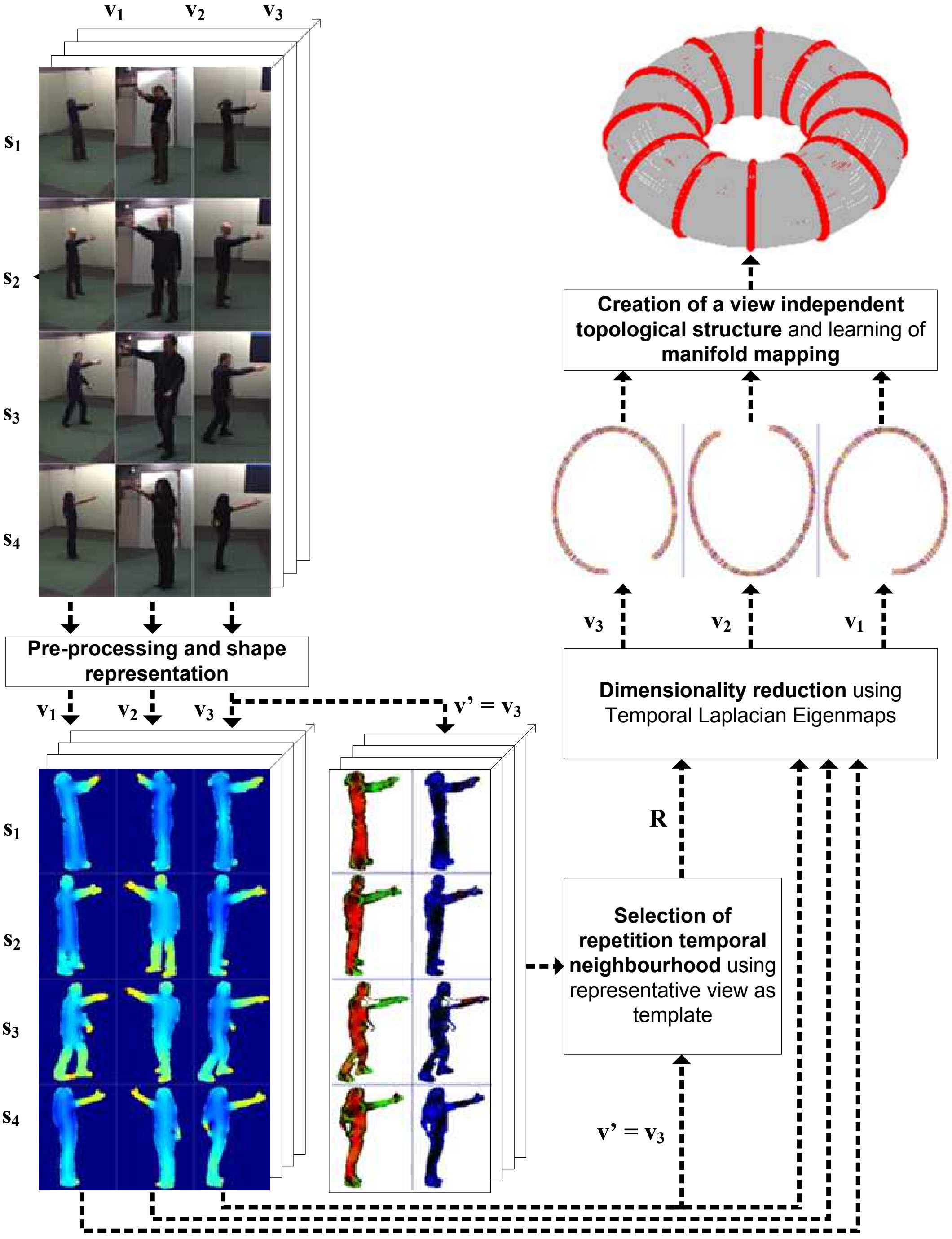

v1
v2
v3
s1
s2
s3
s4
Pre-processing and shape representation
v1
v2
v3
v' = v3
s1
s2
s3
s4
Creation of a view independent topological structure and learning of manifold mapping
v3
v2
v1
Dimensionality reduction using Temporal Laplacian Eigenmaps
R
Selection of repetition temporal neighbourhood using representative view as template
v' = v3

Figure 6.2. Description of the action recognition framework for the 'point' action.

6.4.1.1. View-Dependent Manifold

6.4.1.1.1. Pre-processing and Shape Representation

A frame of video is generally defined by grey scale or colour pixel values. This very high dimensional description makes the process of learning an activity model from a frame sequence costly and inaccurate. However, many studies (see section 2.3.3.1.2) have revealed that a binary representation of moving objects, i.e. silhouettes, is sufficient to capture the activity described by a frame sequence. Consequently, we adopt the space-time extension of binary silhouette representation in our framework (see section 2.3.3.1.2.3).

We extract the region of interest and corresponding binary silhouette y_i^{sv} from each video by a standard background subtraction technique which models each pixel as a Gaussian in RGB space [Weinland et al., 2006b] followed by a frame cropping. When videos consist of multiple instances of a given motion, temporal segmentation is required to extract elementary motion segments Y^{sv} [Cutler and Davis, 2000, Rui and Anandan, 2002, Weinland et al., 2006a]. Here we assume that videos have been segmented.

All silhouettes are normalised to deal with translation and scale variations by using the largest silhouette square bounding box available within the entire action dataset. In order to improve the quality of the normalised silhouettes, two morphological operations, i.e. bridge and open, and a median filter are applied. Lengths of all sequences Y^{sv} are also normalised to match the length of the shortest sequence T' in the set Y using standard bicubic spline interpolation to reduce computational cost by cutting the number of training and testing frames.

A sequence of binary silhouettes can be considered as a space-time shape surrounded by a closed surface (Figure 6.3a, see also section 2.3.3.1.2.3). This allows representing each silhouette by a local space-time saliency feature (Figure

6.3c) extracted from the solution of the Poisson equation of the corresponding volumetric surface S (Figure 6.3b), which takes into account the time domain [Gorelick et al., 2007]:

$$\Delta U(p_x, p_y, t) = -1 \tag{6.1}$$

with $(p_x, p_y, t) \in S$, where the Laplacian of U is defined as $\Delta U = U_{xx} + U_{yy} + U_{tt}$ subject to the Dirichlet boundary conditions $U(p_x, p_y, t) = 0$ at the bounding surface ∂S. The space-time saliency feature is defined by the function w at every pixel (p_x, p_y, t) in shape S [Gorelick et al., 2007] according to the following equation:

$$w(p_x, p_y, t) = 1 - \frac{\log(1 + U + 1.5\|\nabla U\|^2)}{\max\limits_{(p_x, p_y, t) \in S} \log(1 + U + 1.5\|\nabla U\|^2)} \tag{6.2}$$

This representation assigns highest gradient values within fast moving limbs which are usually much more informative for identifying actions, whereas the torso has relatively smaller values inside (Figure 6.3c). As a consequence, such descriptor is significantly more powerful than binary representation [Gorelick et al., 2007]. As shown later in section 6.4.1.1.3, this descriptor is also essential in the procedure allowing the selection of the TLE repetition neighbourhoods. The generated shape descriptor is 3364-dimensional (58×58 pixels).

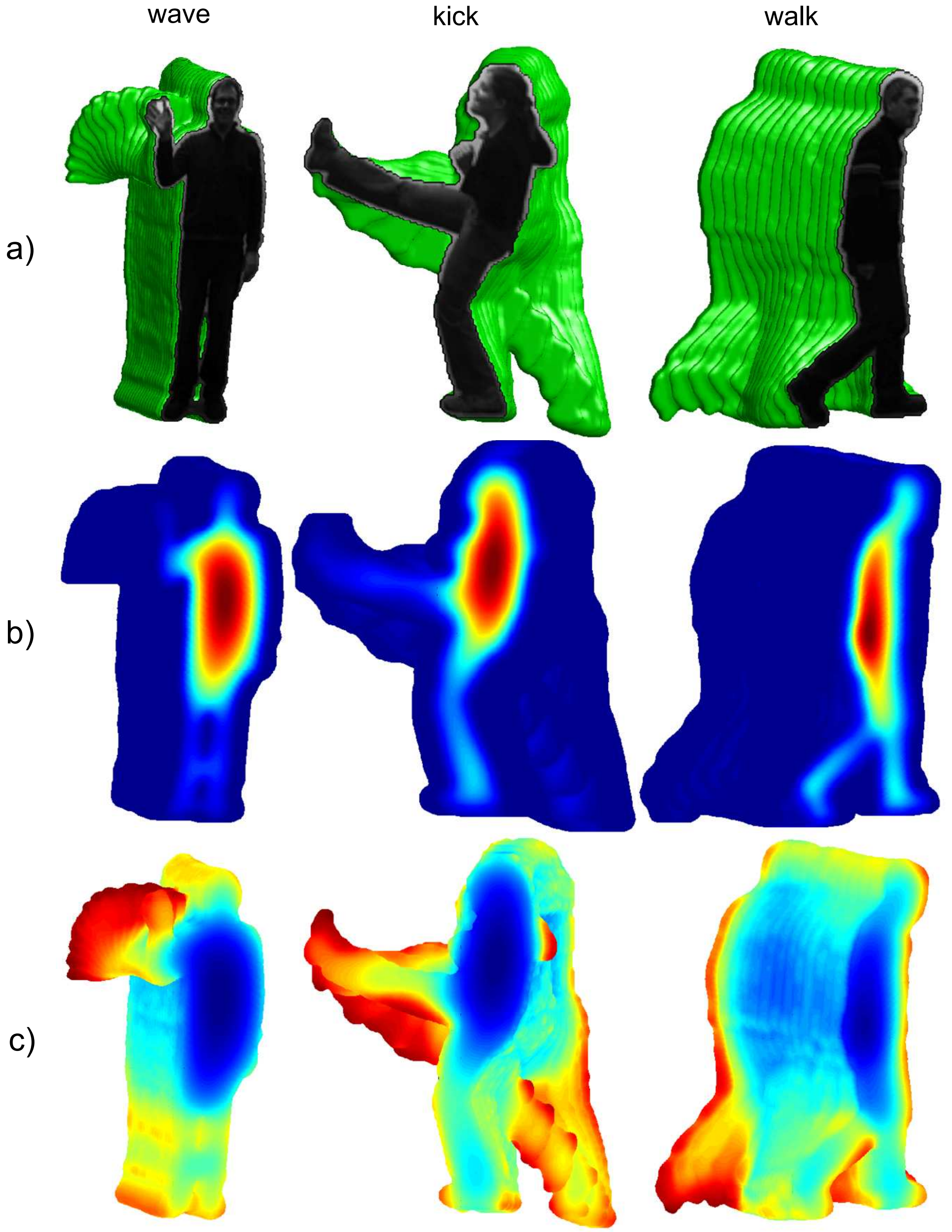


Figure 6.3. Extractions of shape representation for 'wave', 'kick', 'walk' actions: a) space-time shapes; b) the solution to the Poisson equation on space-time shapes; and c) the local space-time saliency features. The values in b,c are encoded using a colour spectrum from blue (low values) to red (high values).

6.4.1.1.2. Dimensionality Reduction

Even after the generation of shape descriptors, the high dimensionality of the feature space Y prevents obtaining a descriptive action representation. Consequently, our model of action is produced by nonlinear dimensionality reduction of feature space using a variant of TLE (chapter 4), which is tailored to human action videos. The standard TLE extracts unique and descriptive pattern from MTS and, at the same time, suppresses stylistic variability. Moreover, MTS are implicitly aligned along the time axis as a result of spatio-temporal detection of repetition neighbours in different MTS. In the context of action recognition, any action primitive is considered to be a single MTS, whereas different subjects correspond to different sources of MTS. The alignment of MTS results in the speed normalisation of the action. Here, the standard TLE is extended by using a more advanced repetition neighbourhood estimation procedure which is designed to deal with human action videos. It is described in the subsequent section 6.4.1.1.3.

The dimensionality reduction transformation is applied in each view independently, i.e. for each Y^v. As a result, a set of style-invariant but view-dependent action models X^v is obtained. In this set, each action model is a 1-dimensional manifold embedded in 2-dimensional space to model the nonlinearity of human motion. The intrinsic dimension of each action corresponds to innate configuration of motion over time.

6.4.1.1.3. Selection of Repetition Temporal Neighbourhood

Successful dimensionality reduction using TLE depends on the appropriate identification of repetition neighbours for each frame. The repetition temporal neighbourhood corresponds to the number of times an action is repeated in the training set. Although video lengths are normalised for each action, it cannot be assumed that these videos are synchronous. Firstly, they may start with different postures and, secondly, due to style and speed variations, there may not be frame to frame correspondences between two action instances. Consequently, the estimation of the size and location of the repetition neighbourhood is essential.

In order to take full advantage of the shape descriptors generated in section 6.4.1.1.1, we propose an advanced method for automatic determination of repetition neighbourhoods which is tailored for dimensionality reduction of human action videos. This is achieved by adopting the action detection method proposed in [Gorelick et al., 2007] to formulate a procedure which is conceptually equivalent to the usage of DTW metric (see section 4.4.2.1). Since a high dimensional human motion pattern in a video is theoretically equivalent to a high dimensional curvature of time series fragment, the identification process of time series fragments can be seen as a detection process of similar motion patterns in each video of the training set in the context of human action data. Therefore, repetition neighbours can be extracted from each detected motion pattern in a manner similar to the one used in the case of time series fragments (section 4.4.2.1). This new schema can be straightforwardly deployed within the TLE framework without any modifications of the algorithm core. The key advantage of the new procedure is its computational efficiency in comparison to DTW (see section 6.4.1.1.4).

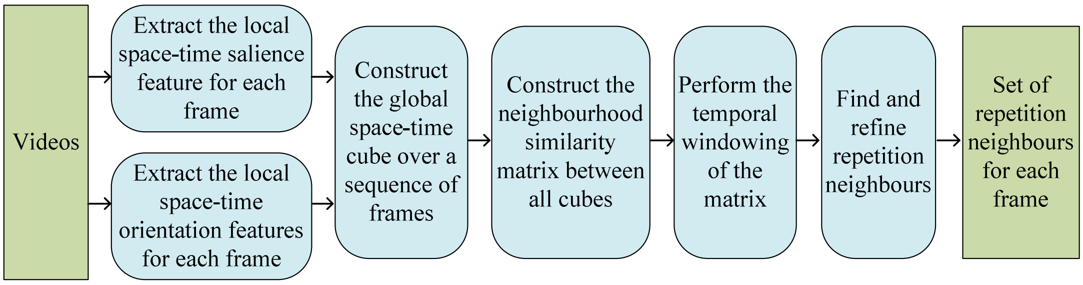

Videos
Extract the local space-time salience feature for each frame
Extract the local space-time orientation features for each frame
Construct the global space-time cube over a sequence of frames
Construct the neighbourhood similarity matrix between all cubes
Perform the temporal windowing of the matrix
Find and refine repetition neighbours
Set of repetition neighbours for each frame

Figure 6.4. Successive steps of the repetition neighbourhood selection procedure tailored to human action videos.

The adaptation of the motion detection procedure [Gorelick et al., 2007] to form the alternative variant of DTW metric for TLE is summarised as follows

(Figure 6.4). First, the local space-time saliency shape descriptor defined in section 4.5.7.1 (Figure 6.5b) is extended with 6 local space-time orientation features (Figure 6.5c,d) [Gorelick et al., 2007]. This allows indentifying regions with vertical, horizontal, and temporal 'plates' and 'sticks' within bodies and define orientation local features. Figure 6.2 and Figure 6.5c,d illustrate examples of 'plate' and 'stick' local features for a good representative view. Blue, red, and green colour regions correspond to temporal, horizontal, and vertical directions of local 'plates' and 'sticks' within a human shape. These features are computed from the 3×3 Hessian H of the solution to the Poisson equation (6.1) at every pixel (p_x, p_y, t), where its eigenvectors correspond to the local principal orientation directions and the corresponding eigenvalues λ are related to the local curvature in the direction of the eigenvectors. The 'stick' is the informative direction which corresponds to the third eigenvector of H, whereas the 'plate' corresponds to the first eigenvector. The space-time orientation feature is defined by the function w at every pixel (p_x, p_y, t) in the shape S [Gorelick et al., 2007] according to the following equation:

$$w_{ij}(p_x, p_y, t) = R_i(p_x, p_y, t) * D_j(p_x, p_y, t) \tag{6.3}$$

where deviations D_j of the informative direction $v(p_x, p_y, t)$ is measured by $D_j = \left|v(p_x, p_y, t) * e_j\right|$ with e_j denoting the unit vectors in the direction of the principal axes ($j \in \{1, 2, 3\}$ respectively horizontal x, vertical y and temporal t direction). In turn R_i is a continuous measure of 'plateness' (pl) or 'stickness' (st) at every space-time point ($i \in \{st, pl\}$):

$$\begin{aligned} R_{pl} &= e^{\alpha\lambda_2/\lambda_1} \\ R_{st} &= (1 - R_{pl})e^{\alpha\lambda_3/\lambda_2} \end{aligned} \tag{6.4}$$

Further on, a space-time cube is associated to each frame y^{sv} in a view-dependent sequence Y^v by sliding a warping window in time. The cube, i.e. the global space-time descriptor, combines local shape and orientations features within a window using weighted moments of the form:

$$m_{oqr} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} w(p_x, p_y, t) g(p_x, p_y, t) p_x^o p_y^q t^r dp_x dp_y dt \qquad (6.5)$$

where p_x, p_y are pixels coordinates, $g(p_x, p_y, t)$ denotes the characteristic function of the space-time shape, $w(p_x, p_y, t)$ is one of the seven possible weighting functionswhich corresponds to local features, i.e. the space-time saliency feature (equation (6.2)) and 6 space-time orientations features (equation (6.3)). As suggested in [Gorelick et al., 2007], spatial and time moments are considered up to order $m_s = 2$ and $m_t = 2$ respectively with $\mathrm{o} + \mathrm{q} \le m_s$ and $r \le m_t$ in equation (6.5). Each space-time cube is centred around its space-time centroid and uniformly scaled to preserve spatial aspect ratio. The dimensionality of each space-time cube equals 126 ($7 \times (m_t + 1) \times (0.5 * (m_s + 1) * (m_s + 2))$).

The obtained global space-time descriptor is a compact and temporally constrained representation of time series fragment. Since, now, each time series fragment is expressed by a single feature vector, the similarity between them is computed effectively using the standard Euclidean norm without the need of computationally expensive temporal alignment of points sequences like in DTW (see section 6.4.1.1.4).

Therefore, a neighbourhood similarity matrix E ($N_v \times N_v$) of Euclidean distances is calculated between all space-times cubes among all sequences for a particular view (Figure 6.6 and right part of Figure 6.8). To emphasise continuity and temporal coherence of the underlying action between sequentially adjacent points in time, we perform temporal windowing of matrix E by averaging distances through time within boundaries of each sequence similarly to the standard procedure (see section 4.4.2.1). This implicitly leads to introducing a temporal history into each data point.

Finally, for each cube we look for the most similar motion pattern in each different repetition of activity based on E according to the standard procedure in section 4.4.2.1. The centre point of each most similar space-time cube becomes a repetition neighbour.

Because of possible substantial differences in speed and imperfect segmentation of action, the repetition neighbours may still not align coherently along time which may result in distortions in the embedded space. To address this problem, we incorporate an optional neighbourhood refinement procedure. In principle, for given point P, we accept only these R neighbours which are within a specific range from a corresponding point in each other sequence:

$$R' = \{P_{(i-1)*T+1} - T' \leq R_j \leq P_{iT} + T'\}, i = 2..N_s, j = 2..N_s \qquad (6.6)$$

where T' is defined as 10% of the normalised sequence length T.

The entire procedure of repetition neighbourhood estimation is performed only once per action for the most discriminative view $Y^{v}{}'$. Since, all view-dependent frames were captured at the same time instants (synchronised cameras), the temporal structure of an action is a view-independent property, which is valid across all views. Since our shape descriptor is derived from a silhouette, there is a view where image evidence facilitates the most the estimation of temporal constraints. For instance, action ‘point’ in the front view provides a very small amount of meaningful information about temporal structure of action, whereas the side view is significantly more informative (Figure 6.5). As it has been mentioned earlier, we assume that one view for each action in the training set is manually labelled as the most discriminative, for most actions it is intuitively the side view. The neighbourhood estimation procedure is then carried out on this view (Figure 6.6). Afterwards, the obtained constraints in the form of temporal and spatio-temporal neighbourhood relations are employed to determine neighbourhoods in all remaining views (Figure 6.7).

Note that the selection of the most discriminative view could be automated by applying the discussed neighbourhood selection procedure in each view independently and then choosing the neighbourhood consensus between them for each frame. Such automatic procedure for the automatic selection of most discriminative view can be considered as a future work. Here, in practice, the choice of the most discriminative view is a very intuitive and simple operation for a human user, who will have already generated the training dataset. Consequently, the manual option is applied in this work for two reasons. First, it reduces computational cost of the learning process. Secondly, it allows generation of visually convincing models which would not suffer from noisy neighbourhoods, which may sometimes be created as result of large disagreement between different views.

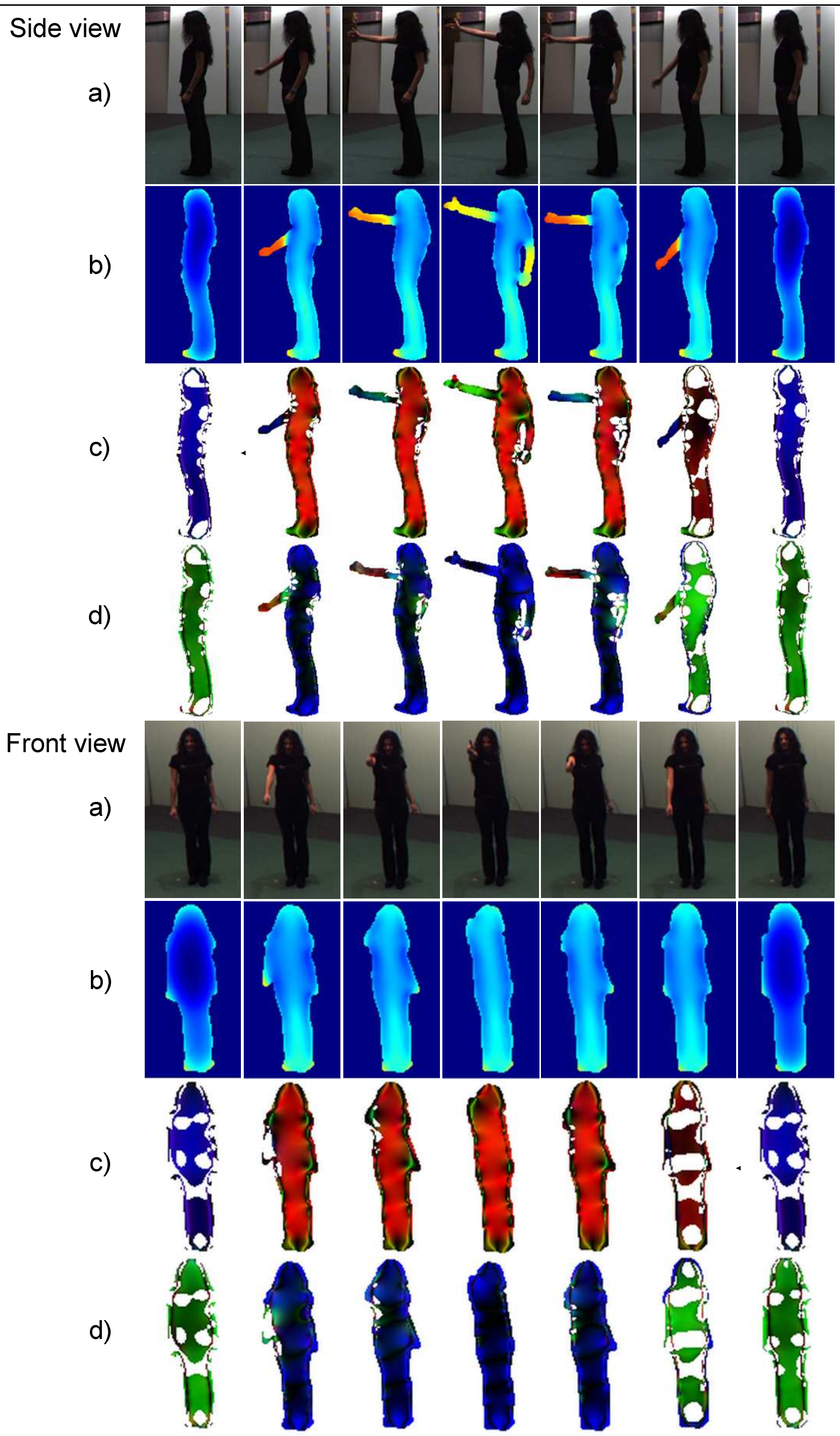


Figure 6.5. Original video (a), space-time saliency features (b), space-time orientations of plates (c) and sticks (d) for the 'point' action in the side and front view. The side view is a representative view, since it exhibits more temporal information about the action (a larger variation of colours in all local features).

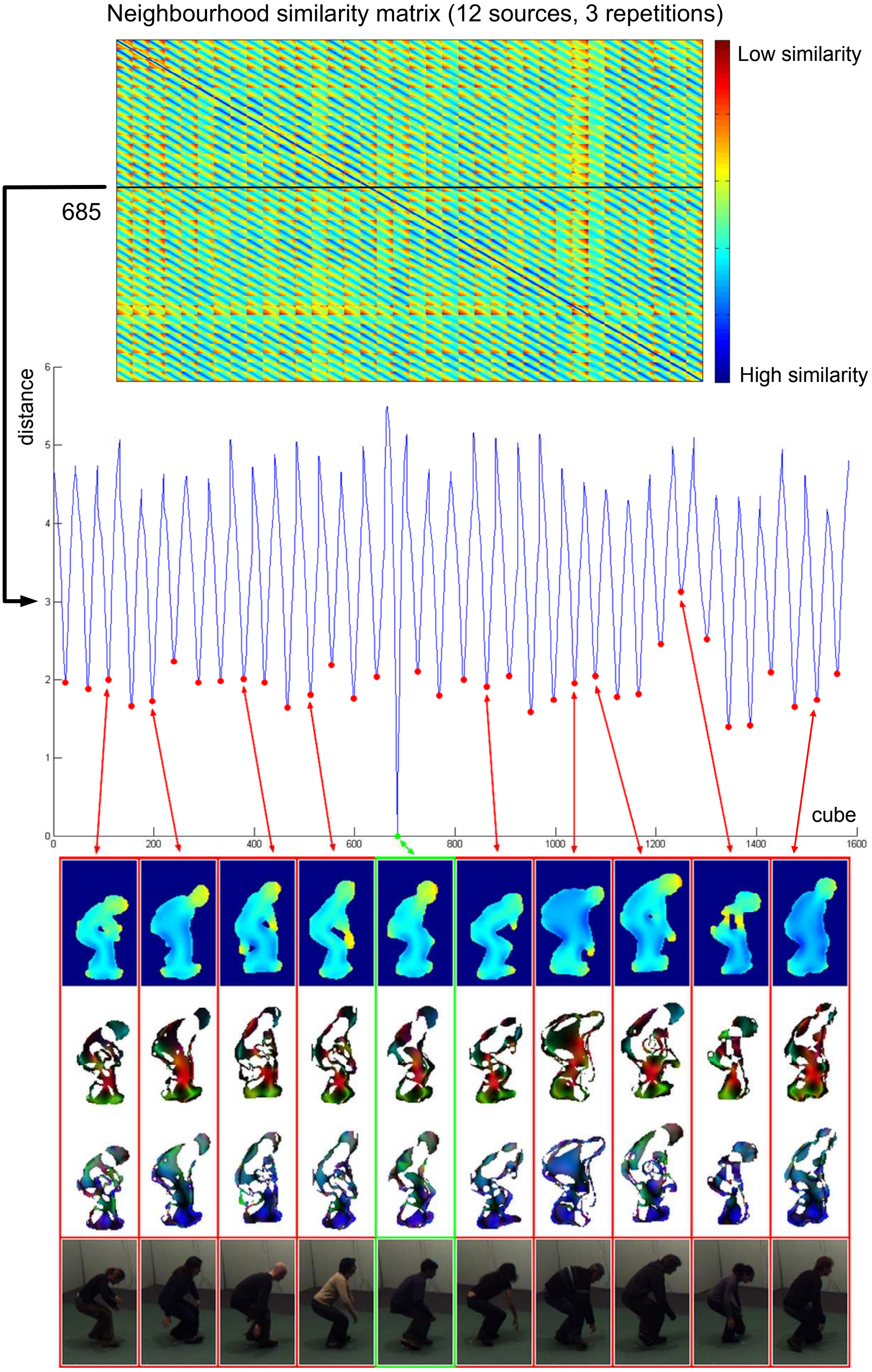


Figure 6.6. Example of neighbourhood similarity matrix created by the TLE using the action ‘sit down’ and 12 subjects with a 3 repetitions each for the side view. Each local minima corresponds to the most similar repetition neighbour in relation to the reference pose (green) extracted from different repetitions of the time series.

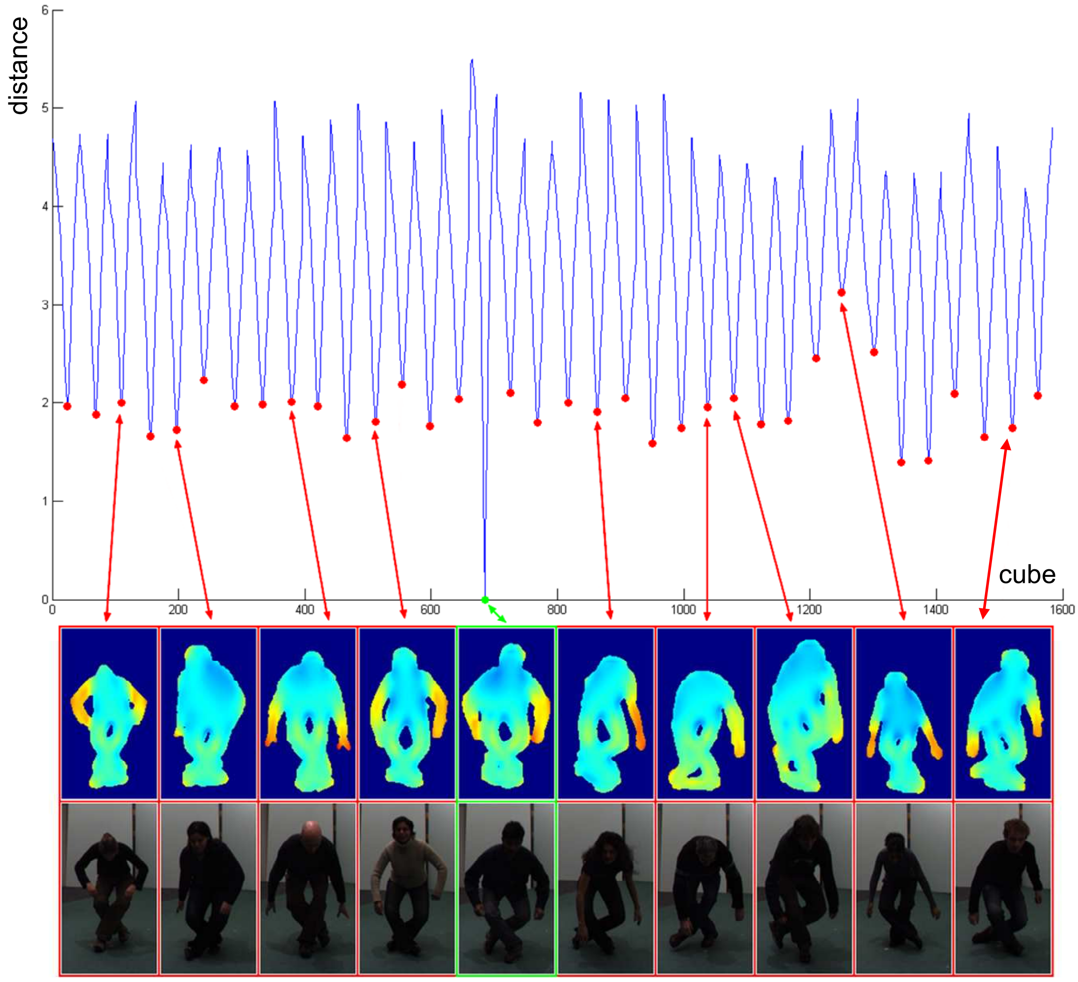
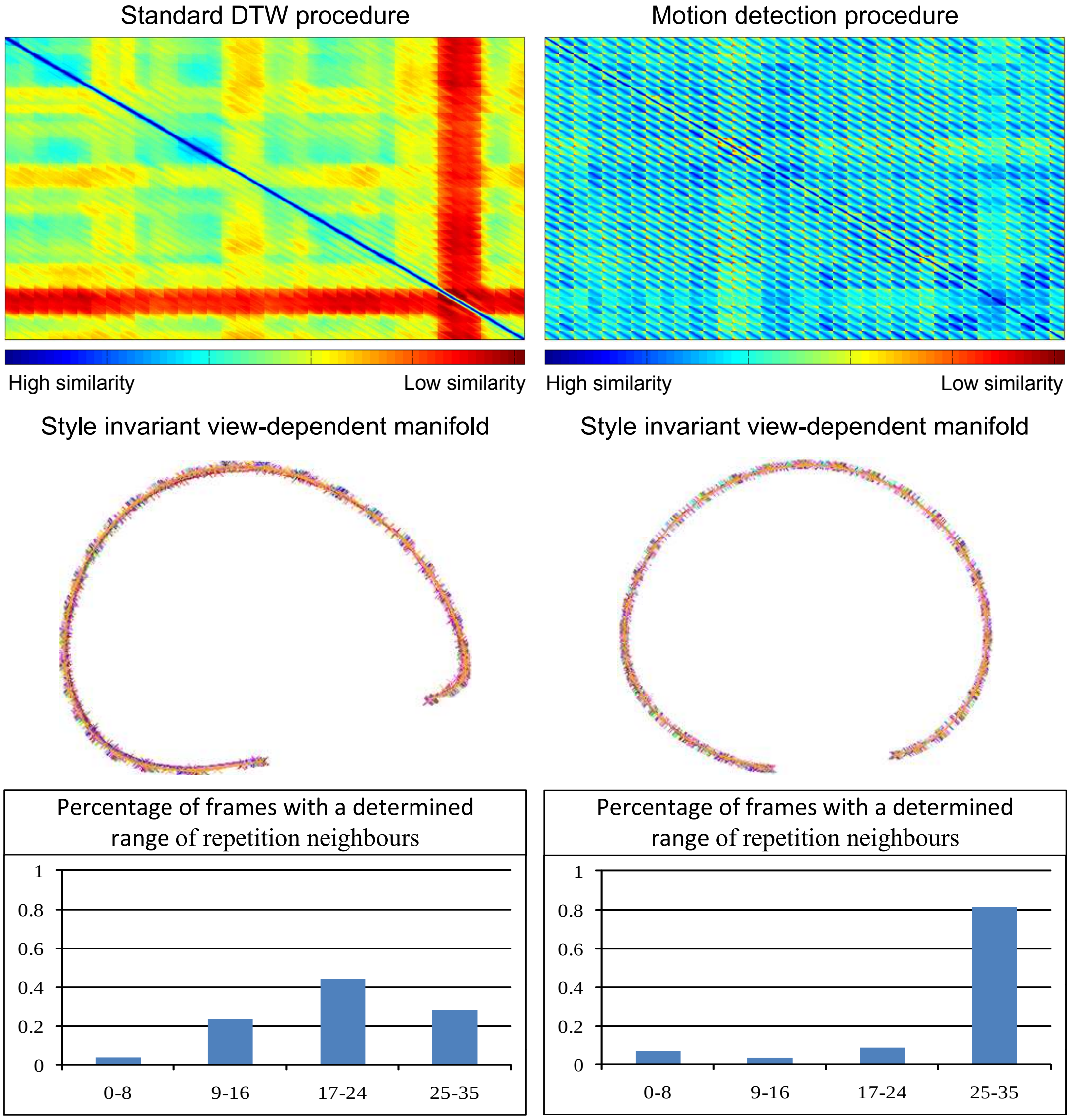


Figure 6.7. Example of repetition neighbourhood in the front view obtained from the side view similarity matrix created by the TLE for the action 'sit down' using 12 subjects with a 3 repetitions each (Figure 6.6). Each local minima corresponds to the most similar repetition neighbour in relation to the reference pose (green) extracted from different repetitions of the time series.

6.4.1.1.4. Comparison of Repetition Neighbourhood Selection Procedures

Figure 6.8 presents two examples of style invariant view-dependent manifolds of the 'wave' action (here front view) and the associated neighbourhood similarity matrices generated by the standard DTW based procedure (left) (section 4.4.2.1) and the proposed motion detection schema (right). These matrices are calculated for 12 sources and 3 repetitions of each source. Darker colours correspond to small distances between time series fragments, whereas brighter colours express larger dissimilarities. In principle, the ideal identification of the most similar time series fragments in different repetitions should result in a uniform spread of colours with clear local minima in approximately diagonal directions, which are used to extract repetition neighbours.



Time	147 min	Time	13 min

Figure 6.8. Neighbourhood similarity matrices for the ‘wave’ action in the front view, which are computed using the standard DTW procedure and the proposed motion detection procedure with the corresponding processing times and discovered style-invariant view-dependent manifolds using TLE.

As seen in Figure 6.8, the DTW metric is not always capable of localising appropriate repetitions of the action, which results in the high variation of distances between time series fragments in the neighbourhood similarity matrix. In particular, one subject performs the ‘wave’ action in a completely different manner (red area in the matrix). The poorer performance of DTW is directly related to the high dimensionality of feature space ($\mathbf{D} = 3364$). The process of DTW alignment is based on the Euclidean distance (see appendix A.1), however because of the ‘concentration phenomenon’ (see section 2.2), this metric is not always suitable to measure similarity in very high dimensional spaces. This drawback in combination with natural style variability between subjects results in difficulties in assessing pair wise similarity between frames and implicitly makes the process of DTW alignment more problematic and inaccurate. In contrast, the proposed motion detection schema is more robust against style variability and capable of recognising similarity even in very challenging cases. Although, the final stage of this procedure also relies on Euclidean distance, the dimensionality of space-time cube ($\mathbf{D} = 126$) is much lower than that of the shape descriptor ($\mathbf{D} = 3364$), thus the metric is expected to be significantly more accurate in the evaluation of similarity.

Tables in Figure 6.8 summarises the number of identified repetition neighbours for each procedure after refinement using equation (6.6). Ideally, in the case of minor style variability and perfect frame to frame correspondence between different repetitions of action, we would expect to determine 35 neighbours for each frame (i.e. 12 sources multiple by 3 repetitions minus the current MTS fragment). In practice, a repetition neighbour may not exist for a given frame, because of style and speed variability between different repetitions. Nevertheless, in most cases, the more repetition neighbours are obtained after refinement, the more constraints are available and, therefore, the better is the alignment of time series fragments during dimensionality reduction.

Finally, as seen in Figure 6.8, both algorithms produce enough constraints for TLE to discover a unique representation of time series. However, besides better detection accuracy, the key advantage of the new procedure is its very low computational cost in comparison to the DTW. Here, we achieve a speed-up of a factor ~10. The processing times reported in Figure 6.8 take into account times for computing space-time features, neighbourhood similarity matrix and extraction of repetition neighbours.

At the same time, we have shown another interesting property of the TLE framework. The proposed algorithm can discover a consistent and meaningful low dimensional representation even though for considerable number of points not all repetition neighbours are found (left table of Figure 6.8).

6.4.1.2. View-Independent Manifold

6.4.1.2.1. Generation of a View-Independent Topological Structure

Discovery of a compact representation of any human activity requires modelling both the view and body configuration jointly in a single space. Here we assume that human motion is observed from different viewpoints along a view circle at fixed camera height (Figure 6.1b). Although such cylindrical setting appears limited, its robustness to view elevation variations, up to 45 degrees as shown in the experimental section, makes it appropriate for many real-life applications such as visual surveillance and sport analysis. It is important to note that this configuration is not critical to our framework since it can easily be extended to a full view sphere-like model using training videos captured from different camera heights.

In section 6.4.1.1.2, style invariant and speed normalised body configuration manifolds could be discovered for each view separately (Figure 6.9a, Figure 6.10b and Figure 6.11b). They are intrinsically 1-dimensional manifolds, which are embedded in 2-dimensional spaces to take into account the nonlinearity of human motion. Since these embedded spaces share the same topology regardless of the view (see Figure 6.2, Figure 6.9a, Figure 6.10b and Figure 6.11b), for a given posture there is a unique correspondence on each of these manifolds. Consequently, the connection of those corresponding points in the order of view angle values creates a closed 1-dimensional manifold (topologically equivalent to a circle) which is the view-independent embedded space of the posture. Therefore, we define the unified representation of an action as the combined space of the two sets of continuous 1-dimensional manifolds, i.e. style invariant posture and view, which are placed orthogonally to each other and embedded nonlinearly in a 3-dimensional space (Figure 6.9c).

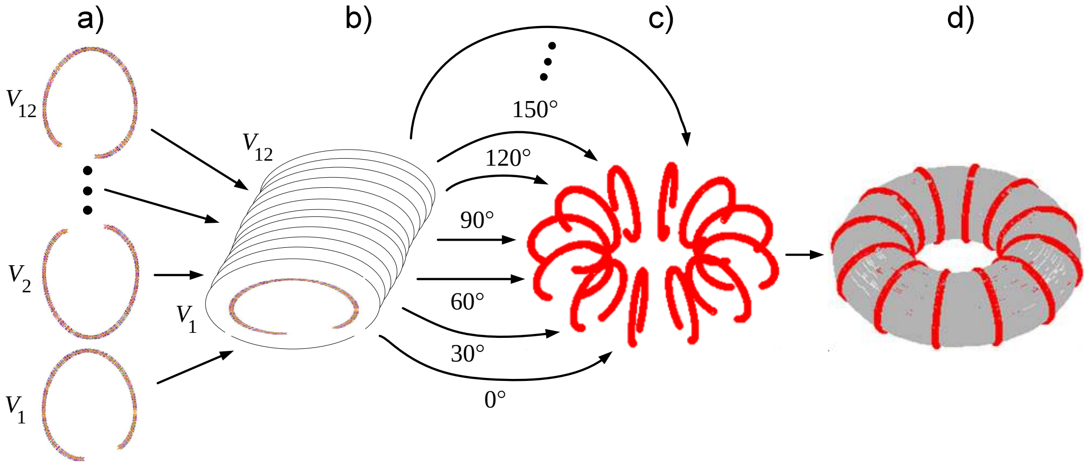


Figure 6.9. Generation of the style and view-independent manifold for the 'point' action: a) style-independent and view-dependent 2D manifolds; b) the set of aligned 2D manifolds; c) the assembled style and view-independent 3D action manifold and d) approximation of the manifold continuity (see subsequent section 6.4.1.2.2).

The process of producing the unified manifold comprises two steps (Figure 6.9). First, the view-dependent representations are combined (Figure 6.9b): the embedded spaces X^v are aligned with respect to a good representative $X^{v}{}'$ using Procrustes analysis [Wang and Mahadevan, 2008]. Since this is a rigid transformation of the spaces, the internal structure of each manifold is not changed. Secondly, each embedded representation X^v is aligned into a three-dimensional structure according to the view angle parameter $\mu^{v} \in < 0, 2\pi >$. The outcome of this procedure reveals a torus-like structure which encapsulates both style and view (Figure 6.9c). We called this structure a view and style-independent action manifold. This result is in line with previous work [Elgammal and Lee, 2009], where the usage of a torus is justified as an ideal representation for modelling both the viewpoint and the body configuration of different actions. However, while, in that work the topological correspondence between data points Y and an ideal torus is artificially enforced, our torus-like representation is data-driven and reflects the temporal structure of the view-dependent data. Therefore, in our approach all types of motions, i.e. periodic, quasi-periodic and non-periodic, see (Figure 6.10c and Figure 6.11c), are handled using the same framework.

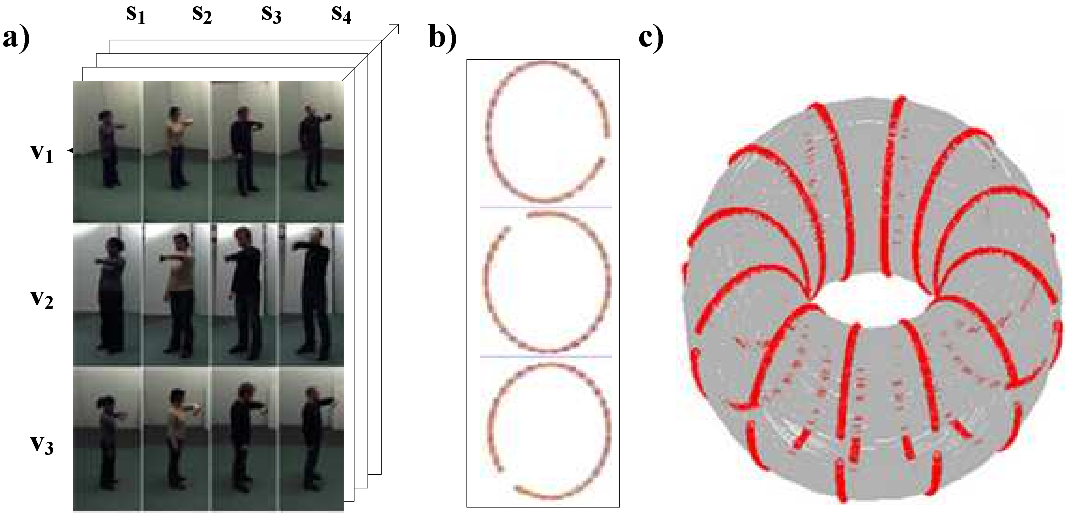


Figure 6.10. Training results for quasi periodic action ”check watch”: a) training videos; b) style-independent low dimensional representation for each view; c) style and view-independent manifold.

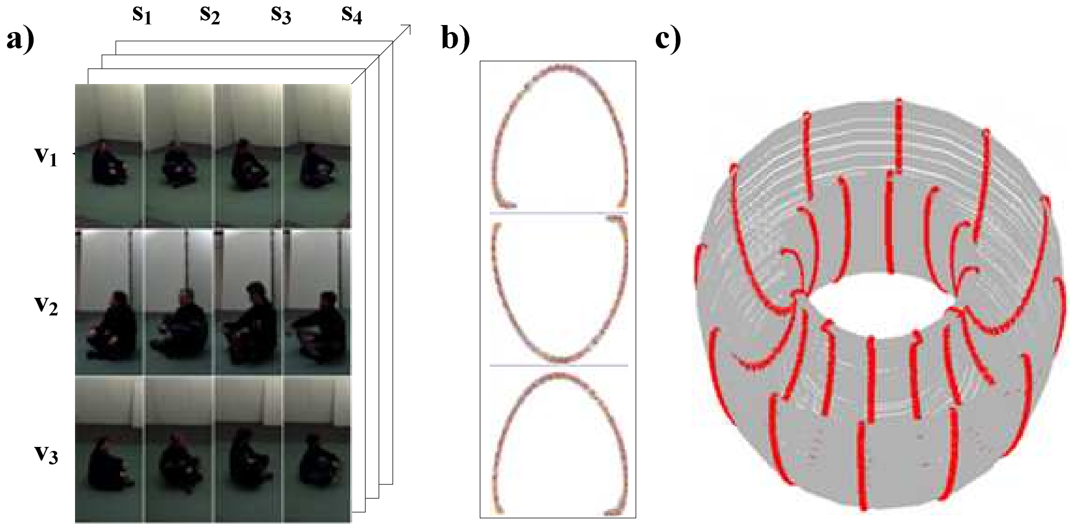


Figure 6.11. Training results for non periodic action ”sit down”: a) training videos; b) style-independent low dimensional representation for each view; c) style and view-independent manifold.

6.4.1.2.2. Manifold Mapping Function

In the previous section, we described how view descriptors could be combined to form anunique view-independent action manifold (Figure 6.9c). Since TLE is a spectral dimensionality reduction method, there is no generative mapping function between observed and embedded spaces. As a consequence, at this stage, the model is only defined on the training data (Figure 6.9c). In order to perform an accurate action classification, the descriptor has to be able to generalise to unseen examples by taking into account not only stylistic variations, but also view changes to avoid over-fitting.

This is achieved by learning a decomposable generative model [Lee and Elgammal, 2006a], which approximates the continuity of descriptor space in the form of a powerful projection function between the low dimensional descriptor space and high dimensional observed space (Figure 6.9d). This model aims at separating the intrinsic action configuration from other factors such as motion style and view. The considered generative model is a generalisation over the model described in section 4.5.7.3 where only style factor has been decomposed. Following [Lee and Elgammal, 2006a] approach, the generative mapping function is modelled using three factors:

- Content B : a representation of the intrinsic body configuration which characterises motion as a function of time. It is invariant to either person or view.
- Style S : a time-invariant person parameter which describes the person appearance, shape and motion style.
- View point V : a time-invariant view parameter which characterises the view point from which the performed action is captured.

In our framework, content evolves along a continuous manifold while style and view are represented by the discrete classes present in the training data. For the last two factors, intermediate states can be interpolated. As a result, we are able to approximate view and style continuity. In addition, we assume that both style and view factors are time-invariant, i.e. both parameters remain constant during any instance of an action. The procedure of fitting the decomposable generative model to the data consists of two steps. First, a set of style and view-dependent functions is trained. Then, all functions are combined into a single style and view-independent projection function.

Since mapping between the embedded manifold and the original space is highly nonlinear, generalised RBFN (see section 2.2.2.4.4) is applied to provide the nonlinear view-dependent mapping. It is expressed by $\mathbf{N}_s$ style-dependent mapping functions using equation (2.62):

$$Y^{sv} = \psi\left(X^{sv}\right)A^{sv} \tag{6.7}$$

where A^{sv} is a $(\mathbf{Z}+\mathbf{d}+1)\times\mathbf{D}$ matrix of mapping coefficients, which encodes style variability in the specific view. The kernel matrix $\psi(\bullet)$ is defined according to (2.64) by:

$$\psi(X^{sv}) = \{[\varphi(\|X^{sv}-c_1\|), \varphi(\|X^{sv}-c_2\|), ..., \varphi(\|X^{sv}-c_Z\|), 1, X^{sv}]\} \qquad (6.8)$$

where $C=\{c_j \mid j=1..\mathbf{Z}\}$ is a set of distinctive representative points in each embedded space and $\varphi(\bullet)$ is a radial basis function (see section 2.2.2.4.4). A^{sv} is calculated by applying the Moore-Penrose pseudo-inverse on matrix $\psi(X^{sv})$ and solving a linear system of equations: $A^{sv}=\psi(X^{sv})^{+}Y^{sv}$, like in section 2.2.2.4.4. In contrast to [Lee and Elgammal, 2006a], our unified manifold representation C is data-driven and independent of the style and speed factors due to the usage of TLE in the generation of view-dependent low dimensional representations. It is obtained by calculating a mean style and view manifold, which is then transformed by a non-rigid point registration procedure [Myronenko et al., 2007] to better fit the data.

The final style and view-independent decomposable generative model is obtained by multi-linear tensor analysis [Vasilescu and Terzopoulos, 2003] in the space of nonlinear mapping coefficients [Lee and Elgammal, 2006a]. Each coefficient matrix A^{sv} is represented as the coefficient vector a^{sv} of dimensionality $\mathbf{N_a}=\mathbf{D}*(\mathbf{Z}+\mathbf{d}+1)$ by column wise stacking (columns of the matrix are concatenated to form a vector). Afterwards, all coefficient vectors a^{sv} are arranged in an order three coefficient tensor A whose dimensionality is $\mathbf{N_s}\times\mathbf{N_v}\times\mathbf{N_a}$. The view and style orthogonal factors are decomposed from the assembled coefficient tensor A using higher order Singular Value Decomposition [Lathauwer et al., 2000]:

$$\mathrm{A} = B\times_1 S\times_2 V\times_3 F = G\times_1 S\times_2 V \qquad (6.9)$$

where S $(\mathbf{N_s}\times\mathbf{N_s})$ is the mode-1 basis of A, which represents the orthogonal basis for the style space. Similarly, V $(\mathbf{N_v}\times\mathbf{N_v})$ is the mode-2 basis matrix which spans the space of viewpoint parameters and F ($\mathbf{N_a} \times \mathbf{N_a}$) represents the mode-3 basis for the mapping coefficient space. B is a core tensor ($\mathbf{N_s} \times \mathbf{N_v} \times \mathbf{N_a}$) which governs the interactions between orthogonal factors represented in the mode basis matrices. Coefficient eigenmodes G is a new core tensor formed by $G = B \times_3 F$ whose dimensionality is $\mathbf{N_s} \times \mathbf{N_v} \times \mathbf{N_a}$. Mode-i $\times_i$ is a tensor product as defined in [Lathauwer et al., 2000]. To avoid over-fitting, the dimensionality of each orthogonal spaces is reduced to retain a subspace representation by preserving 99% of the original information. The reduced dimensionality for tensors B, S, V, F, G are $\mathbf{n_s} \times \mathbf{n_v} \times \mathbf{n_a}$, $\mathbf{N_s} \times \mathbf{n_s}$, $\mathbf{N_v} \times \mathbf{n_v}$, $\mathbf{N_a} \times \mathbf{n_a}$, $\mathbf{n_s} \times \mathbf{n_v} \times \mathbf{N_a}$ respectively, where $\mathbf{n_s}$, $\mathbf{n_v}$, $\mathbf{n_a}$ denote the number of basis maintained for each factor.

As the result, style-independent and view-independent projection function, which generalise the space of the action descriptor (Figure 6.9d), is expressed by equation:

$$y^{sv} = \psi(x) * unstack(G \times_1 s \times_2 v) \tag{6.10}$$

where image observation y^{sv} is generated from the body configuration represented by an embedding coordinate x using the estimated parameters of style s and view v given the learned core tensor G.

6.4.1.3. Action Classification Process

Action classification is performed by projecting an unknown motion sequence into each action descriptor using the generative decomposable model presented in the previous section 6.4.1.2.2. Then, the DTW distance (see appendix A.1) is calculated to measure similarity between the action projection and action model.

Given a new instance of action $\widetilde{Y}$, its length is first normalised as described in section 6.4.1.1.1. Then the embedded coordinates $\widetilde{X}$ of the new action are obtained by least square solution of the following nonlinear system:

$$\arg\min_{A,X} \left\| \widetilde{Y} - \psi(\widetilde{X})\widetilde{A} \right\| \tag{6.11}$$

Its minimum solution is found by determining and optimising coefficient matrix $\widetilde{A}$ given a learned model and then projecting data by solving a linear system of equations using the Moore-Penrose pseudo-inverse:

$$\psi(\widetilde{X}) = \widetilde{Y}\widetilde{A}^{+} \tag{6.12}$$

Coordinates of $\widetilde{X}$ are provided by the last **d** rows of the matrix $\psi(\widetilde{X})$. In order to determine the optimal coefficient matrix $\widetilde{A}$, we adopt an iterative procedure [Lee and Elgammal, 2006a]. First, we calculate a data driven mean view manifold C over all aligned mean styles manifolds C^{v} to obtain a homeomorphic manifold [Lee and Elgammal, 2006a]. Then, the coefficient matrix is initialised by solving the following equation:

$$\widetilde{A} = \psi(C)^{+}\widetilde{Y} \tag{6.13}$$

Let's $\tilde{a}$ denote a vector obtained by column wise stacking of matrix $\widetilde{A}$. Then given a mapping model, as described in the previous section 6.4.1.2.2, and any style vector, $\tilde{s}$, and any view vector $\tilde{v}$, we can define a coefficient vector $\tilde{a}$ by the tensor product:

$$\tilde{a} = G \times_1 \tilde{s} \times_2 \tilde{v} \tag{6.14}$$

Mapping coefficients $\tilde{a}^{sv}$ are optimised to reflect style and view of a new instance action $\widetilde{Y}^{sv}$ by minimising the following error:

$$\arg\min_{s,v} \left\| \tilde{a} - G \times_1 \tilde{s} \times_2 \tilde{v} \right\| \tag{6.15}$$

where G is derived from learning (equation (6.9)). Since tensor G represents the intrinsic body configuration 'content' of the considered action and manages interactions between all factors, an accurate solution for style and view can only be reached for the same type of action.

If the style vector, $\tilde{s}$ is known we can obtain a closed form solution for $\tilde{v}$ and vice versa. This leads to an iterative procedure for estimating $\tilde{s}$ and $\tilde{v}$

simultaneously until equation (6.15) converges [Lee and Elgammal, 2006a]. In practice, we follow Lee's approach where $\tilde{s}$ is initialised with a mean style estimate. Since the view classes are discrete, we identify the closest view class and use it to estimate $\tilde{s}$. Finally, vector $\tilde{a}$ is unstacked to create matrix $\tilde{A}$; then the action $\tilde{Y}$ is embedded into the low dimensional space using equation (6.12).

6.4.2. Probabilistic Action Model

The probabilistic formulation of action descriptors is achieved by feeding the extended version of ST-GPLVM (chapter 5) with the obtained topological structure from section 6.4.1.2.1, which encapsulates style, speed and view variability.

6.4.2.1. View-Independent Manifold

The learning pipeline of probabilistic action manifold is derived from the standard pipeline (Figure 5.1) and summarised in Figure 6.12. The latent space and parameters of ST-GPLVM model are optimised jointly under a new combined prior $p(X \mid L)$ to discover an underlying probabilistic model of action. This prior is derived by taking into account constraints associated with each view ($v = 1...N_v$) and replacing the standard prior (5.1) in the objective function (5.4) with:

$$p(X \mid L) = \prod_{v=1}^{\mathrm{N_v}} \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{tr(X_v^T L_v X_v)}{2\sigma^2}) \tag{6.16}$$

where L is a block diagonal matrix formed by all L_v:

$$L = \begin{bmatrix} L_1 & 0 & 0 & 0 \\ 0 & L_2 & 0 & 0 \\ 0 & 0 & ... & ... \\ 0 & 0 & ... & L_v \end{bmatrix} \tag{6.17}$$

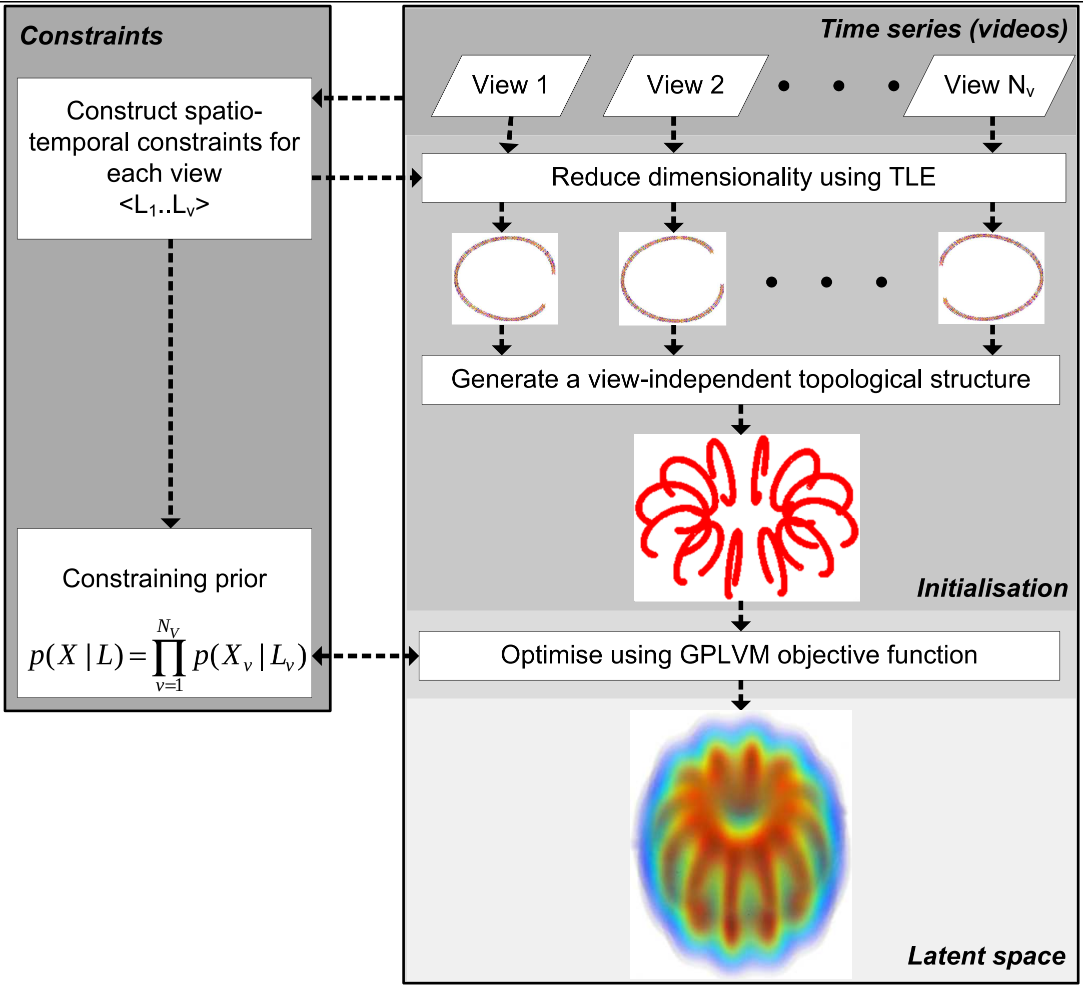

Constraints
Construct spatio-temporal constraints for each view <L1..Lv>
Constraining prior
$p(X \mid L) = \prod_{v=1}^{N_V} p(X_v \mid L_v)$
Time series (videos)
View 1
View 2
View Nv
Reduce dimensionality using TLE
Generate a view-independent topological structure
Initialisation
Optimise using GPLVM objective function
Latent space

Figure 6.12. Pipeline for generation of probabilistic view and style invariant action descriptor.

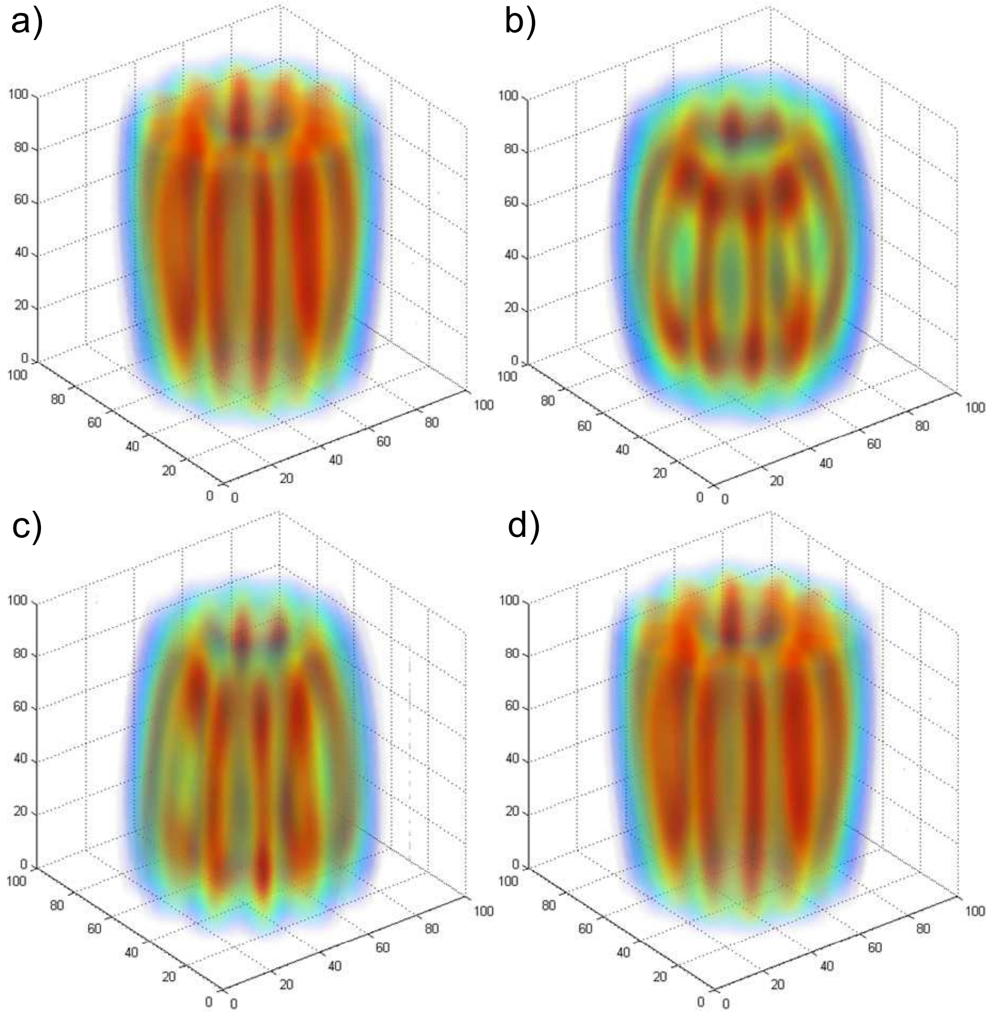


Figure 6.13. Probabilistic view and style invariant action descriptors obtained using ST-GPLVM for a) sit down, b) cross arms, c) turn around and d) kick.

6.4.2.2. Action Classification Process

The probabilistic descriptor handles naturally uncertainties inherent to actions performed by different people with different styles and in different views, therefore it is applied directly for action recognition using maximum likelihood estimation (equations (5.8)).

6.4.3. Summary

The proposed deterministic action manifold and its probabilistic extension possess all desired properties of a robust and descriptive action descriptor (see introduction

6.1). First, our action manifold is a unique and compact high level semantic description of an action, which encapsulates anthropometric and view variability, as well as normalises implicitly execution speed. Secondly, it can handle any type of motion, i.e. periodic, quasi-periodic and non-periodic. Last but not least, thank to its generative abilities, the action model is capable to handle effectively previously unobserved subjects performing the action regardless of view. As we have seen in previous sections, the core of each descriptor learning procedure is founded on our earlier contributions, i.e. TLE (chapter 4) and ST-GPLVM (chapter 5).

In order to evaluate the performance of our descriptors, an action recognition framework is designed which consists of two parts: offline training and online testing. During training, one descriptor for each action class is automatically generated. Then in testing, an instance of new action, which is performed by an unfamiliar individual in an unknown view is projected into each action manifold using descriptor based projection schema. Afterwards, a label is assigned to the new action according to the classification using either the nearest neighbour procedure for the deterministic model (section 6.4.1.3) or maximum likelihood estimation for the probabilistic formulation (section 6.4.2.2).

As we will demonstrate in the evaluation section 6.5, our descriptors are a very attractive alternative to the current state of the art methods and achieve very competitive results in the challenging task of view independent action recognition.

6.5. Evaluation

6.5.1. Experimental Setup

To obtain a dense set of action videos regarding viewpoints for the training of our action manifolds, we follow [Richard and Kyle, 2009] approach where the animated visual hulls are projected onto 12 evenly spaced virtual cameras located around the vertical axis of the subject. In line with other experiments made on this dataset [Liu and Shah, 2008, Liu et al., 2008, Yan et al., 2008, Reddy et al., 2010, Kaâniche and Brémond, 2010], the top view is discarded from the evaluation.

Two recognition tasks are evaluated using by either a single view or multiple views. In multiple views recognition, a simple majority voting rule is applied. Note that testing is performed with views which are not included in the training data. Moreover, these views differ significantly from those used for training, e.g. there is up to 45 degrees of view elevation. Following the original paper introducing the dataset [Weinland et al., 2007] as well as subsequent research [Yan et al., 2008, Tran and Sorokin, 2008, Weinland et al., 2010b, Kaâniche and Brémond, 2010], our recognition rates are computed by the leave-one-source-out method, i.e. at each run, one subject is selected for testing, whereas all remaining actors are used for descriptors learning (see section 2.3.3.4.3). A final error is estimated by the average error rate over all experiments.

In the case of the learning probabilistic descriptor, the global scaling of the constraining prior and the number of inducing variables in FITC (see section 2.2.2.3.2.2.2) are set to 10^4 and 25% of the data in each view respectively. Values of all the other parameters of the models are estimated automatically using maximum likelihood optimisation.

6.5.2. Results

Table 6.1 reports the current state of the art results and ours on this dataset where the top view has been discarded. Unfortunately, not only different approaches do not follow exactly the same evaluation protocol, but also the experimental settings differ in terms of considered number of actions and subjects. As a consequence, it is very difficult to draw any definitive conclusion based purely on those results.

Table 6.1. Average recognition accuracy over all cameras (top view excluded) using either single or multiple views for testing.

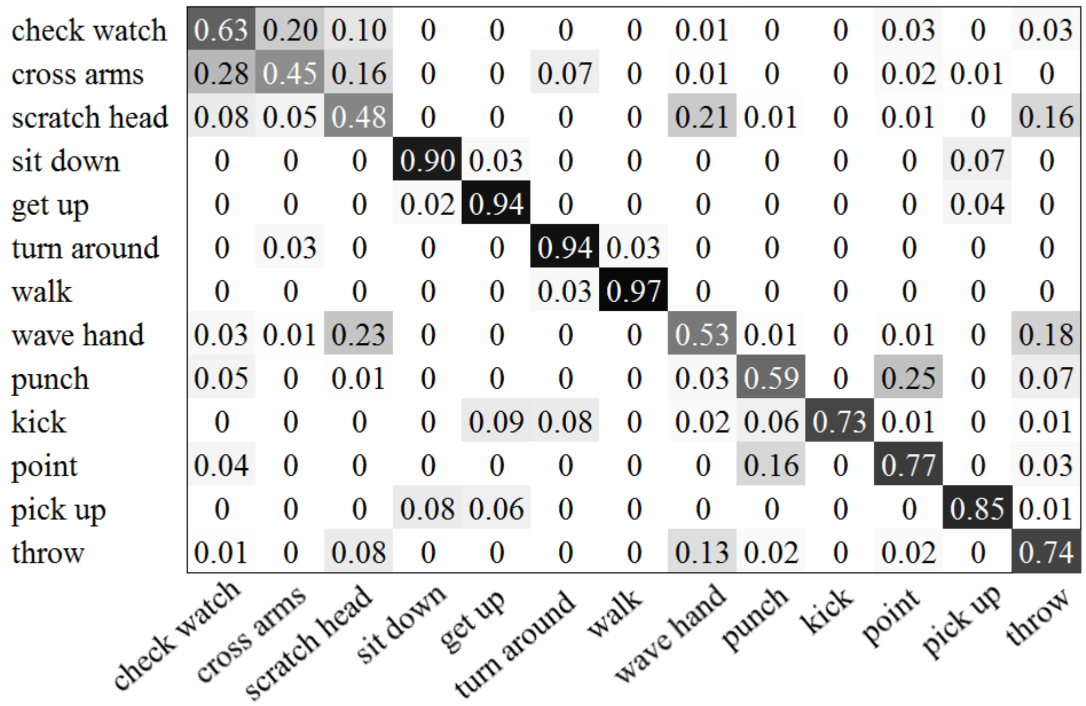
%	Subjects|Actions		Average accuracy	
			Single view	All views
Probabilistic Action Manifold	12	13	76.2	85.6
Deterministic Action Manifold	12	13	73.2	83.3
Lv [Lv and Nevatia, 2007]	10	14	82.9	-
Tran [Tran and Sorokin, 2008]	12	13	80.2	-
Liu [Liu and Shah, 2008]	12	13	73.7	82.8
Kaanische [Kaâniche and Brémond, 2010]	12	13	71.7	90.6
Liu [Liu et al., 2008]	12	13	71.7	78.5
Reddy [Reddy et al., 2010]	12	13	66.5	72.6
Probabilistic Action Manifold	12	11	78.3	84.7
Deterministic Action Manifold	12	11	74.7	83.1
Weinland [Weinland et al., 2010b]	10	11	86.9	-
Junejo [Junejo et al., 2008]	10	11	73.7	-
Yan [Yan et al., 2008]	12	11	64.0	78.0
Weinland [Weinland et al., 2007]	10	11	63.9	81.3

First, the probabilistic formulation of our action descriptor obtains better performance than the deterministic variant. This is expected, since the probabilistic action model provides directly a continuous underlying distribution of the action space, which is used effectively to generalise space to unseen instances of actions regardless of view. In contrast, the continuity of deterministic framework is only discretely approximated using the generative decomposable model (section

6.4.1.2.2). Moreover, because of the unfavourable ratio between the number of available training samples and the dimensionality of the feature space, the learning of the decomposable RBFN model is a challenging process. In particular, a forward mapping from high to low dimensional space cannot be learned directly, therefore it is obtained by an analytical inversion of an inverse mapping from low to high dimensional space, which introduces another level of inaccuracy in the model. On the other hand, the probabilistic generative mapping is more robust against over-fitting even in the case of data sample shortage in relation to the dimensionality of feature space [Lawrence, 2004, Lawrence, 2005].

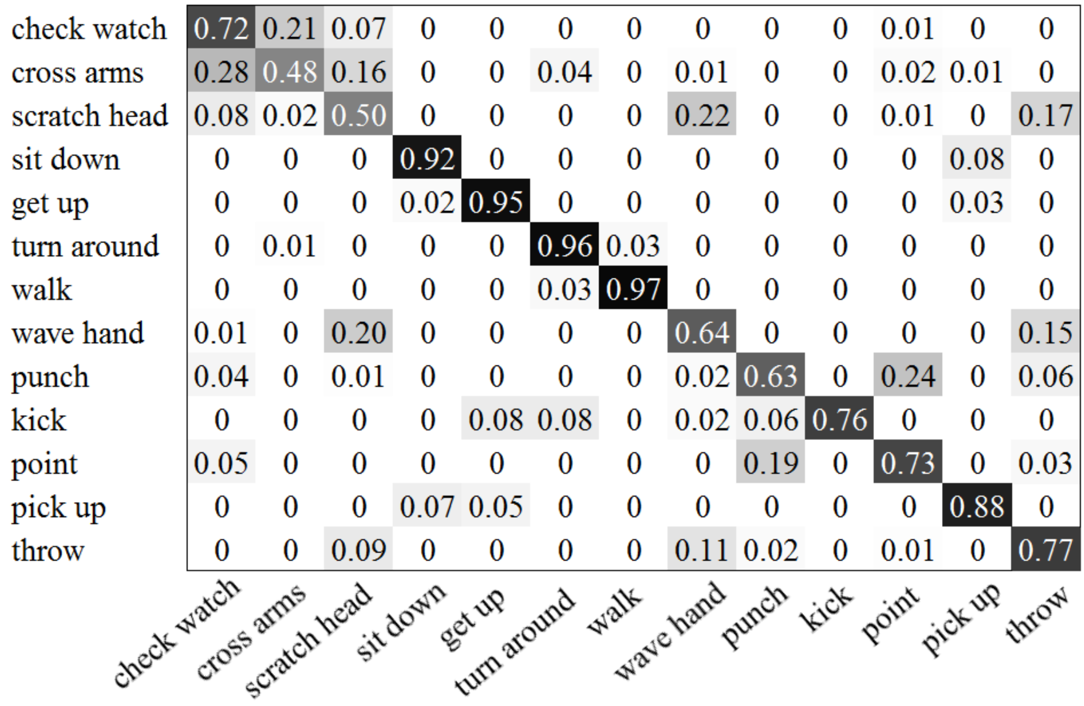
The comparison with the current state of the art approaches reveals that our probabilistic descriptor displays very good performances either when all actions completed by all subjects are considered, i.e. 13, or only 11 actions, when the ’point’ and ‘throw’ actions are discarded. Although [Tran and Sorokin, 2008] and [Weinland et al., 2010b] seem to obtain better results, both frameworks are actually trained and tested using the same views, whereas in our validation a testing view is completely unknown and thus different from the training views. As consequence, it is unclear how results of these two competitors [Tran and Sorokin, 2008, Weinland et al., 2010b] would extrapolate to the more complex scenario of action recognition in an unfamiliar view. In the light of those results, our descriptor exhibits an exceptional robustness not only to subject style variability but also to view variations in terms of azimuth and elevation angles. Note that results of [Lv and Nevatia, 2007] are reported only for a single sequence (out of three) per actor. This sequence was selected to achieve the best results, thus making a direct comparison impossible, since all repetitions are considered in our validation. Furthermore, some approaches [Weinland et al., 2007, Lv and Nevatia, 2007, Junejo et al., 2008, Weinland et al., 2010b] and especially two of our main competitors [Lv and Nevatia, 2007, Weinland et al., 2010b] use a smaller set of available subjects which may further favour their approaches. Finally, results cannot be compared with [Richard and Kyle, 2009], because, instead of evaluating their method with original video data, they did it by using projections of the visual hulls.

Figure 6.14 and Figure 6.15 present the confusion matrices of recognition for the ‘single view’ experiment, whereas Figure 6.16 and Figure 6.17 depict the confusion matrices for the ’all-view’ experiment using deterministic and probabilistic descriptor respectively. They reveal that our framework performed better when dealing with motions involving the whole body, i.e. ”walk”, ”sit down”, ”get up”, ”turn around” and ”pick up”. Since temporal information is essential when dealing with highly dynamic motions and TLE aims at preserving temporal structure in each view, action manifolds of those activities are more representative. Due to more powerful generative abilities, the probabilistic descriptor outperforms the deterministic variant especially by reducing confusion between hand related motions.



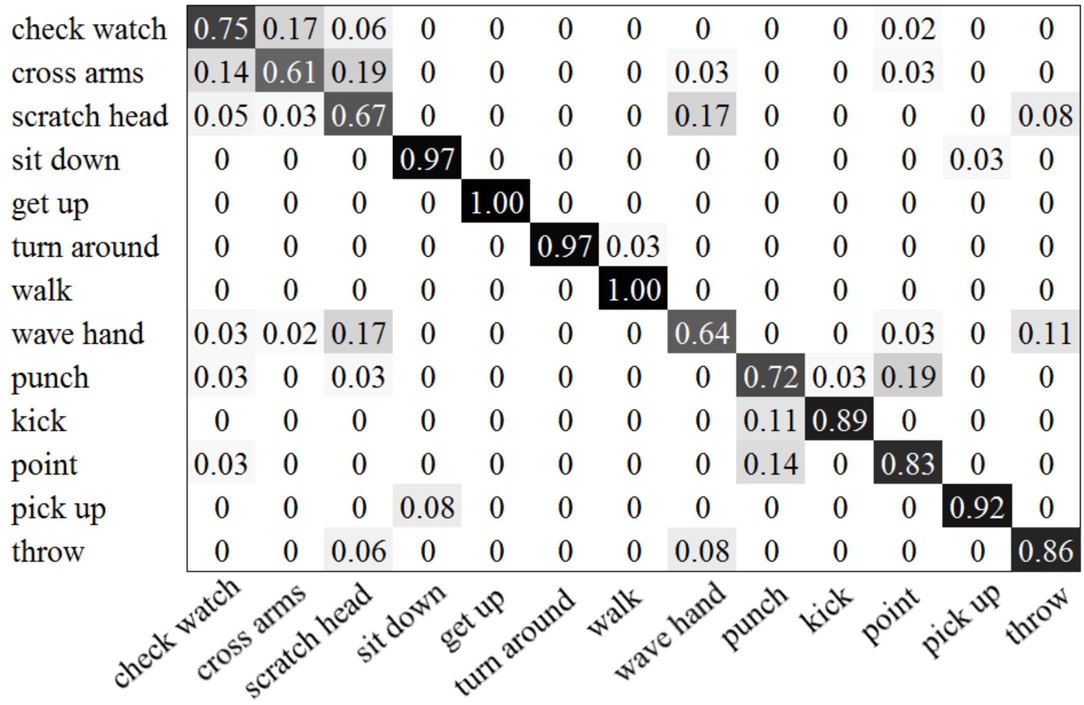
	check watch	cross arms	scratch head	sit down	get up	turn around	walk	wave hand	punch	kick	point	pick up	throw
check watch	0.63	0.20	0.10	0	0	0	0	0.01	0	0	0.03	0	0.03
cross arms	0.28	0.45	0.16	0	0	0.07	0	0.01	0	0	0.02	0.01	0
scratch head	0.08	0.05	0.48	0	0	0	0	0.21	0.01	0	0.01	0	0.16
sit down	0	0	0	0.90	0.03	0	0	0	0	0	0	0.07	0
get up	0	0	0	0.02	0.94	0	0	0	0	0	0	0.04	0
turn around	0	0.03	0	0	0	0.94	0.03	0	0	0	0	0	0
walk	0	0	0	0	0	0.03	0.97	0	0	0	0	0	0
wave hand	0.03	0.01	0.23	0	0	0	0	0.53	0.01	0	0.01	0	0.18
punch	0.05	0	0.01	0	0	0	0	0.03	0.59	0	0.25	0	0.07
kick	0	0	0	0	0.09	0.08	0	0.02	0.06	0.73	0.01	0	0.01
point	0.04	0	0	0	0	0	0	0	0.16	0	0.77	0	0.03
pick up	0	0	0	0.08	0.06	0	0	0	0	0	0	0.85	0.01
throw	0.01	0	0.08	0	0	0	0	0.13	0.02	0	0.02	0	0.74

Figure 6.14. Class-confusion matrix using average recognition over single views for deterministic action manifolds. The average performance is 73.2%.



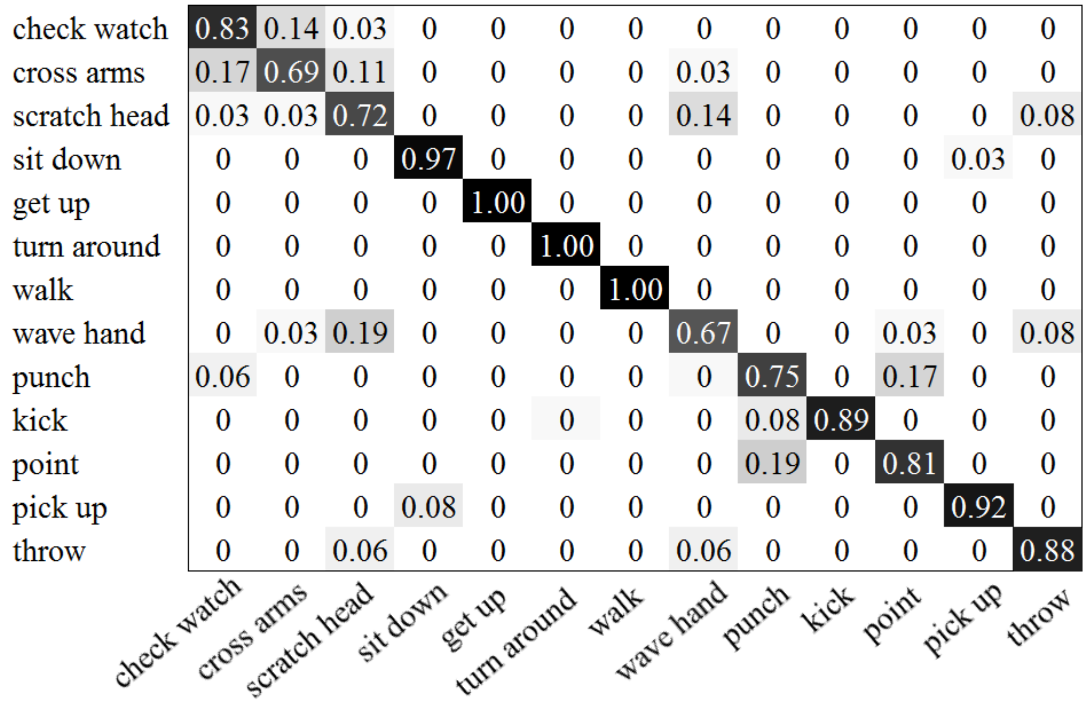
	check watch	cross arms	scratch head	sit down	get up	turn around	walk	wave hand	punch	kick	point	pick up	throw
check watch	0.72	0.21	0.07	0	0	0	0	0	0	0	0.01	0	0
cross arms	0.28	0.48	0.16	0	0	0.04	0	0.01	0	0	0.02	0.01	0
scratch head	0.08	0.02	0.50	0	0	0	0	0.22	0	0	0.01	0	0.17
sit down	0	0	0	0.92	0	0	0	0	0	0	0	0.08	0
get up	0	0	0	0.02	0.95	0	0	0	0	0	0	0.03	0
turn around	0	0.01	0	0	0	0.96	0.03	0	0	0	0	0	0
walk	0	0	0	0	0	0.03	0.97	0	0	0	0	0	0
wave hand	0.01	0	0.20	0	0	0	0	0.64	0	0	0	0	0.15
punch	0.04	0	0.01	0	0	0	0	0.02	0.63	0	0.24	0	0.06
kick	0	0	0	0	0.08	0.08	0	0.02	0.06	0.76	0	0	0
point	0.05	0	0	0	0	0	0	0	0.19	0	0.73	0	0.03
pick up	0	0	0	0.07	0.05	0	0	0	0	0	0	0.88	0
throw	0	0	0.09	0	0	0	0	0.11	0.02	0	0.01	0	0.77

Figure 6.15. Class-confusion matrix using average recognition over single views for probabilistic action manifolds. The average performance is 76.2%.



	check watch	cross arms	scratch head	sit down	get up	turn around	walk	wave hand	punch	kick	point	pick up	throw
check watch	0.75	0.17	0.06	0	0	0	0	0	0	0	0.02	0	0
cross arms	0.14	0.61	0.19	0	0	0	0	0.03	0	0	0.03	0	0
scratch head	0.05	0.03	0.67	0	0	0	0	0.17	0	0	0	0	0.08
sit down	0	0	0	0.97	0	0	0	0	0	0	0	0.03	0
get up	0	0	0	0	1.00	0	0	0	0	0	0	0	0
turn around	0	0	0	0	0	0.97	0.03	0	0	0	0	0	0
walk	0	0	0	0	0	0	1.00	0	0	0	0	0	0
wave hand	0.03	0.02	0.17	0	0	0	0	0.64	0	0	0.03	0	0.11
punch	0.03	0	0.03	0	0	0	0	0	0.72	0.03	0.19	0	0
kick	0	0	0	0	0	0	0	0	0.11	0.89	0	0	0
point	0.03	0	0	0	0	0	0	0	0.14	0	0.83	0	0
pick up	0	0	0	0.08	0	0	0	0	0	0	0	0.92	0
throw	0	0	0.06	0	0	0	0	0.08	0	0	0	0	0.86

Figure 6.16. Class-confusion matrix using multiple views for deterministic action manifolds. The average performance is 83.3%.



	check watch	cross arms	scratch head	sit down	get up	turn around	walk	wave hand	punch	kick	point	pick up	throw
check watch	0.83	0.14	0.03	0	0	0	0	0	0	0	0	0	0
cross arms	0.17	0.69	0.11	0	0	0	0	0.03	0	0	0	0	0
scratch head	0.03	0.03	0.72	0	0	0	0	0.14	0	0	0	0	0.08
sit down	0	0	0	0.97	0	0	0	0	0	0	0	0.03	0
get up	0	0	0	0	1.00	0	0	0	0	0	0	0	0
turn around	0	0	0	0	0	1.00	0	0	0	0	0	0	0
walk	0	0	0	0	0	0	1.00	0	0	0	0	0	0
wave hand	0	0.03	0.19	0	0	0	0	0.67	0	0	0.03	0	0.08
punch	0.06	0	0	0	0	0	0	0	0.75	0	0.17	0	0
kick	0	0	0	0	0	0	0	0	0.08	0.89	0	0	0
point	0	0	0	0	0	0	0	0	0.19	0	0.81	0	0
pick up	0	0	0	0.08	0	0	0	0	0	0	0	0.92	0
throw	0	0	0.06	0	0	0	0	0.06	0	0	0	0	0.88

Figure 6.17. Class-confusion matrix using multiple views for probabilistic action manifolds. The average performance is 85.6%.

The best recognition rates for both descriptors are obtained for camera 2 and 4 respectively (Figure 6.18 and Figure 6.19). This was expected, since both views are the most similar among those used for training. Moreover, when dealing with either different, i.e. camera 1, or even significantly different views, i.e. camera 3, our framework still achieves reasonable recognition rates (Figure 6.18 and Figure 6.19), which confirms the outstanding generalisation properties of the descriptors to view alteration.

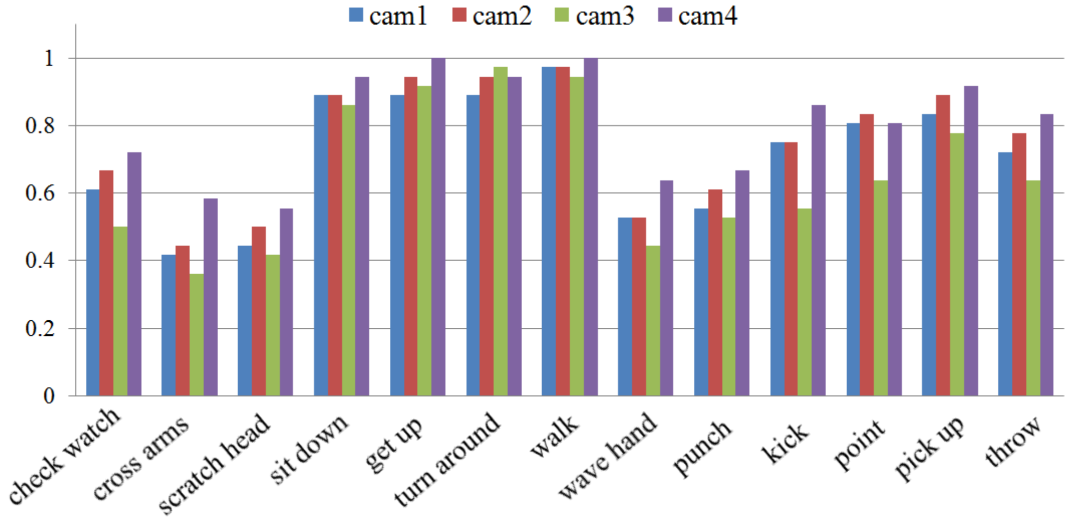


Figure 6.18. Action recognition rates using single views for deterministic action manifolds. Average values are 71.6%, 74.9%, 65.8% and 80.6% for cameras 1 to 4, respectively.

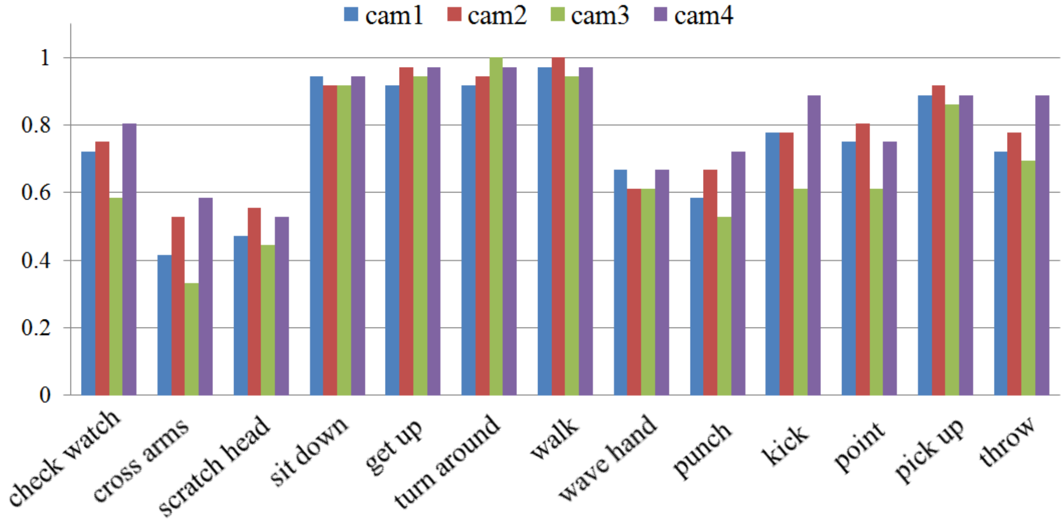


Figure 6.19. Action recognition rates using single views for probabilistic action manifolds. Average values are 75.0%, 78.6%, 69.9% and 81.4% for cameras 1 to 4, respectively.

Table 6.2 provides insight into the approximated processing times of generating the proposed action descriptors (training) and recognising of a new action (testing) based on an unoptimised Matlab code and single 3Ghz CPU. First, the generation of deterministic descriptors is significantly more efficient than probabilistic ones because the deterministic learning process is solely analytical and non iterative,. Since TLE is very efficient (Table 4.3), especially using the repetition neighbourhood selection procedure tailored to action videos (sections 6.4.1.1.3 and 6.4.1.1.4), most of the training time is spent on the generation of shape descriptors (section 6.4.1.1.1) and the learning of generative decomposable model (section 6.4.1.2.2). In contrast, the recognition phase in both descriptors involves computationally expensive optimisation procedure (sections 6.4.1.3 and 6.4.2.2), thus processing times are relatively large and proportional to the action lengths, especially for the probabilistic action model where a more complex and unconstrained problem is optimised. Finally, note that both frameworks were implemented in Matlab and a code optimisation was not our main concern. As a result, both frameworks should be considered as prototypes of solution, which validate successfully the proposed methodology, rather than the productive applications. A significant improvement of efficiency may be achieved by using a more advanced programming language like C++. Moreover, since a training of different models as well as recognition of different action primitives are independent processes, they can be easily parallelised using a cluster environment thus further reducing processing times.

Table 6.2. Average processing time of generating models and recognising actions using an unoptimised Matlab code and single 3Ghz CPU.

Average time [hours]	Deterministic Action Manifold	Probabilistic Action Manifold
Training per action model	~5	~62
Testing per action primitive	~7	~11

6.6. Summary

In this chapter, our contributions from chapters 4 and 5, were applied in a realistic and challenging computer vision task, i.e. view-independent action recognition from a monocular video. As a consequence, a novel human action recognition framework was proposed, which is based on intuitive and compact action descriptors, which reside in a low dimensional space. We introduced two action models, i.e. deterministic and probabilistic which represent any action independently from camera views, execution rates and individuals' styles. The learning procedures involve the TLE, and ST-GPLVM, for the probabilistic model, in order to extract the descriptive action pattern, while maintaining appropriate adaptability to all forms of variations within the action class.

Although the discussed methods cannot be compared purely on the reported action recognition performances, we believe that our action models are superior, especially in comparison to local feature based descriptors (section 2.3.3.1.1) [Liu and Shah, 2008, Liu et al., 2008, Junejo et al., 2008, Reddy et al., 2010, Kaâniche and Brémond, 2010, Weinland et al., 2010b]. Due to the sparsity of the data relative to the diversity of naturally plausible motions and the difficulty of acquiring larger amounts of appropriate training data, generative action models like ours seem to be more practical in real-life applications. This is because of their outstanding generalisation properties to previously unobserved styles, speeds and views as we have demonstrated in the evaluation (section 6.5).

In conclusion, our contributions were proved to be applicable in a real application and obtained very satisfactory and promising results. In addition, since our action models are general, they should benefit many other applications beyond action recognition such as visual surveillance or sport analysis.

7. Conclusions

This chapter concludes the dissertation. In section 7.1, we summarise briefly our contributions to the fields of machine learning/pattern recognition and computer vision. Then, a general discussion of our achievements is given in section 7.2. Afterwards, we highlight remaining open issues and limitations of the proposed solutions as well as a number of avenues for future research in section 7.3. Finally, closing remarks are provided in section 7.4.

7.1. Summary of Contributions

In this thesis, we explored comprehensively the field of dimensionality reduction with a special focus on computer vision applications.

First, in chapter 2, we provided an extensive review and discussion of the main directions of research in dimensionality reduction and computer vision. A detailed analysis of both fields allowed uncovering some fundamental research problems which had not yet been solved satisfactory by the research community. Thus we decided to address them in this dissertation

Our research began in chapter 3 with thorough examination of a family of powerful nonlinear spectral dimensionality reduction methods and study of their limitations, i.e. selection of free parameters and lack of generative abilities of unseen examples. We proposed a framework for the automatic configuration of spectral dimensionality reduction methods, which overcomes identified weaknesses (section 3.3). First, the mutual information measure was adopted to develop an automatic procedure for neighbourhood size selection (section 3.3.1). Then, we automated and adjusted a process of Radial Basis Function Network (RBFN) learning to design a generative mapping function between embedded and data spaces (section 3.3.2). This was achieved by taking advantage of the efficient Markov Cluster algorithm and the graph constructed during the dimensionality reduction process. The combination of these two innovative ideas allowed proposing a flexible and unified methodology for the automatic configuration of spectral dimensionality reduction techniques, which should benefit areas wherever scientists face the problem of analysing high dimensional data.

Then, since a key feature of many natural phenomena is that their course is expressed in the time domain, we examined issues related to the usage of dimensionality reduction techniques to time oriented data, i.e. multidimensional time series. Despite the huge research effort that has been dedicated to dimensionality reduction (section 2.2) the majority of work does not take into consideration appropriately the dynamic characteristics of many phenomena. To address this challenging research problem, in chapter 4, we proposed a novel spectral dimensionality reduction method, called Temporal Laplacian Eigenmaps (TLE), which exploits temporal coherence as an essential clue of the dimensionality reduction process. This was achieved by taking advantage of spatial and temporal coherency relationships between time series in order to extract the intrinsic parameterisation of the high dimensional time series space regardless of data variations. These time series constraints are expressed in the form of two complementary temporal graphs (sections 4.4.2.1 and 4.4.2.2), which are incorporated into the standard Laplacian Eigenmap framework (section 4.4.2.3) without requiring the manual tuning of parameters. Based on this original concept, the proposed method aims at preserving implicitly the local and global temporal topologies of observed spaces during dimensionality reduction instead of maintaining only geometry as it is usually the case. This allowed producing automatically meaningful and generalised low dimensional representations tailored to multidimensional time series data.

In some scenarios, in order to cover adequately the complexity and richness of measured phenomena, massive amounts of representative data are required to learn appropriate data-driven models. Since the capture of large quantity of data may be impractical, a solution may be to generalise known data samples to the entire phenomenon space to obtain a reliable model. In chapter 5, motivated by the spatio-temporal constraints of TLE, we introduced the concept of a spatio-temporal conditioned prior which is placed over a latent space and constrains the optimisation process of Gaussian Process Latent Variable Model (section 5.3.2). As a consequence, a novel generative nonlinear dimensionality reduction algorithm, which is called Spatio-Temporal Gaussian Process Latent Variable Model (ST-GPLVM), was proposed. This innovative approach is capable of approximating a compact underlying distribution of time series space in the presence of data variations. As a result, a core pattern of multivariate time series is extracted in the form of generative and continuous mapping function from a low to a high dimensional space with associated uncertainties of prediction.

Finally, in chapter 6, we investigated further the practicality of our contributions from chapters 4 and 5 in a realistic and challenging real-life computer vision task of view-independent action recognition. As a consequence, a novel human action recognition framework was developed, which is based on devised deterministic (section 6.4.1) and probabilistic (section 6.4.2) variants of temporally constrained action manifolds. These descriptors encapsulate style, view and speed variability of any type of motion in a compact and consistent low dimensional representation. The key advantage of the introduced descriptors is their generalisation abilities to previously unobserved motions regardless of view. Very satisfactory and promising results confirmed the usefulness of our contributions in a real application and suggest many potential applications beyond computer vision.

7.2. Discussion

The main emphasis of this thesis is the modelling of multidimensional time series data using an underlying low dimensional representation with applications to human motion analysis. Although chapter 3 did not directly address this issue, it was an essential step in our research. It allowed for a thorough insight into theoretical and practical aspects of spectral dimensionality reduction transformations and thus implicitly stimulated the proposal of TLE in chapter 4. In turn, chapter 5 with the introduced ST-GPLVM shows an attractive evolution of the TLE concept to the generative modelling of multidimensional time series data. Finally, the action recognition framework presented in chapter 6 was derived from the previous contributions (chapters 4 and 5) to demonstrate a deployment of proposed ideas in a real-life computer vision application of multivariate time series classification.

Note that, to some extent, TLE and ST-GPLVM are competitive methods for dimensionality reduction of time series data. The choice of the algorithm is not straightforward and depends on the application as well as the amount of available training data. On one hand, although ST-GPLVM is significantly more computationally expensive than TLE, it exhibits better generalisation properties as seen in sections 5.4.4 and 6.5.2. On the other hand, TLE has superior scalability in terms of dataset size and dataset dimensionality. Moreover, in a combination with RBFN, it is often able to produce similar performances assuming that ‘enough’ data are collected for training.

Performance of both approaches, i.e. TLE and ST-GPLVM, relies heavily on the appropriate identification of repetition neighbours in order to construct adequately spatio-temporal constraints between time series. Although we have proved that the DTW-based repetition neighbourhood selection procedure is capable to tackle effectively this issue in various applications, in some cases, it may be useful to customise it in order to take full advantage of a domain specific feature representation. As seen in sections 4.5.4 and 6.4.1.1.3, the adaptation of procedure should not be a major issue since it does not require any modification of the dimensionality reduction method core.

Other drawbacks of our approaches are inherited from their respective parents. Similarly to other spectral dimensionality reduction methods, TLE does not provide any inherent generalisation abilities to unseen data. As a consequence, in many situations, the additional post-processing step of RBFN learning is required, which increases the computational cost of model learning. In the case of ST-GPLVM, it is computationally expensive by design, because of the reliance on an iterative optimisation process.

In addition to these, an important limitation of our proposed view independent action recognition frameworks is that they assume the required video processing step can be solved satisfactorily, thus providing sufficient information for the machine learning and recognition processes. Unfortunately, this is usually only a valid assumption when dealing with data captured in a controlled environment: analysis of unconstrained videos is an open and difficult scientific challenge. Moreover, our proposed frameworks cannot deal with unknown actions; hence, our action recognition frameworks rely on training datasets which are composed of all possible actions, which may appear during the recognition phase.

Although, we made significant steps towards solutions of a few essential problems in dimensionality reduction, i.e. the automatisation of dimensionality reduction process (chapter 3) and the deterministic/probabilistic parameterisation of time series data (chapters 4 and 5), we have certainly not provided a definitive solution but rather a solid and appealing foundations for further research and improvements. Similarly, our view independent action recognition framework in chapter 6, despite several advantages over existing approaches, has some limitations. A few still open issues, suggestions of possible extensions, and the promising directions for further research are outlined in the next section 7.3.

7.3.Future Work

While the graph-based RBFN has been proved to be an efficient approach tailored to spectral dimensionality reduction methods, the selection of radial basis activation functions in a network design is still an open question. In this research, we chose the Gaussian basis function as suggested by [Poggio and Girosi, 1990]. However different basis functions, such as thin plate spline, multiquadratic, Cauchy and many others [Powell, 1987], may produce mapping functions which display different performances. Consequently, it would be interesting to examine the sensitivity of the graph-based RBFN to this choice. More fundamentally, since there is no general rule suggested in the literature about how to automate the basis function selection process, this is an area which would be worth investigating and could impact significantly on further improvement of generalisation capabilities of spectral dimensionality reduction methods.

The proposed TLE maintains the temporal continuity of time series during dimensionality reduction process and suppress stylistic variations displayed by different sources of time series by aligning them in a low dimensional space. However, style variability is actually not completely removed from the low dimensional representation but only drastically marginalised during the dimensionality reduction process in order to extract the intrinsic pattern of time series data (see justification in section 4.4.2.4). Since the maintenance of style information may be advantageous in some applications, such as tracking, an interesting idea to investigate would be to model explicitely style variability along an extra dimension. This could be done by compensating the domination of the temporal constraints over spatio-temporal ones using a balancing mechanism between constraints in the optimisation process. This would enforce equal importance in the preservation of both stylistic and temporal variations of time series. This is ongoing research and preliminary results are shown in [Martinez-del Rincon et al., 2011].

Although ST-GPLVM proved to be a powerful extension of the TLE concept and confirmed to be computationally more attractive than the standard GPLVM framework, it may still be impractical for large and high dimensional datasets. In addition, it requires empirical selection of a few parameters which introduces another level of complexity in its exploitation. One interesting possibility, which is worth of further study is a direct reformulation of TLE into the generative framework inspired by [Lu et al., 2007, Kanaujia et al., 2007]. As a consequence, TLE could be extended with a bi-directional probabilistic mapping between a latent and observed space, which reflects the underlying data distribution. Similarly to TLE, such enhancement would be parameterless and according to [Lu et al., 2007, Kanaujia et al., 2007] more efficient than GPLVM based approaches.

Finally, the creation of a robust and full pipeline for view independent action recognition in a realistic visual surveillance scenario is a very ambitious project, which is well beyond the scope of a single PhD. This thesis demonstrates promising progress towards such a goal, however a number of simplifications and shortcuts had to be employed to obtain a running prototype system. Our main intention was to validate our contributions in a real computer vision application rather than building a productive application. First, we focused on a high level semantic description of an action. In line with other research in the field (section 2.3.3.1.2), we assumed that localisation and segmentation of a moving person, as well as a temporal segmentation of action into primitives can be carried out sensibly by some low level pre-processing of video data.

In order to develop a full pipeline for action recognition from videos, the proposed framework should be extended by incorporation of some advanced techniques for video analysis, such as [Dalal and Triggs, 2005, Felzenszwalb et al., 2010, Simonnet and Velastin, 2010] for person localisation in videos, [Stauffer and Grimson, 1999, Fuentes and Velastin, 2001] for background/foreground segmentation, [Yin et al., 2008a] for ghost removal and finally [Cutler and Davis, 2000, Rui and Anandan, 2002, Weinland et al., 2006a] for the temporal segmentation of actions into primitives. A further interesting aspect to investigate is the usage of more advanced feature representation, such as optical flow [Efros et al., 2003] or a variant of space-time interest points [Laptev and Lindeberg, 2003, Laptev, 2005, Dollar et al., 2005, Dalal and Triggs, 2005]. However note that a change of feature representation will impose the design of an appropriate neighbourhood selection procedure in TLE for the determination of repetition neighbours.

Another software engineering problem is that the current prototype implementation of frameworks is computationally prohibitive for a productive application. Thus it would be desirable to redevelop the proposed methodologies using a more computationally efficient programming language like C++. In terms of scientific challenges, the frameworks could be extended to deal with complex actions by using action primitive models as a codebook in some sort of hierarchical classification schema. Alternatively, a high level fusion or voting module could be introduced which would allow for interaction recognition by combining independent classification results of each individual using action primitive models.

In addition, in order to make an objective comparison between different algorithms, but without a loss of generality, we have used publicly available IXMAS database for the evaluation. Although, it is one of the most challenging datasets in view independent action recognition available for research community, it is captured in a controlled environment, thus differs from real visual surveillance recordings. One of the essential tasks for future research should be an evaluation of the descriptors using a more realistic visual surveillance dataset in the recognition stage.

7.4. Closing Remarks

We believe that this thesis has contributed to the principles and practice of dimensionality reduction field and hopefully it is a significant step towards the applicability of dimensionality reduction to a wider range of scientific problems wherever there is a need to explore large volumes of multivariate data. The automatisation of spectral dimensionality reduction approaches simplify usage of these algorithms, and thus may help many scientists in taking advantage of a dimensionality reduction transformation to eliminate undesired properties of high dimensional data before applying domain specific processing. Similarly, wherever time is an essential characteristic of examined phenomena, we equipped scientists with two powerful methodologies for the deterministic or probabilistic representation of such multidimensional time series data using only key underlying parameters.

Our contributions proved to be especially useful in two computer vision tasks, i.e. human pose recovery and action recognition, and inspired us to propose a promising and advanced view independent action recognition framework which may open the door to the longstanding aspiration of robust and automatic interpretation of human motion.

Finally, the presented contributions are intended to motivate future research in the area of machine learning/pattern recognition with applications to computer vision problems and, hopefully, built a firmer foundation for a next generation of nonlinear dimensionality reduction methods for time series.

A. Appendices

A.1. Dynamic Time Warping

Dynamic Time Warping [Rabiner and Juang, 1993, Senin, 2008] (DTW) is an algorithm for measuring similarity between two time series (high dimensional curves) which minimises the effects of shifting and distortion in time by allowing “elastic" transformation of time series in order to detect similar shapes with different phases.

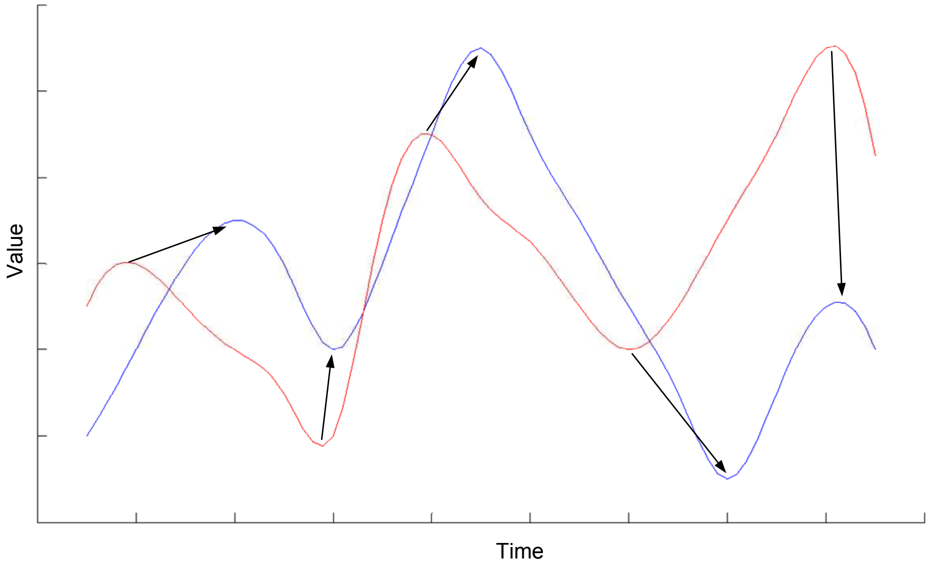


Figure 7.1. Raw time series where arrows show desirable points of alignment.

Given two time series $A = (a_1, a_2, ..., a_N)$ $(a_i \in \mathbb{R}^D)$ and $B = (b_1, b_2, ..., b_M)$ $(b_i \in \mathbb{R}^D)$, optimal matching becomes the task of aligning two sequences of points in order to generate the most representative distance measure of their overall difference (Figure 7.1). The naive approach of aligning points is a plain linear matching, where every i*th* point of the first curve matches with i*th* point of the second curve, and both curves are of equal length (Figure 7.2a). However, this procedure produces a poor similarity score. Alternatively, DTW allows for a nonlinear (elastic) alignment of times series by minimizing the warping cost function (Figure 7.2b). As a result, a more intuitive similarity measure is obtained, which allows matching similar shapes even if they are out of phase in the time axis and/or they are not of equal size.

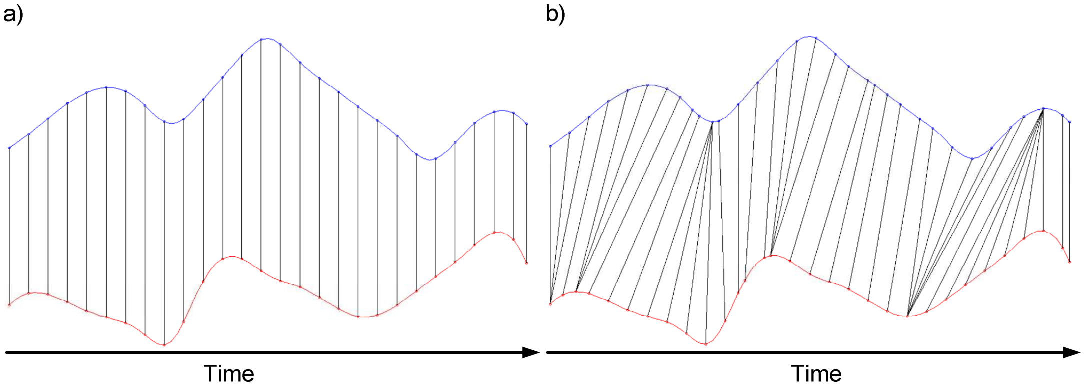


Figure 7.2. Time series alignment: a) Linear matching "one to one", b) nonlinear matching by warping time axis.

The algorithm starts by building the Euclidean cost distance matrix $E = \{e_{ij}\}_{i=1..N,\, j=1..M}$ representing all pair wise distances between A and B:

$$e_{ij} = \|a_i - b_j\|, i = 1..N, j = 1..M \tag{6.18}$$

Once the cost matrix is built, the algorithm finds the best alignment path (i.e. warping path) which satisfies the following criteria:

- Boundary condition which assigns first and last elements of A and B to each other.
- Monotonicity condition which preserves the time-ordering of points.
- Continuity condition which limits the warping path from long jumps (shifts in time) while aligning sequences.

Let's an accumulated global cost matrix is denoted by P where the first row and the first column are initialised according to the following equations:

$$P(1, j) = \sum_{k=1}^{j} e_{1k}, j = 1..M \tag{6.19}$$

$$P(i,1) = \sum_{k=1}^{i} e_{k1}, i = 1..N \tag{6.20}$$

then the cost function associated with a warping path is computed with respect to the distance matrix E expressed by:

$$P(i, j) = \min\{P(i-1, j-1), P(i-1, j), P(i, j-1)\} + e_{ij}, i = 2..N, j = 2..M \tag{6.21}$$

The final warping path in the global cost matrix P, i.e. the correspondence between elements of A and B, is illustrated in Figure 7.3. The path is found by simple backtracking from the end point $P(N,M)$ to the start point $P(1,1)$ following a greedy strategy. More information about DTW is provided by [Rabiner and Juang, 1993, Senin, 2008].

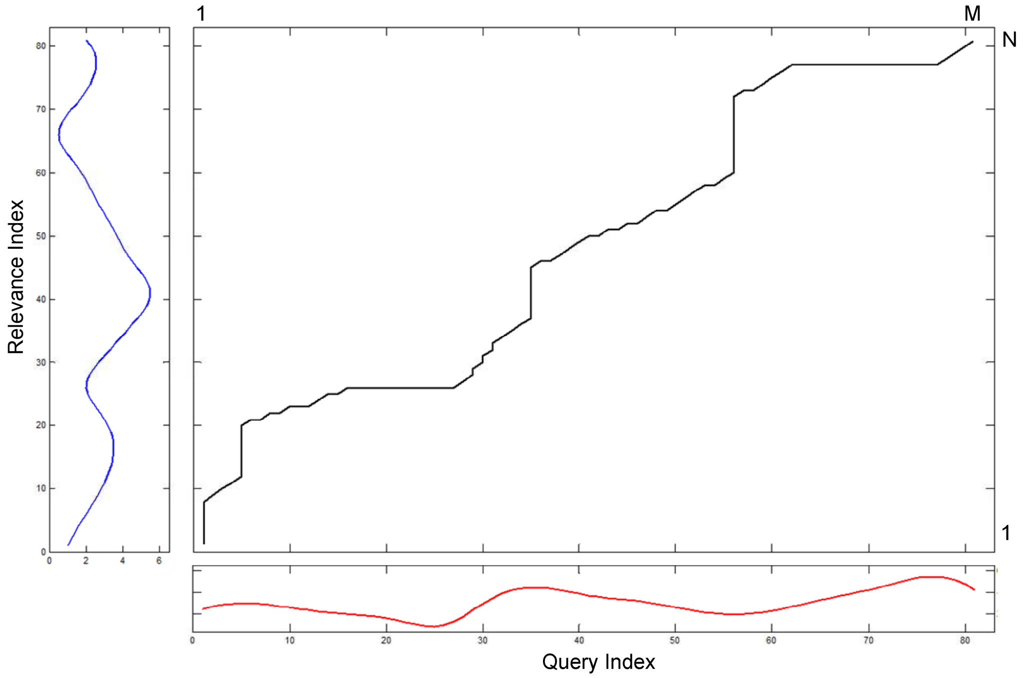


Figure 7.3. The optimal warping path aligning time series from the Figure 7.1.

Since DTW is expensive to calculate, techniques to speed up similarity search have been introduced. The most popular include:

- Global constraints like Sakoe-Chiba band [Sakoe and Chiba, 1990] and Itakura parallelogram [Itakura, 1990].
- Lower bounding techniques [Yi et al., 1998, Kim et al., 2001].

A.2. Optical Flow

Optical flow (image velocity) is a measurement of pixel change between consecutive image frames which are used as a rich source of information in many computer vision tasks including 3D shape acquisition, object recognition and scene understanding. The goal is to compute an approximation to the 2d motion field - a projection of the 3d velocities of surface points onto the image plane - from spatio-temporal patterns of image intensity. A common starting point for differential optical flow estimation [Lucas and Kanade, 1981, Horn and Schunck, 1981, Nagel and Enkelmann, 1986] is to assume that pixel intensities are translated from one frame to the next using:

$$I(p_x, p_y, t) \approx I(p_x + dx, p_y + dy, t + dt) \tag{6.22}$$

where $I(p_x, p_y, t)$ is image intensity as a function of space, t denotes time, whereas dx and dy are displacements of the pixel after time dt . Assuming that the displaced image is well approximated by a first-order Taylor series, the right side of equation (6.22) is expanded:

$$I(p_x + dx, p_y + dy, t + dt) \approx I(p_x, p_y, t) + I_x dx + I_y dy + I_t dt \tag{6.23}$$

where $I_x = \partial I / \partial p_x$ and $I_y = \partial I / \partial p_y$ are spatial partial derivative of the image, whereas $I_t = \partial I / \partial t$ denotes the time partial derivative of the image. By ignoring higher-order terms in the Taylor series and then substituting the linear approximation (6.23) into (6.22) or more generally from an assumption that intensity is conserved $dI(p_x, p_y, t)/dt = 0$, the gradient constraint equation is derived:

$$I_x dx + I_y dy + I_t dt = 0 \tag{6.24}$$

After devision by dt, the 2d image velocity vector $\vec{u} = (dx/dt, dy/dt)$ is obtained:

$$I_x \frac{dx}{dt} + I_y \frac{dy}{dt} + I_t = I_x u_x + I_y u_y + I_t = \nabla I \cdot \vec{u} + I_t = 0 \tag{6.25}$$

and $\nabla I = (I_x, I_y)$ is the spatial intensity gradient. Of course, the above equation is heavily under constrained since we have two unknowns; therefore some additional constraints are required to solve it.

One common way to further constrain $\vec{u}$ is to use gradient constraints from local neighbourhood pixels, assuming that they share the same constant 2D velocity. This is achieved by solving the basic optical flow equations for all the pixels in that neighbourhood using the weighted least squares estimator like in Lucas and Kanade method [Lucas and Kanade, 1981] (Figure 7.4). Alternatively, Horn and Schunck [Horn and Schunck, 1981] combines the gradient constraint (6.25) with a global smoothness term to constrain the estimated velocity field over image domain. Nagel and Enkelmann [Nagel and Enkelmann, 1986] extends that work and suggests an *oriented-smoothness constraint* in spatio-temporal domain, in which the optic flow is only smoothed in the direction perpendicular to the image brightness gradient, so that discontinuity boundaries are much better preserved.

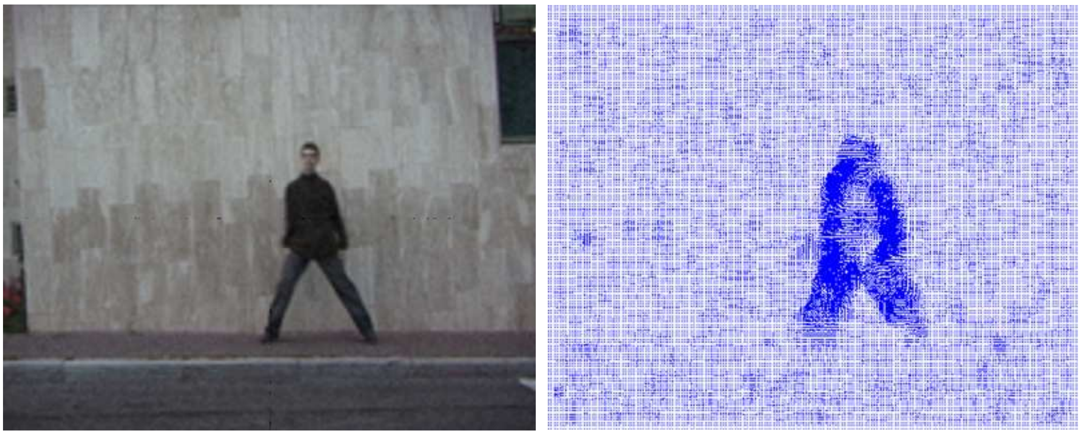


Figure 7.4. Estimation of the optical flow using Lucas and Kanade method.

The discussed approaches are well established algorithms for the optical flow estimation, however many other methods have been proposed. Further details and more comprehensive review can be found in [Barron et al., 1994, Beauchemin and Barron, 1995, Fleet and Weiss, 2006].

A.3. Hausdorff Distance

The Hausdorff distance is a simple metric for measuring similarity between two arbitrary high dimensional curves $A = (a_1, a_2, ..., a_N)$ $(a_i \in \mathbb{R}^D)$ and $B = (b_1, b_2, ..., b_M)$ $(b_i \in \mathbb{R}^D)$. In contrast to the DTW, the computation of this metric does not involved determining an explicit correspondence of points. By definition the Hausdorff distance is the maximum distance of a sequence to the nearest point in the other sequence [*Rote, 1991, Huttenlocher et al., 1993*]. More formally, the Hausdorff distance between time series A and B is a maximin function, defined as:

$$H(A,B) = \max_{a \in A}\{\min_{b \in B}\{\|a-b\|\}\} \tag{6.26}$$

It should be noted that the Hausdorff distance is oriented, i.e. asymmetric, which means that most of the time $H(A,B)$ is not equal to $H(B,A)$ (Figure 7.5). However, in a classification task, a distance is expected to be symmetric to adequately express similarity between two sequences. To tackle this problem a variant, called the symmetric median Hausdorff Distance, was proposed [Gorelick et al., 2007, Wang and Suter, 2007a]:

$$H'(A,B) = \underset{a \in A}{\mathrm{median}}\{\min_{b \in B}\{\|a-b\|\}\} \tag{6.27}$$

$$H(A,B) = H'(A,B) + H'(B,A) \tag{6.28}$$

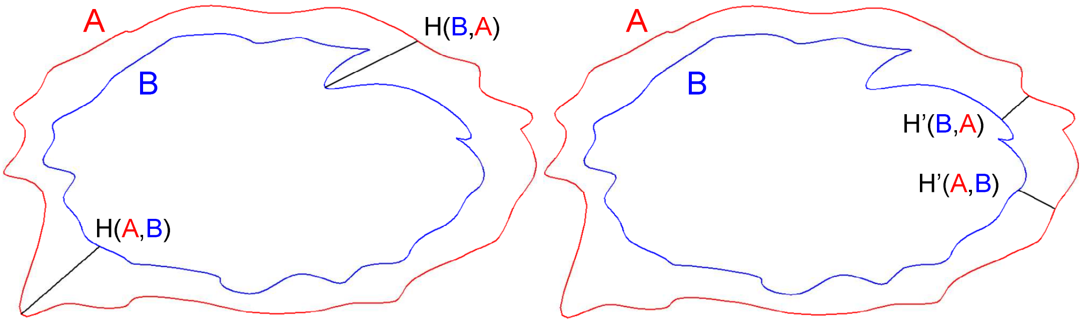


Figure 7.5. Hausdorff distance on toy example between two sequences: standard definition based on equation (6.26) (left) and median variation based on equation (6.27) (right).

References

[5DT, 2011] 5DT (2011). The fifth dimension technologies data glove (http://www.5dt.com/) [last accessed on 12/01/2011].

[Agarwal and Triggs, 2006] Agarwal, A. and Triggs, B. (2006). Learning methods for recovering 3d human pose from monocular images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(1):44–58.

[Aggarwal et al., 2001] Aggarwal, C., Hinneburg, A., and Keim, D. (2001). On the surprising behavior of distance metrics in high dimensional spaces. *Lecture Notes in Computer Science*, 1973:420–434.

[Aha et al., 1991] Aha, D., Kibler, D., and Albert, M. (1991). Instance-based learning algorithms. *Machine Learning*, 6(1):37–66.

[Ahmad and Lee, 2008] Ahmad, M. and Lee, S.-W. (2008). Human action recognition using shape and clg-motion flow from multi-view image sequences. *Pattern Recognition*, 41(7):2237–2252.

[Ali et al., 2007] Ali, S., Basharat, A., and Shah, M. (2007). Chaotic invariants for human action recognition. *Proceedings of the 11th International Conference on Computer Vision*, pages 1–8.

[Almuallim and Dietterich, 1994] Almuallim, H. and Dietterich, T. (1994). Learning boolean concepts in the presence of many irrelevant features. *Artificial Intelligence*, 69(1-2):279–305.

[Arnoldi, 1951] Arnoldi, W. (1951). The principle of minimized iterations in the solution of the matrix eigenvalue problem. *Quarterly of Applied Mathematics*, 9:17–25.

[Backer et al., 1998] Backer, S., Naud, A., and Scheunders, P. (1998). Non-linear dimensionality reduction techniques for unsupervised feature extraction. *Pattern Recognition Letters*, 19:711–720.

[Bai et al., 2000] Bai, Z., Demmel, J., Dongarra, J., Ruhe, R., and van der Vorst, H. (2000). *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*. Society for Industrial and Applied Mathematics.

[Barrón and Kakadiaris, 2003] Barrón, C. and Kakadiaris, I. A. (2003). On the improvement of anthropometry and pose estimation from a single uncalibrated image. *Machine Vision and Applications*, 14(4):229–236.

[Barron et al., 1994] Barron, J., Fleet, D., and Beauchemin, S. (1994). Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43–77.

[Bartholomew, 1984] Bartholomew, D. (1984). The foundations of factor analysis. *Biometrika*, 71(2):221.

[Bartholomew, 1987] Bartholomew, D. J. (1987). *Latent Variable Models and Factor Analysis*. Charles Griffin & Company Ltd.

[Battiti and Masulli, 1990] Battiti, R. and Masulli, F. (1990). Bfgs optimization for faster and automated supervised learning. *Proceedings of the International Neural Network Conference*, pages 757–760.

[Baum et al., 1970] Baum, L., Petrie, T., Soules, G., and Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains. *The Annals of Mathematical Statistics*, 41(1):164–171.

[Beauchemin and Barron, 1995]Beauchemin, S. and Barron, J. (1995). The computation of optical flow. *ACM Computing Surveys*, 27(3):433–466.

[Belkin and Niyogi, 2002] Belkin, M. and Niyogi, P. (2002). Laplacian eigenmaps and spectral techniques for embedding and clustering. *Advances in Neural Information Processing Systems*, 14:585–591.

[Belkin and Niyogi, 2003] Belkin, M. and Niyogi, P. (2003). Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6):1373–1396.

[Belkin and Niyogi, 2007] Belkin, M. and Niyogi, P. (2007). Convergence of laplacian eigenmaps. *Advances in Neural Information Processing Systems*, 19:129.

[Bell and Wang, 2000] Bell, D. and Wang, H. (2000). A formalism for relevance and its application in feature subset selection. *Machine Learning*, 41(2):175–195.

[Bellman, 1961] Bellman, R. (1961). *Adaptive control processes - A guided tour*. Princeton University Press.

[Belongie et al., 2002]Belongie, S., Malik, J., and Puzicha, J. (2002). Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 509–522.

[Bengio et al., 2003] Bengio, Y., Paiement, J.-F., and Vincent, P. (2003). Out-of-sample extensions for lle, isomap, mds, eigenmaps, and spectral clustering. *Advances in Neural Information Processing Systems*, pages 177–184.

[Bentley, 1975] Bentley, J. (1975). Multidimensional binary search trees used for associative searching. *Communications of the ACM*, 18(9):509–517.

[Beyer et al., 1999] Beyer, K., Goldstein, J., Ramakrishnan, R., and Shaft, U. (1999). When is "nearest neighbor" meaningful? *Proceedings of the 7th International Conference on Database Theory*, 1540:217–235.

[Bicego et al., 2009] Bicego, M., Pekalska, E., Tax, D., and Duin, R. (2009). Component-based discriminative classification for hidden markov models. *Pattern Recognition*, 42(11):2637–2648.

[Biesiada and Duch, 2005] Biesiada, J. and Duch, W. (2005). Feature selection for high-dimensional data: A kolmogorov-smirnov correlation-based filter. *Computer Recognition Systems*, 30:95–103.

[Biesiada and Duch, 2007] Biesiada, J. and Duch, W. (2007). Feature selection for high-dimensional data: A pearson redundancy based filter. *Computer Recognition Systems*, 45:242–249.

[Bishop, 1995] Bishop, C. (1995). *Neural Networks for Pattern Recognition*. Oxford University Press.

[Bishop, 1997] Bishop, C. (1997). Gtm through time. *In 5th International Conference on Artificial Neural Networks*, pages 111–116.

[Bishop, 1999] Bishop, C. (1999). *Latent Variable Models*. Learning in graphical models. MIT Press.

[Bishop et al., 1998] Bishop, C., Svensén, M., and Williams, C. (1998). Gtm: The generative topographic mapping. *Neural computation*, 10(1):215–234.

[Bitzer and Vijayakumar, 2009]Bitzer, S. and Vijayakumar, S. (2009). Latent spaces for dynamic movement primitives. *Proceedings of 9th International Conference on Humanoid Robots*.

[Blackburn and Ribeiro, 2007] Blackburn, J. and Ribeiro, E. (2007). Human motion recognition using isomap and dynamic time warping. *Lecture Notes in Computer Science*, 4814:285–298.

[Blum and Langley, 1997] Blum, A. and Langley, P. (1997). Selection of relevant features and examples in machine learning. *Artificial intelligence*, 97(1-2):245–271.

[Bobick and Davis, 2001] Bobick, A. and Davis, J. (2001). The recognition of human movement using temporal templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(3):257–267.

[Böhm et al., 2009] Böhm, C., Läer, L., Plant, C., and Zherdin, A. (2009). Model-based classification of data with time series-valued attributes. *Proceedings of the 13th Business, Technology und Web Conference*, 144:287–296.

[Boser et al., 1992]Boser, B., Guyon, I., and Vapnik, V. (1992). A training algorithm for optimal margin classifiers. *Proceedings of the 5th Annual Workshop on Computational Learning Theory*, pages 144–152.

[Boyd and Vandenberghe, 2004] Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press.

[Brassard and Bratley, 1996]Brassard, G. and Bratley, P. (1996). *Fundamentals of Algorithms*. Prentice-Hall International.

[Brendel and Todorovic, 2010] Brendel, W. and Todorovic, S. (2010). Activities as time series of human postures. *Proceedings of the 11th European Conference on Computer Vision*.

[Brubaker et al., 2010] Brubaker, M., Sigal, L., and Fleet, D. (2010). *Video-Based People Tracking*, volume 2 of *Handbook of Ambient Intelligence and Smart Environments*. Springer-Verlag.

[Burges, 1998] Burges, C. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167.

[Caillette et al., 2005] Caillette, F., Galata, A., and Howard, T. (2005). Real-time 3d human body tracking using variable length markov models. *Proceedings of the 16th British Machine Vision Conference*, 1:469–478.

[Caillette et al., 2008] Caillette, F., Galata, A., and Howard, T. (2008). Real-time 3-d human body tracking using learnt models of behaviour. *Computer Vision and Image Understanding*, 109(2):112–125.

[Camastra, 2003] Camastra, F. (2003). Data dimensionality estimation methods: A survey. *Pattern Recognition*, 36(12):2945–2954.

[Camastra and Vinciarelli, 2001] Camastra, F. and Vinciarelli, A. (2001). Intrinsic dimension estimation of data: An approach based on grassberger-procaccias algorithm. *Neural Processing Letters*, 14(1):27–34.

[Camastra and Vinciarelli, 2002] Camastra, F. and Vinciarelli, A. (2002). Estimating the intrinsic dimension of data with a fractal-based method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(10):1404–1407.

[Camastra and Vinciarelli, 2008] Camastra, F. and Vinciarelli, A. (2008). *Machine Learning for Audio, Image and Video Analysis: Theory and Applications*. Advanced Information and Knowledge Processing. Springer Verlag.

[Cao and Saha, 2005] Cao, Y. and Saha, P. (2005). Improved branch and bound method for control structure screening. *Chemical Engineering Science*, 60(6):1555–1564.

[Cardie, 1993] Cardie, C. (1993). Using decision trees to improve case-based learning. *Proceedings of the 10th International Conference on Machine Learning*, 25:32.

[Carreira-Perpinán, 2001] Carreira-Perpinán, M. (2001). *Continuous Latent Variable Models for Dimensionality Reduction and Sequential Data Reconstruction*. Phd thesis, University of Sheffield, Sheffield, UK.

[Cham and Rehg, 1999] Cham, T. and Rehg, J. (1999). A multiple hypothesis approach to figure tracking. *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 2:239–245.

[Chatfield, 1996] Chatfield, C. (1996). *The Analysis of Time Series - An Introduction*. Chapman and Hall, 5 edition.

[Chen et al., 1991] Chen, S., Cowan, C., and Grant, P. (1991). Orthogonal least squares learning algorithm for radial basis function networks. *IEEE Transactions on Neural Networks*, 2(2):302–309.

[Chen, 2003] Chen, X. (2003). An improved branch and bound algorithm for feature selection. *Pattern Recognition Letters*, 24(12):1925–1933.

[Chen and Chai, 2009] Chen, Y. and Chai, J. (2009). 3d reconstruction of human motion and skeleton from uncalibrated monocular video. *Proceedings of the Asian Conference on Computer Vision*, pages 71–82.

[Cheung et al., 2005] Cheung, K.-M., Baker, S., and Kanade, T. (2005). Shape-from-silhouette across time part i: Theory and algorithms. *International Journal of Computer Vision*, 62:221–247.

[Chin and Suter, 2008] Chin, T. and Suter, D. (2008). Out-of-sample extrapolation of learned manifolds. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(9):1547–1556.

[Chin et al., 2007] Chin, T., Wang, L., Schindler, K., and Suter, D. (2007). Extrapolating learned manifolds for human activity recognition. *Proceedings of the International Conference on Image Processing*, 1.

[Choi and Choi, 2007]Choi, H. and Choi, S. (2007). Robust kernel isomap. *Pattern Recognition*, 40(3):853–862.

[CMU, 2010] CMU (2010). Carnegie mellon university graphics lab motion capture database (http://mocap.cs.cmu.edu/) [last accessed on 14/10/2010].

[Cohen and Li, 2003] Cohen, I. and Li, H. (2003). Inference of human postures by classification of 3d human body shape. *Proceedings of the International Workshop on Analysis and Modeling of Faces and Gestures*.

[Colak and Isik, 2003]Colak, S. and Isik, C. (2003). Feature subset selection for blood pressure classification using orthogonal forward selection. *Proceedings of the 29th Annual Bioengineering Conference*, pages 122—123.

[Cortes and Vapnik, 1995] Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3):273–297.

[Costa and Hero, 2004] Costa, J. and Hero, A. (2004). Geodesic entropic graphs for dimension and entropy estimation in manifold learning. *IEEE Transactions on Signal Processing*, 52(8):2210–2221.

[Cotter et al., 2001] Cotter, S., Kreutz-Delgado, K., and Rao, B. (2001). Backward sequential elimination for sparse vector subset selection. *Signal Processing*, 81(9):1849–1864.

[Courant and Hilbert, 1953] Courant, R. and Hilbert, D. (1953). *Methods of Mathematical Physics*, volume 1. Interscience.

[Cover and Thomas, 1991] Cover, T. M. and Thomas, J. (1991). *Elements of Information Theory*. Wiley-Interscience.

[Cox and Cox, 1994] Cox, T. and Cox, M. (1994). *Multidimensional Scaling*. Chapman & Hall.

[Cutler and Davis, 2000] Cutler, R. and Davis, L. (2000). Robust real-time periodic motion detection, analysis, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):781–796.

[Dalal and Triggs, 2005] Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 1:886–893.

[Darby et al., 2010] Darby, J., Li, B., and Costen, N. (2010). Tracking human pose with multiple activity models. *Pattern Recognition*, 43:3042–3058.

[Dash and Liu, 1997] Dash, M. and Liu, H. (1997). Feature selection for classification. *Intelligent Data Analysis*, 1:131–156.

[Dash and Liu, 2003] Dash, M. and Liu, H. (2003). Consistency-based search in feature selection. *Artificial Intelligence*, 151(1-2):155–176.

[De Ridder et al., 2003] De Ridder, D., Kouropteva, O., Okun, O., Pietikainen, M., and Duin, R. (2003). Supervised locally linear embedding. *Lecture Notes in Computer Science*, 2714:333–341.

[De Silva and Tenenbaum, 2003] De Silva, V. and Tenenbaum, J. (2003). Global versus local methods in nonlinear dimensionality reduction. *Advances in Neural Information Processing Systems*, 15:705–712.

[Deena and Galata, 2009]Deena, S. and Galata, A. (2009). Speech-driven facial animation using a shared gaussian process latent variable model. *Proceedings of the 5th International Symposium on Advances in Visual Computing*, pages 89–100.

[Dempster et al., 1977] Dempster, A., Laird, N., and Rubin, D. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society*, 39(1):1–38.

[Deutscher et al., 2000] Deutscher, J., Blake, A., and Reid, I. (2000). Articulated body motion capture by annealed particle filtering. *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 2:126—33.

[Deutscher and Reid, 2005] Deutscher, J. and Reid, I. (2005). Articulated body motion capture by stochastic search. *International Journal of Computer Vision*, 61(2):185–205.

[Devijver and Kittler, 1982] Devijver, P. and Kittler, J. (1982). *Pattern Recognition: A Statistical Approach*. Prentice-Hall International.

[Dijkstra, 1959]Dijkstra, E. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik*, 1(1):269–271.

[Ding and Peng, 2003] Ding, C. and Peng, H. (2003). Minimum redundancy feature selection from microarray gene expression data. *Proceedings of the IEEE Computer Society Conference on Bioinformatics*, page 523.

[Do Carmo, 1976] Do Carmo, M. (1976). *Differential Geometry of Curves and Surfaces*. Prentice-Hall International.

[Doak, 1992] Doak, J. (1992). An evaluation of feature selection methods and their application to computer security. Technical report, University of California at Davis Technical Report.

[Dollar et al., 2005] Dollar, P., Rabaud, V., Cottrell, G., and Belongie, S. (2005). Behavior recognition via sparse spatio-temporal features. *Proceedings of the 14th International Conference on Computer Communications and Networks*, pages 65–72.

[Dongen, 2000]Dongen, S. (2000). *Graph Clustering by Flow Simulation*. Phd thesis, University of Utrecht, Utrecht, Netherlands.

[Donoho and Grimes, 2003] Donoho, D. and Grimes, C. (2003). Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data. *Proceedings of the National Academy of Sciences*, 100(10):5591.

[Draminski et al., 2008] Draminski, M., Rada-Iglesias, A., Enroth, S., Wadelius, C., Koronacki, J., and Komorowski, J. (2008). Monte carlo feature selection for supervised classification. *Bioinformatics*, 24(1):110.

[Easterby et al., 1982] Easterby, R., Kroemer, K., and Chaffin, D. (1982). *Anthropometry and Biomechanics: Theory and Application*. Plenum Press.

[Efros et al., 2003] Efros, A., Berg, A., Mori, G., and Malik, J. (2003). Recognizing action at a distance. *Proceedings of the 9th International Conference on Computer Vision*, 2.

[Ek et al., 2009] Ek, C., Jaeckel, P., Campbell, N., Lawrence, N., and Melhuish, C. (2009). Shared gaussian process latent variable models for handling ambiguous facial expressions. *AIP Conference Proceedings*, 1107.

[Ek et al., 2007] Ek, C., Torr, P., and Lawrence, N. (2007). Gaussian process latent variable models for human pose estimation. *Proceedings of the 4th International Conference on Machine Learning for Multimodal Interaction*, pages 132–143.

[Elgammal and Lee, 2004a] Elgammal, A. and Lee, C. (2004a). Inferring 3d body pose from silhouettes using activity manifold learning. *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 3:681–688.

[Elgammal and Lee, 2004b] Elgammal, A. and Lee, C. (2004b). Separating style and content on a nonlinear manifold. *International Conference on Computer Vision and Pattern Recognition*, 1.

[Elgammal and Lee, 2007] Elgammal, A. and Lee, C. (2007). Nonlinear manifold learning for dynamic shape and dynamic appearance. *Computer vision and Image Understanding*, 106(1):31–46.

[Elgammal and Lee, 2009] Elgammal, A. and Lee, C.-S. (2009). Tracking people on a torus. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(3):520–538.

[Errity, 2010] Errity, A. (2010). *Exploring the Dimensionality of Speech using Manifold Learning and Dimensionality Reduction Methods*. Phd thesis, City University, Dublin, Ireland.

[Everitt, 1984] Everitt, B. (1984). *An Introduction to Latent Variable Models*. Monographs on Statistics and Applied Probability. Chapman & Hall.

[Everitt and Hand, 1981] Everitt, B. and Hand, D. (1981). *Finite Mixture Distributions*. Monographs on Statistics and Applied Probability. Chapman & Hall.

[Fan et al., 2010] Fan, M., Gu, N., Qiao, H., and Zhang, B. (2010). Intrinsic dimension estimation of data by principal component analysis. *Computing Research Repository*.

[Fan et al., 2009] Fan, M., Qiao, H., and Zhang, B. (2009). Intrinsic dimension estimation of manifolds by incising balls. *Pattern Recognition*, 42(5):780–787.

[Fang et al., 2009] Fang, C., Chen, J., Tseng, C., and Lien, J. (2009). Human action recognition using spatio-temporal classification. *Proceedings of the 9th Asian Conference on Computer Vision*, pages 98–109.

[Farnell, 1999] Farnell, B. (1999). Moving bodies, acting selves. *Annual Review of Anthropology*, 28:341–373.

[Felzenszwalb et al., 2010] Felzenszwalb, P., Girshick, R., McAllester, D., and Ramanan, D. (2010). Object detection with discriminatively trained part based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9).

[Feng and Perona, 2002] Feng, X. and Perona, P. (2002). Human action recognition by sequence of movelet codewords. *Proceedings of the International Symposium on 3D Data Processing, Visualization, and Transmission*, pages 717–721.

[Ferris et al., 2007]Ferris, B., Fox, D., and Lawrence, N. (2007). Wifi-slam using gaussian process latent variable models. *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pages 2480–2485.

[Fishman, 1996] Fishman, G. (1996). *Monte Carlo: Concepts, Algorithms, and Applications*. Springer Verlag.

[Fleet and Weiss, 2006] Fleet, D. and Weiss, Y. (2006). Optical flow estimation. *Handbook of Mathematical Models in Computer Vision*, pages 237–257.

[Floyd, 1962] Floyd, R. (1962). Algorithm 97: Shortest path. *Communications of the ACM*, 5(6):345.

[Fokkema et al., 1999] Fokkema, D., Sleijpen, G., and Van der Vorst, H. (1999). Jacobi-davidson style qr and qz algorithms for the partial reduction of matrix pencils. *SIAM Journal on Scientific Computing*, 20(1):94–125.

[Forman, 2003] Forman, G. (2003). An extensive empirical study of feature selection metrics for text classification. *The Journal of Machine Learning Research*, 3:1289–1305.

[Forman, 2008] Forman, G. (2008). Bns feature scaling: An improved representation over tf-ldf for svm text classification. *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, pages 263–270.

[Francois et al., 2007] Francois, D., Wertz, V., and Verleysen, M. (2007). The concentration of fractional distances. *IEEE Transactions on Knowledge and Data Engineering*, 19:873–886.

[Frenkel and Basri, 2003]Frenkel, M. and Basri, R. (2003). Curve matching using the fast marching method. *Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 35–51.

[Fuentes and Velastin, 2001]Fuentes, L. and Velastin, S. (2001). Foreground segmentation using luminance contrast. *Proceedings of the WSES International Conference on Speech, Signal and Image Processing*, 240:241–243.

[Fukunaga, 1982] Fukunaga, K. (1982). Intrinsic dimensionality extraction. *Classification, Pattern Recognition and Reduction of Dimensionality*, pages 347–362.

[Fukunaga, 1990] Fukunaga, K. (1990). *Introduction to Statistical Pattern Recognition*. Computer Science and Scientific Computing Series. Academic Press, 2 edition.

[Fukunaga and Olsen, 1971] Fukunaga, K. and Olsen, D. R. (1971). An algorithm for finding intrinsic dimensionality of data. *IEEE Transactions on Computers*, C-20(2):176–183.

[Ghahramani and Hinton, 1997]Ghahramani, Z. and Hinton, G. (1997). The em algorithm for mixtures of factor analyzers. Crg-tr-96-1, University of Toronto Technical Report.

[Gheyas and Smith, 2010] Gheyas, I. and Smith, L. (2010). Feature subset selection in large dimensionality domains. *Pattern Recognition*, 43(1):5–13.

[Goldberg and Ritov, 2009] Goldberg, Y. and Ritov, Y. (2009). Local procrustes for manifold embedding: A measure of embedding quality and embedding algorithms. *Machine Learning*, 77(1):1–25.

[Good, 1973] Good, I. (1973). What are degrees of freedom? *The American Statistician*, 27(5):227–228.

[Gordon et al., 1993] Gordon, N., Salmond, D., and Smith, A. (1993). Novel approach to nonlinear/non-gaussian bayesian state estimation. *IEE Proceedings*, 140(2):107–113.

[Gorelick et al., 2007] Gorelick, L., Blank, M., Shechtman, E., Irani, M., and Basri, R. (2007). Actions as space-time shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(12):2247.

[Gorelick et al., 2006] Gorelick, L., Galun, M., Sharon, E., Basri, R., and Brandt, A. (2006). Shape representation and classification using the poisson equation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(12):1991–2005.

[Green, 1963] Green, D. (1963). On the effectiveness of receptors in recognition systems. *IEEE Transactions on Information Theory*, 9(1):11–17.

[Grochow et al., 2004] Grochow, K., Martin, S., Hertzmann, A., and Popovic, Z. (2004). Style-based inverse kinematics. *ÁCM Transactions on Graphics*, 23(3):522–531.

[Gross and Shi, 2001] Gross, R. and Shi, J. (2001). The cmu motion of body (mobo) database. Cmu-ri-tr-01-18, Carnegie Mellon University Technical Report.

[Guo and Qian, 2008] Guo, F. and Qian, G. (2008). Monocular 3d tracking of articulated human motion in silhouette and pose manifolds. *Journal on Image and Video Processing*, 2008:1–18.

[Gupta et al., 2008] Gupta, A., Chen, T., Chen, F., Kimber, D., and Davis, L. (2008). Context and observation driven latent variable model for human pose estimation. *International Conference on Computer Vision and Pattern Recognition*, pages 1–8.

[Gutierrez-Osuna, 2006] Gutierrez-Osuna, R. (2006). Introduction to pattern analysis. *Texas A&M University*. Lecture.

[Guyon and Elisseeff, 2003] Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research,* 3:1157–1182.

[Haifeng et al., 2006] Haifeng, G., Chunhong, P., Qing, Y., Hanqing, L., and Songde, M. (2006). Neural network modeling of spectral. embedding. *Proceedings of the 17th British Machine Vision Conference*.

[Hall, 2000] Hall, M. (2000). Correlation-based feature selection for discrete and numeric class machine learning. *Proceedings of the 17th International Conference on Machine Learning*, pages 359–366.

[Hamilton, 1844] Hamilton, W. (1844). On quaternions, or on a new system of imaginaries in algebra. *Philosophical Magazine*, 25(3):489–495.

[Hannan, 1970] Hannan, E. J. (1970). *Multiple Time Series*. Wiley.

[Harris and Stephens, 1988] Harris, C. and Stephens, M. (1988). A combined corner and edge detection. *Proceedings of the 4th Alvey Vision Conference*, pages 147–151.

[Haykin, 1998] Haykin, S. (1998). *Neural Networks: A Comprehensive Foundation (2nd Edition)*. Prentice-Hall International, 2 edition.

[He et al., 2005] He, X., Cai, D., Yan, S., and Zhang, H. (2005). Neighborhood preserving embedding. *Proceedings of the 10th International Conference on Computer Vision*, 2:1208–1213.

[He et al., 2004] He, X., Ma, W.-Y., and Zhang, H.-J. (2004). Learning an image manifold for retrieval. *Proceedings of the 12th Annual ACM International Conference on Multimedia*, pages 17–23.

[He and Niyogi, 2004a] He, X. and Niyogi, P. (2004a). Locality preserving projections. *Advances in Neural Information Processing Systems*, 16.

[He and Niyogi, 2004b] He, X. and Niyogi, P. (2004b). Locality preserving projections. *Advances in Neural Information Processing Systems*, 16:153–160.

[Hinneburg et al., 2000] Hinneburg, A., Aggarwal, C., and Keim, D. (2000). What is the nearest neighbor in high dimensional spaces? *Proceedings of the 26th International Conference on Very Large Data Bases*, pages 506–515.

[Hirsch, 1976] Hirsch, M. (1976). *Differential Topology*, volume 33 of *Graduate Texts in Mathematics*. Springer Verlag.

[Holland, 1975]Holland, J. (1975). *Adaptation in natural and artificial systems*. University of Michigan Press.

[Horn and Schunck, 1981] Horn, B. and Schunck, B. (1981). Determining optical flow. *Artificial Intelligence*, 17(1-3):185–203.

[Horn and Johnson, 1985] Horn, R. and Johnson, C. (1985). *Matrix Analysis*. Cambridge University Press.

[Hotelling, 1933] Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of Education Psychology*, 24:417–441.

[Hou et al., 2009] Hou, C., Chenping, Z., Wu, Y., and Jiao, Y. (2009). Stable local dimensionality reduction approaches. *Pattern Recognition*, 42(9).

[Hou et al., 2007] Hou, S., Galata, A., Caillette, F., Thacker, N., and Bromiley, P. (2007). Real-time body tracking using a gaussian process latent variable model. *Proceedings of the 11th International Conference on Computer Vision*.

[Hsu and Lin, 2002] Hsu, C. and Lin, C. (2002). A comparison of methods for multiclass support vector machines. *IEEE Transactions on Neural Networks*, 13(2):415–425.

[Hua et al., 2009] Hua, J., Tembe, W., and Dougherty, E. (2009). Performance of feature-selection methods in the classification of high-dimension data. *Pattern Recognition*, 42(3):409–424.

[Huan and Hiroshi, 1998]Huan, L. and Hiroshi, M. (1998). *Feature Selection for Knowledge Discovery and Data Mining*. Kluwer Academic Publishers.

[HumanEvaI, 2010] HumanEvaI (2010). Brown university image & mocap synchronized dataset (http://www.cs.brown.edu/ ls/software/index.html) [last accessed on 15/12/2010].

[Huttenlocher et al., 1993] Huttenlocher, D., Klanderman, G., and Rucklidge, W. (1993). Comparing images using the hausdorff distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9):850–863.

[Ikizler-Cinbis and Sclaroff, 2010] Ikizler-Cinbis, N. and Sclaroff, S. (2010). Object, scene and actions: Combining multiple features for human action recognition. *Proceedings of the 11th European Conference on Computer Vision*.

[Inition, 2011] Inition (2011). Ascension flock-of-birds magnetic position tracker (http://www.inition.co.uk/) [last accessed on 12/01/2011].

[Isard and Blake, 1998] Isard, M. and Blake, A. (1998). Condensation - conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1):5–28.

[Itakura, 1990] Itakura, F. (1990). Minimum prediction residual principle applied to speech recognition. *Readings in Speech Recognition*, pages 154–158.

[Jackson, 1991]Jackson, J. (1991). *A User's Guide to Principal Components*. Wiley Series in Probability and Mathematical Statistics. Wiley-Interscience.

[Jafari and Almasganj, 2010] Jafari, A. and Almasganj, F. (2010). Using laplacian eigenmaps latent variable model and manifold learning to improve speech recognition accuracy. *Speech Communication*.

[Jain et al., 2000] Jain, A., Duin, R., and Mao, J. (2000). Statistical pattern recognition: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1):4–37.

[Jain and Zongker, 1997] Jain, A. and Zongker, D. (1997). Feature selection: Evaluation, application and small sample performance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(2):153–158.

[Jain and Saul, 2004] Jain, V. and Saul, L. (2004). Exploratory analysis and visualization of speech and music by locally linear embedding. *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, 3:984–987.

[Jenkins and Mataric, 2004] Jenkins, O. and Mataric, M. (2004). Á spatio-temporal extension to isomap nonlinear dimension reduction. *Proceedings of the 21st International Conference on Machine Learning*, pages 441–448.

[Jhuang et al., 2007] Jhuang, H., Serre, T., Wolf, L., and Poggio, T. (2007). A biologically inspired system for action recognition. *Proceedings of the 11th International Conference on Computer Vision*, 1(2):1–8.

[Ji and Liu, 2010] Ji, X. and Liu, H. (2010). Advances in view-invariant human motion analysis: A review. *IEEE Transactions on Systems, Man, and Cybernetics*, 40(1):13–24.

[Jia and Yeung, 2008] Jia, K. and Yeung, D. (2008). Human action recognition using local spatio-temporal discriminant embedding. *International Conference on Computer Vision and Pattern Recognition*, pages 1–8.

[Jian and Enhua, 2005] Jian, C. and Enhua, W. (2005). 3d human motion reconstruction from monocular videos through iterative optimization. *Journal of Computer-Aided Design & Computer Graphics*, 17(7):1523–1528.

[Jimenez and Landgrebe, 1998] Jimenez, L. and Landgrebe, D. (1998). Supervised classification in high dimensional space: Geometrical, statistical, and asymptotical properties of multivariate data. *IEEE Transactions on System, Man, and Cybernetics*, 28:39–54.

[Jing et al., 2008] Jing, W., Yafeng, Y., and Hong, M. (2008). Multiple human tracking using particle filter with gaussian process dynamical model. *EURASIP Journal on Image and Video Processing*.

[Johansson et al., 1992] Johansson, E., Dowla, F., and Goodman, D. (1992). Backpropagation learning for multi-layer feed-forward neural networks using the conjugate gradient method. *International Journal of Neural Systems*, 2(4):291–301.

[Johansson, 1973] Johansson, G. (1973). Visual perception of biological motion and a model for its analysis. *Perception Psychophys*, 14(2):201–211.

[Jolliffe, 1989] Jolliffe, I. (1989). *Principal Component Analysis*. Springer Verlag.

[Junejo et al., 2008] Junejo, I., Dexter, E., Laptev, I., and Pérez, P. (2008). Cross-view action recognition from temporal self-similarities. *Proceedings of the 10th European Conference on Computer Vision*, 12.

[Kaâniche and Brémond, 2009] Kaâniche, M. and Brémond, F. (2009). Tracking hog descriptors for gesture recognition. *Proceedings of the 6th International Conference on Advanced Video and Signal Based Surveillance*, pages 140–145.

[Kaâniche and Brémond, 2010] Kaâniche, M. and Brémond, F. (2010). Gesture recognition by learning local motion signatures. *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 2745–2752.

[Kadous and Sammut, 2005] Kadous, M. and Sammut, C. (2005). Classification of multivariate time series and structured data using constructive induction. *Journal Machine Learning*, 58:179–216.

[Kalman, 1960] Kalman, R. (1960). A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(1):35–45.

[Kanaujia et al., 2007] Kanaujia, A., Sminchisescu, C., and Metaxas, D. (2007). Spectral latent variable models for perceptual inference. *Proceedings of the 11th International Conference on Computer Vision*, pages 1–8.

[Kanungo et al., 2002] Kanungo, T., Mount, D., Netanyahu, N., Piatko, C., Silverman, R., and Wu, A. (2002). A local search approximation algorithm for k-means clustering. *Proceedings of the 18th Annual Symposium on Computational Geometry*, pages 10–18.

[Karbauskait et al., 2007] Karbauskait, R., Kurasova, O., and Dzemyda, G. (2007). Selection of the number of neighbours of each data point for the locally linear embedding algorithm. *Information Technology and Control*, 36(4):359–364.

[Kegl, 2003]Kegl, B. (2003). Intrinsic dimension estimation using packing numbers. *Advances in Neural Information Processing Systems*, 15:681–688.

[Kellokumpu et al., 2008] Kellokumpu, V., Zhao, G., and Pietikäinen, M. (2008). Human activity recognition using a dynamic texture based method. *Proceedings of the 19th British Machine Vision Conference*, pages 885–894.

[Kilian et al., 2005] Kilian, Q., Benjamin, D., and Lawrence, K. (2005). Nonlinear dimensionality reduction by semidefinite programming and kernel matrix factorization. *Proceedings of the 10th International Workshop on Artificial Intelligence and Statistics*, pages 381–388.

[Kim et al., 2001] Kim, S.-W., Park, S., and Chu, W. (2001). An index-based approach for similarity search supporting time warping in large sequence databases. *Proceedings of the 17th International Conference on Data Engineering*, pages 607–614.

[King and Forsyth, 2000] King, O. and Forsyth, D. (2000). How does condensation behave with a finite number of samples? *Proceedings of the 6th European Conference on Computer Vision*, pages 695–709.

[Kira and Rendell, 1992] Kira, K. and Rendell, L. (1992). The feature selection problem: Traditional methods and a new algorithm. *Proceedings of the National Conference on Artificial Intelligence*, pages 129–129.

[Kirkpatrick et al., 1983] Kirkpatrick, S., Gelatt, C., and Vecchi, M. (1983). Optimization by simulated annealing. *Science*, 220(4598):671–680.

[Kittler, 1978] Kittler, J. (1978). Feature set search algorithms. *Pattern Recognition and Signal Processing*, pages 41–60.

[Kläser et al., 2008] Kläser, A., Marszaek, M., and Schmid, C. (2008). A spatio-temporal descriptor based on 3d-gradients. *Proceedings of the 19th British Machine Vision Conference*, pages 995–1004.

[Kline, 1998] Kline, M. (1998). *Calculus: an intuitive and physical approach*. Dover Publications, 2 edition.

[Knossow et al., 2008] Knossow, D., Ronfard, R., and Horaud, R. (2008). Human motion tracking with a kinematic parameterization of extremal contours. *International Journal of Computer Vision*, 79(3):247–269.

[Knyazev, 2002] Knyazev, A. (2002). Toward the optimal preconditioned eigensolver: Locally optimal block preconditioned conjugate gradient method. *SIAM Journal on Scientific Computing*, 23(2):517–541.

[Kohavi and John, 1997] Kohavi, R. and John, G. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, 97(1):273–324.

[Kohonen, 1982] Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological cybernetics*, 43(1):59–69.

[Kokiopoulou and Saad, 2007] Kokiopoulou, E. and Saad, Y. (2007). Orthogonal neighborhood preserving projections: A projection-based dimensionality reduction technique. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(12):2143–2156.

[Kononenko, 1994] Kononenko, I. (1994). Estimating attributes: Analysis and extensions of relief. *European Conference on Machine Learning*, pages 171–182.

[Korn et al., 2001] Korn, F., Pagel, B., and Faloutsos, C. (2001). On the “dimensionality curse” and the “self-similarity blessing”. *IEEE Transactions on Knowledge and Data Engineering*, 13(1):96–111.

[Kouropteva et al., 2002] Kouropteva, O., Okun, O., and Pietikainen, M. (2002). Selection of the optimal parameter value for the locally linear embedding algorithm. *Proceedings of the 1st International Conference on Fuzzy Systems and Knowledge Discovery*, pages 359–363.

[Kovashka and Grauman, 2010]Kovashka, A. and Grauman, K. (2010). Learning a hierarchy of discriminative space-time neighborhood features for human action recognition. *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 2046–2053.

[Kraskov et al., 2004] Kraskov, A., Stoegbauer, H., and Grassberger, P. (2004). Estimating mutual information. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 69(6).

[Kruskal, 1964]Kruskal, J. (1964). Non metric multidimensional scaling: a numerical method. *Psychometrika*, 29:115–129.

[Kuo et al., 2009] Kuo, P., Ammar, T., Lewandowski, M., Makris, D., and Nebel, J.-C. (2009). Exploiting human bipedal motion constraints for 3d pose recovery from a single uncalibrated camera. *Proceedings of the 4th International Conference on Computer Vision Theory and Applications*, 1:557–564.

[Kuo et al., 2008] Kuo, P., Makris, D., Megherbi, N., and Nebel, J.-C. (2008). Integration of local image cues for probabilistic 2d pose recovery. *Proceedings of the 4th International Symposium on Advances in Visual Computing*, pages 214–223.

[Kuo et al., 2007] Kuo, P., Nebel, J., and Makris, D. (2007). Camera auto-calibration from articulated motion. *Proceedings of the Conference on Advanced Video and Signal Based Surveillance*, pages 135–140.

[Lafferty et al., 2001] Lafferty, J., McCallum, A., and Pereira, F. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *Proceedings of the 18th International Conference on Machine Learning*, pages 282–289.

[Lapedes and Farber, 1988] Lapedes, A. and Farber, R. (1988). How neural nets work. *Advances in Neural Information Processing Systems*, pages 442–456.

[Laptev, 2005] Laptev, I. (2005). On space-time interest points. *International Journal of Computer Vision*, 64(2-3):107–123.

[Laptev and Lindeberg, 2003] Laptev, I. and Lindeberg, T. (2003). Space-time interest points. *Proceedings of the 9th International Conference on Compu*, pages 432–439.

[Laptev et al., 2008] Laptev, I., Marszalek, M., Schmid, C., and Rozenfeld, B. (2008). Learning realistic human actions from movies. *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 1–8.

[Lathauwer et al., 2000] Lathauwer, L., Moor, B., and Vandewalle, J. (2000). A multilinear singular value decomposition. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1253–1278.

[Lawrence, 2004] Lawrence, N. (2004). Gaussian process latent variable models for visualisation of high dimensional data. *Advances in Neural Information Processing Systems*, 16.

[Lawrence, 2005] Lawrence, N. (2005). Probabilistic non-linear principal component analysis with gaussian process latent variable models. *The Journal of Machine Learning Research*, 6:1816.

[Lawrence, 2007] Lawrence, N. (2007). Learning for larger datasets with the gaussian process latent variable model. *Proceedings of the 11th International Conference on Artificial Intelligence and Statistics*.

[Lawrence and Quinonero-Candela, 2006] Lawrence, N. and Quinonero-Candela, J. (2006). Local distance preservation in the gp-lvm through back constraints. *Proceedings of the 23rd International Conference on Machine Learning*, 148:513–520.

[LeCun, 2000] LeCun, Y. (2000). Mnist handwritten digits dataset (http://yann.lecun.com/exdb/mnist/index.html) [last accessed on 05/10/2010].

[Ledermann and Vajda, 1961] Ledermann, W. and Vajda, S. (1961). *Analysis*, volume 4 of *Handbook of Applicable Mathematics*. Wiley.

[Lee and Elgammal, 2006a] Lee, C. and Elgammal, A. (2006a). Homeomorphic manifold analysis: Learning decomposable generative models for human motion analysis. *Workshop on Dynamical Vision at European Conference on Computer Vision*, pages 100–114.

[Lee and Elgammal, 2006b] Lee, C.-S. and Elgammal, A. (2006b). Simultaneous inference of view and body pose using torus manifolds. *Proceedings of the 18th International Conference on Pattern Recognition*, pages 489–494.

[Lee and Verleysen, 2007] Lee, J. and Verleysen, M. (2007). *Nonlinear Dimensionality Reduction.* Springer Verlag.

[Lee and Cohen, 2006] Lee, M. and Cohen, I. (2006). A model-based approach for estimating human 3d poses in static images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 905–916.

[Levenberg, 1944] Levenberg, K. (1944). A method for the solution of certain problems in least squares. *The Quarterly of Applied Mathematics*, 2:164–168.

[Levina and Bickel, 2005] Levina, E. and Bickel, P. (2005). Maximum likelihood estimation of intrinsic dimension. *Advances in Neural Information Processing Systems*, 17:777–784.

[Lewandowski et al., 2009] Lewandowski, M., Makris, D., and Nebel, J. (2009). Automatic configuration of spectral dimensionality reduction methods for 3d human pose estimation. *Workshop on Visual Surveillance at International Conference on Computer Vision*.

[Lewandowski et al., 2010a] Lewandowski, M., Makris, D., and Nebel, J.-C. (2010a). Automatic configuration of spectral dimensionality reduction methods. *Pattern Recognition Letters*, 31.

[Lewandowski et al., 2010b] Lewandowski, M., Makris, D., and Nebel, J.-C. (2010b). View and style-independent action manifolds for human activity recognition. *Proceedings of the 11th European Conference on Computer Vision*, 6316.

[Lewandowski et al., 2011] Lewandowski, M., Makris, D., and Nebel, J.-C. (2011). Probabilistic feature extraction from multivariate time series using spatio-temporal constraints. *Proceedings of the 15th Pacific-Asia Conference on Knowledge Discovery and Data Mining*.

[Lewandowski et al., 2010c] Lewandowski, M., Martinez-del Rincon, J., Makris, D., and Nebel, J.-C. (2010c). Temporal extension of laplacian eigenmaps for unsupervised dimensionality reduction of time series. *Proceedings of the 20th International Conference on Pattern Recognition*.

[Li et al., 2006] Li, C., Khan, L., and Prabhakaran, B. (2006). Real-time classification of variable length multi-attribute motions. *Knowledge and Information Systems*, 10:163–183.

[Li et al., 2007a] Li, C., Khan, L., and Prabhakaran, B. (2007a). Feature selection for classification of variable length multiattribute motions. *Multimedia Data Mining and Knowledge Discovery*, pages 116–137.

[Li et al., 2007b] Li, C.-G., Guo, J., and Nie, X. (2007b). Intrinsic dimensionality estimation with neighborhood convex hull. *Proceedings of the International Conference on Computational Intelligence and Security*, pages 75–79.

[Li et al., 2007c] Li, R., Tian, T., and Sclaroff, S. (2007c). Simultaneous learning of nonlinear manifold and dynamical models for high-dimensional time series. *Proceedings of the 11th International Conference on Computer Vision*, pages 1–8.

[Li et al., 2010] Li, R., Tian, T.-P., Sclaroff, S., and Yang, M.-H. (2010). 3d human motion tracking with a coordinated mixture of factor analyzers. *International Journal of Computer Vision*, 87(1-2):170–190.

[Lin et al., 2006] Lin, R., Liu, C., Yang, M., Ahuja, N., and Levinson, S. (2006). Learning nonlinear manifolds from time series. *Proceedings of the 9th European Conference on Computer Vision*, 2:239–250.

[Lin et al., 2008] Lin, S., Lee, Z., Chen, S., and Tseng, T. (2008). Parameter determination of support vector machine and feature selection using simulated annealing approach. *Applied soft computing*, 8(4):1505–1512.

[Liu and Setiono, 1996] Liu, H. and Setiono, R. (1996). A probabilistic approach to feature selection - a filter solution. *Proceedings of the 13th International Conference on Machine Learning*, pages 319–327.

[Liu and Yu, 2005]Liu, H. and Yu, L. (2005). Toward integrating feature selection algorithms for classification and clustering. *IEEE Transactions on Knowledge and Data Engineering*, 17:491–502.

[Liu et al., 2008] Liu, J., Ali, S., and Shah, M. (2008). Recognizing human actions using multiple features. *Proceedings of the International Conference on Computer Vision and Pattern Recognition*.

[Liu and Kavakli, 2010] Liu, J. and Kavakli, M. (2010). Hand gesture recognition based on segmented singular value decomposition. *Knowledge-Based and Intelligent Information and Engineering Systems*, pages 214–223.

[Liu et al., 2009] Liu, J., Luo, J., and Shah, M. (2009). Recognizing realistic actions from videos "in the wild". *International Conference on Computer Vision and Pattern Recognition*.

[Liu and Shah, 2008] Liu, J. and Shah, M. (2008). Learning human actions via information maximization. *Proceedings of the International Conference on Computer Vision and Pattern Recognition*.

[Liul et al., 1998] Liul, H., Motoda, H., and Dash, M. (1998). A monotonic measure for optimal feature selection. *European Conference on Machine Learning*, pages 101–106.

[Lloyd, 1982] Lloyd, S. (1982). Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2):129–137.

[Lu and Little, 2006] Lu, W.-L. and Little, J. (2006). Simultaneous tracking and action recognition using the pca-hog descriptor. *Proceedings of the 3rd Canadian Conference on Computer and Robot Vision*, page 6.

[Lu et al., 2007] Lu, Z., Carreira-Perpinan, M., and Sminchisescu, C. (2007). People tracking with the laplacian eigenmaps latent variable model. *Advances in Neural Information Processing Systems*.

[Lucas and Kanade, 1981] Lucas, B. and Kanade, T. (1981). An iterative image registration technique with an application to stereo vision. *Proceedings of the 7th International Joint Conference on Artificial intelligence*, pages 674–679.

[Lv and Nevatia, 2007] Lv, F. and Nevatia, R. (2007). Single view human action recognition using key pose matching and viterbi path searching. *International Conference on Computer Vision and Pattern Recognition*, pages 1–8.

[MacKay, 1995] MacKay, D. (1995). Bayesian neural networks and density networks. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 354(1):73–80.

[MacKay and Ghahramani, 2005] MacKay, D. and Ghahramani, Z. (2005). Comments on 'maximum likelihood estimation of intrinsic dimension' by e. levina and p. bickel,. Technical report, University College London. http://www.inference.phy.cam.ac.uk/mackay/dimension/.

[Magnus and Neudecker, 1999] Magnus, J. and Neudecker, H. (1999). *Matrix differential calculus with applications in statistics and econometrics*. John Wiley & Sons, 2nd edition.

[Marquardt, 1963] Marquardt, D. (1963). An algorithm for least-squares estimation of nonlinear parameters. *SIAM Journal on Applied Mathematics*, 11(2):431–441.

[Martinez-Contreras et al., 2009] Martinez-Contreras, F., Orrite-Urunuela, C., Herrero-Jaraba, E., Ragheb, H., and Velastin, S. (2009). Recognizing human actions using silhouette-based hmm. *Proceedings of the 6th International Conference on Advanced Video and Signal Based Surveillance*, pages 43–48.

[Martinez-del Rincon et al., 2011] Martinez-del Rincon, J., Lewandowski, M., Makris, D., and Nebel, J.-C. (2011). Style temporal laplacian eigenmaps for learning stylistic variations. *Submitted to 23rd International Conference on Computer Vision and Pattern Recognition.*

[Masoud and Papanikolopoulos, 2003] Masoud, O. and Papanikolopoulos, N. (2003). A method for human action recognition. *Image and Vision Computing*, 21(8):729–743.

[Matikainen et al., 2010] Matikainen, P., Hebert, M., and Sukthankar, R. (2010). Representing pairwise spatial and temporal relations for action recognition. *Proceedings of the 11th European Conference on Computer Vision*.

[McLachlan and Krishnan, 2008] McLachlan, G. and Krishnan, T. (2008). *The EM Algorithm and Extensions*. Wiley Series in Probability and Statistics. Wiley-Interscience, 2 edition.

[Meiri and Zahavi, 2006] Meiri, R. and Zahavi, J. (2006). Using simulated annealing to optimize the feature selection problem in marketing applications. *European Journal of Operational Research*, 171(3):842–858.

[Mekuz and Tsotsos, 2006] Mekuz, N. and Tsotsos, J. (2006). Parameterless isomap with adaptive neighborhood selection. *Proceedings of the 28th Annual Symposium of the German Association for Pattern Recognition*, pages 364–373.

[Menache, 1999] Menache, A. (1999). *Understanding Motion Capture for Computer Animation and Video Games*. Morgan Kaufmann Publishers Inc.

[Meng and Pears, 2009] Meng, H. and Pears, N. (2009). Descriptive temporal template features for visual motion recognition. *Pattern Recognition Letters*, 30(12):1049–1058.

[Mercer, 1909] Mercer, J. (1909). Functions of positive and negative type, and their connection with the theory of integral equations. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, pages 415–446.

[Metropolis et al., 1953] Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., and Teller, E. (1953). Simulated annealing. *Journal of Chemical Physics*, 21:1087.

[Micchelli, 1986] Micchelli, C. (1986). Interpolation of scattered data: Distance matrices and conditionally positive definite functions. *Constructive Approximation*, 2(1):11–22.

[Mikic et al., 2003] Mikic, I., Trivedi, M., Hunter, E., and Cosman, P. (2003). Human body model acquisition and tracking using voxel data. *Ínternational Journal of Computer Vision*, 53(3):199–223.

[Mizrahi and Sullivan, 1990] Mizrahi, A. and Sullivan, M. (1990). *Calculus and Analytic Geometry*. Wadsworth Publishing Company, 3 edition.

[Moeslund et al., 2006] Moeslund, T., Hilton, A., and Krüger, V. (2006). A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 104(2):90–126.

[Möller, 1993] Möller, M. (1993). A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks*, 6(4):525—533.

[Moon and Pavlovic, 2008] Moon, K. and Pavlovic, V. (2008). 3d human motion tracking using dynamic probabilistic latent semantic analysis. *Proceedings of the 5th Canadian Conference on Computer and Robot Vision*, pages 155–162.

[Mori and Malik, 2006] Mori, G. and Malik, J. (2006). Recovering 3d human body configurations using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1052–1062.

[Moutzouris et al., 2011] Moutzouris, A., Martinez-del Rincon, J., Lewandowski, M., Nebel, J.-C., and Makris, D. (2011). Human pose tracking in low dimensional spaces enhanced by limb correction. *Submitted to 18th International Conference on Image Processing*.

[MuHAVi, 2010] MuHAVi (2010). Kingston university multicamera human action video data (http://dipersec.king.ac.uk/muhavi-mas/) [last accessed on 17/10/2010].

[Muybridge, 1901] Muybridge, E. (1901). *The Human Figure In Motion*. Chapman and Hall.

[Myronenko et al., 2007] Myronenko, A., Song, X., and Carreira-Perpinán, M. (2007). Non-rigid point set registration: Coherent point drift. *Advances in Neural Information Processing Systems*, 19:1009.

[Nagel and Enkelmann, 1986] Nagel, H. and Enkelmann, W. (1986). An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(5):565–593.

[Nakariyakul and Casasent, 2009] Nakariyakul, S. and Casasent, D. (2009). An improvement on floating search algorithms for feature subset selection. *Pattern Recognition*, 42(9):1932–1940.

[Narendra and Fukunaga, 1977] Narendra, P. and Fukunaga, K. (1977). A branch and bound algorithm for feature subset selection. *IEEE Transactions on Computers*, C-269:917–922.

[Natarajan and Nevatia, 2008] Natarajan, P. and Nevatia, R. (2008). View and scale invariant action recognition using multiview shape-flow models. *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 1–8.

[Nene et al., 1996] Nene, S., Nayar, S., and Murase, H. (1996). Columbia object image library (coil-100). Cucs-006-96, Columbia University Technical Report.

[Niebles et al., 2008] Niebles, J., Wang, H., and Fei-Fei, L. (2008). Unsupervised learning of human action categories using spatial-temporal words. *International Journal of Computer Vision*, 79(3):299–318.

[Nocedal and Wright, 2006] Nocedal, J. and Wright, S. (2006). *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering. Springer Verlag, 2 edition.

[Ogale et al., 2005] Ogale, A., Karapurkar, A., and Aloimonos, Y. (2005). View-invariant modeling and recognition of human actions using grammars. *Workshop on Dynamical Vision at International Conference on Computer Vision*, 5:115–126.

[Oh et al., 2004] Oh, I., Lee, J., and Moon, B. (2004). Hybrid genetic algorithms for feature selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(11):1424–1437.

[O'Hagan, 1992] O'Hagan, A. (1992). Some bayesian numerical analysis. *Bayesian Statistics*, 4:345–363.

[Ohbuchi et al., 2008] Ohbuchi, R., Kobayashi, J., Yamamoto, A., and Shimizu, T. (2008). Comparison of dimension reduction methods for database-adaptive 3d model retrieval. *Adaptive Multimedial Retrieval: Retrieval, User, and Semantics*, pages 196–210.

[Parameswaran, 2004]Parameswaran, V. (2004). *View-Invariant in Visual Human Motion Analysis*. Phd thesis, University of Maryland, College Park, USA.

[Parameswaran and Chellappa, 2006] Parameswaran, V. and Chellappa, R. (2006). View invariance for human action recognition. *International Journal of Computer Vision*, 66(1):83–101.

[Pavlovic et al., 2001] Pavlovic, V., Rehg, J., and MacCormick, J. (2001). Learning switching linear models of human motion. *Advances in Neural Information Processing Systems*, pages 981–987.

[Pehlivan and Duygulu, 2010] Pehlivan, S. and Duygulu, P. (2010). A new pose-based representation for recognizing actions from multiple cameras. *Computer Vision and Image Understanding*.

[Peng et al., 2005] Peng, H., Long, F., and Ding, C. (2005). Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1226–1238.

[Peng et al., 2009] Peng, X., Zou, B., Chen, S., and Luo, P. (2009). Reconstruction of 3d human motion pose from uncalibrated monocular video sequences. *Journal of Information and Systems Sciences*, 5(3-4):503–5155.

[Penrose, 1955] Penrose, R. (1955). A generalized inverse for matrices. *Mathematical Proceedings of the Cambridge Philosophical Society*, 51:406–413.

[Peursum et al., 2007] Peursum, P., Venkatesh, S., and West, G. (2007). Tracking-as-recognition for articulated full-body human motion analysis. *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 1–8.

[Poggio and Girosi, 1990] Poggio, T. and Girosi, F. (1990). Networks for approximation and learning. *Proceedings of the IEEE*, 78(9):1481–1497.

[Poppe, 2007a] Poppe, R. (2007a). Evaluating example-based pose estimation: Experiments on the humaneva sets. *Computer Vision and Pattern Recognition workshop on Evaluation of Articulated Human Motion and Pose Estimation*.

[Poppe, 2007b] Poppe, R. (2007b). Vision-based human motion analysis: An overview. *Computer Vision and Image Understanding*, 108:4–18.

[Poppe, 2010] Poppe, R. (2010). A survey on vision-based human action recognition. *Image and Vision Computing*, 28(6):976–990.

[Powell, 1987] Powell, M. J. D. (1987). Radial basis functions for multivariable interpolation: A review. *Algorithms for Approximation*, pages 143–167.

[Pudil et al., 1994] Pudil, P., Novovicová, J., and Kittler, J. (1994). Floating search methods in feature selection. *Pattern Recognition Letters*, 15(11):1119–1125.

[Quinlan, 1993] Quinlan, J. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc.

[Quirion et al., 2008] Quirion, S., Duchesne, C., Laurendeau, D., and Marchand, M. (2008). Comparing gplvm approaches for dimensionality reduction in character animation. *Journal of WSCG*, 16(1-3):41–48.

[Rabiner, 1989] Rabiner, L. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

[Rabiner and Juang, 1993] Rabiner, L. and Juang, B.-H. (1993). *Fundamentals of Speech Recognition*. Prentice-Hall, Inc.

[Ragheb et al., 2008] Ragheb, H., Velastin, S., Remagnino, P., and Ellis, T. (2008). Vihasi: Virtual human action silhouette data for the performance evaluation of silhouette-based action recognition methods. *Proceedings of the 2nd ACM/IEEE International Conference on Distributed Smart Cameras*, pages 1–10.

[Ramos et al., 2007] Ramos, A., Socas-Navarro, H., Ariste, A., and Gonzlez, M. (2007). The intrinsic dimensionality of spectropolarimetric data. *The Astrophysical Journal*, 660(2).

[Rasmussen and Williams, 2006] Rasmussen, C. and Williams, C. (2006). *Gaussian Processes for Machine Learning*. Adaptive Computation and Machine Learning. The MIT Press.

[Raymer et al., 2000] Raymer, M., Punch, W., Goodman, E., Kuhn, L., and Jain, A. (2000). Dimensionality reduction using genetic algorithms. *IEEE Transactions on Evolutionary Computation*, 4(2):164–171.

[Reddy et al., 2010] Reddy, K., Liu, J., and Shah, M. (2010). Incremental action recognition using feature-tree. *Proceedings of the 12th International Conference on Computer Vision*, pages 1010–1017.

[Remondino and Roditakis, 2003] Remondino, F. and Roditakis, A. (2003). 3d reconstruction of human skeleton from single images or monocular video sequences. *Proceedings of the 25th Pattern Recognition Symposium*, pages 100–107.

[Richard and Kyle, 2009] Richard, S. and Kyle, P. (2009). Viewpoint manifolds for action recognition. *Journal on Image and Video Processing*.

[Rodriguez et al., 2008] Rodriguez, M., Ahmed, J., and Shah, M. (2008). Action mach a spatio-temporal maximum average correlation height filter for action recognition. *International Conference on Computer Vision and Pattern Recognition*, pages 1–8.

[Rodriguez-Lujan et al., 2010] Rodriguez-Lujan, I., Huerta, R., Elkan, C., and Santa Cruz, C. (2010). Quadratic programming feature selection. *Journal of Machine Learning Research*, 11:1491–1516.

[Romeijn and Smith, 1994] Romeijn, H. and Smith, R. (1994). Simulated annealing for constrained global optimization. *Journal of Global Optimization*, 5(2):101–126.

[Rose and Christina, 2005] Rose, D. and Christina, R. (2005). *A Multilevel Approach to the Study of Motor Control and Learning*. Benjamin Cummings.

[Rote, 1991] Rote, G. (1991). Computing the minimum hausdorff distance between two point sets on a line under translation. *Information Processing Letters*, 38(3):123–127.

[Roth et al., 2009] Roth, P., Mauthner, T., Khan, I., and Bischof, H. (2009). Efficient human action recognition by cascaded linear classification. *1st IEEE Workshop on Video-Oriented Object and Event Classification*.

[Roweis and Saul, 2000] Roweis, S. and Saul, L. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326.

[Roweis et al., 2002] Roweis, S., Saul, L., and Hinton, G. (2002). Global coordination of local linear models. *Advances in Neural Information Processing Systems*, (14):889–896.

[Rozado et al., 2010] Rozado, D., Rodriguez, F., and Varona, P. (2010). Optimizing hierarchical temporal memory for multivariable time series. *Proceedings of the 20th International Conference on Artificial Neural Networks*, pages 506–518.

[Rubin and Thayer, 1982] Rubin, D. and Thayer, D. (1982). Em algorithms for ml factor analysis. *Psychometrika*, 47(1):69–76.

[Rubinstein and Kroese, 2008] Rubinstein, R. and Kroese, D. (2008). *Simulation and the Monte Carlo Method*. Wiley, 2 edition.

[Rui and Anandan, 2002] Rui, Y. and Anandan, P. (2002). Segmenting visual actions based on spatio-temporal motion patterns. *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 1:111–118.

[Rumelhart et al., 1985] Rumelhart, D., Hinton, G., and Williams, R. (1985). Learning internal representations by error propagation. *Parallel Distributed Processing: Exploration in the Microstructure of Cognition*.

[Russell and Norvig, 2003] Russell, S. and Norvig, P. (2003). *Artificial Intelligence: A Modern Approach*. Pearson Education, 2nd edition.

[Ryoo and Aggarwal, 2009] Ryoo, M. S. and Aggarwal, J. K. (2009). Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities. *Proceedings of the 12th International Conference on Computer Vision*.

[Saeys et al., 2007]Saeys, Y., Inza, I., and Larrañaga, P. (2007). A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23(19):2507–2517.

[Sakoe and Chiba, 1990] Sakoe, H. and Chiba, S. (1990). Dynamic programming algorithm optimization for spoken word recognition. *Readings in Speech Recognition*, pages 159–165.

[Salzmann et al., 2008] Salzmann, M., Urtasun, R., and Fua, P. (2008). Local deformation models for monocular 3d shape recovery. *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 1–8.

[Samaria and Harter, 1994] Samaria, F. and Harter, A. (1994). Parameterisation of a stochastic model for human face identification (http://www.cs.toronto.edu/roweis/data.html) [last accessed on 05/10/2010]. Workshop on Applications of Computer Vision.

[Samko et al., 2006] Samko, O., Marshall, A., and Rosin, P. (2006). Selection of the optimal parameter value for the isomap algorithm. *Pattern Recognition Letters*, 27(9):968–979.

[Sammon, 1969] Sammon, J. (1969). A nonlinear mapping for data structure analysis. *IEEE Transactions on Computers*, 18(5):401–409.

[Satkin and Hebert, 2010] Satkin, S. and Hebert, M. (2010). Modeling the temporal extent of actions. *Proceedings of the 11th European Conference on Computer Vision*, pages 536–548.

[Schapire, 1999] Schapire, R. (1999). A brief introduction to boosting. *Proceedings of the 16th International Joint Conference on Artificial Intelligence*.

[Schindler and van Gool, 2008] Schindler, K. and van Gool, L. (2008). Action snippets: How many frames does human action recognition require? *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 1–8.

[Schölkopf and Smola, 2002] Schölkopf, B. and Smola, A. (2002). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press.

[Schölkopf et al., 1997] Schölkopf, B., Smola, A., and Müller, K. (1997). Kernel principal component analysis. *Artificial Neural Networks*, pages 583–588.

[Schuldt et al., 2004] Schuldt, C., Laptev, I., and Caputo, B. (2004). Recognizing human actions: A local svm approach. *Proceedings of the 17th International Conference on Pattern Recognition*, 3:32–36.

[Scott and Thompson, 1983] Scott, D. and Thompson, J. (1983). Probability density estimation in higher dimensions. *Computer Science and Statistics: Proceedings of the Fifteenth Symposium on the Interface*, 528:173–179.

[Seber, 2004] Seber, G. (2004). *Multivariate Observations*. Wiley.

[Senin, 2008] Senin, P. (2008). Dynamic time warping algorithm review. Technical report, University of Hawaii Technical Report.

[Seo et al., 2009] Seo, S., Kang, J., and R.K., H. (2009). Multivariable stream data classification using motifs and their temporal relations. *Journal Information Sciences*, 179:3489–3504.

[Sha and Saul, 2005] Sha, F. and Saul, L. (2005). Analysis and extension of spectral methods for nonlinear dimensionality reduction. *Proceedings of the 22nd International Conference on Machine learning*, pages 784–791.

[Shakhnarovich et al., 2003] Shakhnarovich, G., Viola, P., and Darrell, T. (2003). Fast pose estimation with parameter-sensitive hashing. *Proceedings of the 9th International Conference on Computer Vision*, 2.

[Sheikh et al., 2005] Sheikh, Y., Sheikh, M., and Shah, M. (2005). Exploring the space of a human action. *Proceedings of the 10th International Conference on Computer Vision*, 1:144–149.

[Shi et al., 2005] Shi, L., He, P., Liu, B., Fu, K., and Wu, Q. (2005). A robust generalization of isomap for new data. *Proceedings of the International Conference on Machine Learning and Cybernetics*, pages 1707–1712.

[Shon et al., 2006] Shon, A., Grochow, K., Hertzmann, A., and Rao, R. (2006). Learning shared latent structure for image synthesis and robotic imitation. *Advances in Neural Information Processing Systems*, 18:1233.

[Sibson, 1979] Sibson, R. (1979). Studies in the robustness of multidimensional scaling: Perturbational analysis of classical scaling. *Journal of the Royal Statistical Society*, pages 217–229.

[Siddiqi et al., 2007] Siddiqi, S., Gordon, G., and Moore, A. (2007). Fast state discovery for hmm model selection and learning. *Proceedings of the 11th International Conference on Artificial Intelligence and Statistics*.

[Sidenbladh et al., 2000] Sidenbladh, H., Black, M., and Fleet, D. (2000). Stochastic tracking of 3d human figures using 2d image motion. *Proceedings of the 6th European Conference on Computer Vision*, pages 702–718.

[Sidenbladh et al., 2002] Sidenbladh, H., Black, M., and Sigal, L. (2002). Implicit probabilistic models of human motion for synthesis and tracking. *Proceedings of the 7th European Conference on Computer Vision*, pages 784–800.

[Siedlecki and Sklansky, 1989] Siedlecki, W. and Sklansky, J. (1989). A note on genetic algorithms for large-scale feature selection. *Pattern Recognition Letters*, 10(5):335–347.

[Sigal et al., 2010] Sigal, L., Balan, A., and Black, M. (2010). Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. *International Journal of Computer Vision*, 87(1):4–27.

[Simon, 1996] Simon, H. (1996). *The Sciences of the Artificial*. The MIT Press, 3 edition.

[Simonnet and Velastin, 2010] Simonnet, D. and Velastin, S. (2010). Pedestrian detection based on adaboost algorithm with a pseudo-calibrated camera. *Proceedings of 2nd International Conference on Image Processing Theory, Tools and Applications*, pages 54–59.

[Singh-Miller et al., 2007] Singh-Miller, N., Collins, M., and Hazen, T. (2007). Dimensionality reduction for speech recognition using neighborhood components analysis. *Proceedings of Interspeech*, pages 1158–1161.

[Skalak, 1994] Skalak, D. (1994). Prototype and feature selection by sampling and random mutation hill climbing algorithms. *Proceedings of the 11th International Conference on Machine Learning*, pages 293–301.

[Sminchisescu et al., 2006] Sminchisescu, C., Kanaujia, A., and Metaxas, D. (2006). Conditional models for contextual human motion recognition. *Computer Vision and Image Understanding*, 104(2):210–220.

[Sminchisescu and Triggs, 2003] Sminchisescu, C. and Triggs, B. (2003). Kinematic jump processes for monocular 3d human tracking. *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 1:69–76.

[Sminchisescu et al., 2001] Sminchisescu, C., Triggs, B., Gravir, I., and Montbonnot, F. (2001). Covariance scaled sampling for monocular 3d body tracking. *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 1:447—454.

[Somol et al., 2004] Somol, P., Pudil, P., and Kittler, J. (2004). Fast branch & bound algorithms for optimal feature selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(7):900–912.

[Somol et al., 1999] Somol, P., Pudil, P., and Novovicová, J. (1999). Adaptive floating search methods in feature selection. *Pattern Recognition Letters*, 20(11-13):1157–1163.

[Starck and Hilton, 2005] Starck, J. and Hilton, A. (2005). Spherical matching for temporal correspondence of non-rigid surfaces. *Proceedings of the 10th International Conference on Computer Vision*, pages 15–21.

[Starner, 1995] Starner, T. (1995). Visual recognition of american sign language using hidden markov models. *Proceedings of the International Workshop on Automatic Face and Gesture Recognition*, pages 189–194.

[Stauffer and Grimson, 1999] Stauffer, C. and Grimson, W. E. L. (1999). Adaptive background mixture models for real-time tracking. *International Conference on Computer Vision and Pattern Recognition*, 2:246–252.

[Ta et al., 2010a] Ta, A., Wolf, C., Lavoué, G., and Baskurt, A. (2010a). Recognizing and localizing individual activities through graph matching. *Proceedings of the 7th International Conference on Advanced Video and Signal Based Surveillance*.

[Ta et al., 2010b] Ta, A., Wolf, C., Lavoué, G., Baskurt, A., and Jolion, J.-M. (2010b). Pairwise features for human action recognition. *Proceedings of the 20th International Conference on Pattern Recognition*.

[Takiguchi and Ariki, 2007] Takiguchi, T. and Ariki, Y. (2007). Pca-based speech enhancement for distorted speech recognition. *Journal of Multimedia*, 2(5):13.

[Taylor, 2000] Taylor, C. (2000). Reconstruction of articulated objects from point correspondences in a single uncalibrated image. *Computer Vision and Image Understanding*, 80(3):349–363.

[Teh and Roweis, 2003] Teh, Y. and Roweis, S. (2003). Automatic alignment of local representations. *Advances in Neural Information Processing Systems*, (15):865–872.

[Tenenbaum and Freeman, 2000] Tenenbaum, J. and Freeman, W. (2000). Separating style and content with bilinear models. *Neural Computation*, 12(6):1247–1283.

[Tenenbaum et al., 2000] Tenenbaum, J., Silva, V., and Langford, J. (2000). A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323.

[Thornton et al., 1998] Thornton, I., Pinto, J., and Shiffrar, M. (1998). The visual perception of human locomotion. *Cognitive Neuropsychology*, 15:535–552.

[Tian et al., 2005] Tian, T.-P., Li, R., and Sclaroff, S. (2005). Articulated pose estimation in a learned smooth space of feasible solutions. *Proceedings of the International Conference on Computer Vision and Pattern Recognition* - *Workshop*, 3:50.

[Tipping, 2000] Tipping, M. (2000). Relevance vector machine. *Advances in Neural Information Processing Systems*.

[Tipping, 2001] Tipping, M. (2001). Sparse kernel principal component analysis. *Advances in Neural Information Processing Systems*, pages 633–639.

[Tipping and Bishop, 1999a] Tipping, M. and Bishop, C. (1999a). Mixtures of probabilistic principal component analyzers. *Neural computation*, 11(2):443–482.

[Tipping and Bishop, 1999b] Tipping, M. and Bishop, C. (1999b). Probabilistic principal component analysis. *Journal of the Royal Statistical Society*, 61(3):611–622.

[Torgerson, 1952] Torgerson, W. (1952). Multidimensional scaling: I. theory and method. *Psychometrika*, 17(4):401–419.

[Tran and Sorokin, 2008] Tran, D. and Sorokin, A. (2008). Human activity recognition with metric learning. *Proceedings of the 10th European Conference on Computer Vision*, pages 548–561.

[Trunk, 1979] Trunk, G. V. (1979). A problem of dimensionality: A simple example. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 3:306–307.

[Tsai, 1987] Tsai, R. (1987). A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, 3(4):323–344.

[Turaga et al., 2008a] Turaga, P., Chellappa, R., Subrahmanian, V. S., and Udrea, O. (2008a). Machine recognition of human activities: A survey. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(11):1473–1488.

[Turaga et al., 2008b] Turaga, P., Veeraraghavan, A., and Chellappa, R. (2008b). Statistical analysis on stiefel and grassmann manifolds with applications in computer vision. *International Conference on Computer Vision and Pattern Recognition*, pages 1–8.

[Urtasun and Darrell, 2007] Urtasun, R. and Darrell, T. (2007). Discriminative gaussian process latent variable model for classification. *Proceedings of the 24th International Conference on Machine Learning*, 227:927–934.

[Urtasun et al., 2006a]Urtasun, R., Fleet, D., and Fua, P. (2006a). 3d people tracking with gaussian process dynamical models. *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 1:238–245.

[Urtasun et al., 2006b] Urtasun, R., Fleet, D., and Fua, P. (2006b). Temporal motion models for monocular and multiview 3d human body tracking. *Computer Vision and Image Understanding*, 104(2-3):157–177.

[Urtasun et al., 2008] Urtasun, R., Fleet, D., Geiger, A., Popovic, J., Darrell, T., and Lawrence, N. (2008). Topologically-constrained latent variable models. *Proceedings of the 25th International Conference on Machine Learning*, 307:1080–1087.

[Urtasun et al., 2005] Urtasun, R., Fleet, D., Hertzmann, A., and Fua, P. (2005). Priors for people tracking from small training sets. *Proceedings of the 10th International Conference on Computer Vision*, 1:403–410.

[Urtasun et al., 2007] Urtasun, R., Fleet, D., and Lawrence, N. (2007). Modeling human locomotion with topologically constrained latent variable models. *Proceedings of the 2nd Conference on Human motion: Understanding, Modeling, Capture and Animation*, pages 104–118.

[Vafaie and De Jong, 1993] Vafaie, H. and De Jong, K. (1993). Robust feature selection algorithms. *Proceedings of the 5th International Conference on Tools with Artificial Intelligence*, pages 356–363.

[van der Maaten et al., 2009]van der Maaten, L., Postma, E., and van den Herik, H. (2009). Dimensionality reduction: A comparative review. Ticc-tr-2009-005, Tilburg University Technical Report.

[Vandenberghe and Boyd, 1996] Vandenberghe, L. and Boyd, S. (1996). Semidefinite programming. *SIAM Review*, 38(1):49–95.

[Vapnik, 1998] Vapnik, V. (1998). *Statistical Learning Theory*. John Wiley and Sons.

[Vasilescu and Terzopoulos, 2003] Vasilescu, M. and Terzopoulos, D. (2003). Multilinear subspace analysis of image ensembles. *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 2.

[Vezzani et al., 2010] Vezzani, R., Baltieri, D., and Cucchiara, R. (2010). Hmm based action recognition with projection histogram features. *Proceedings of the 20th International Conference on Pattern Recognition: Contest on Semantic Description of Human Activities*.

[Vinci, 1492] Vinci, L. D. (1492). *Description of ’Vitruvian Man’*.

[Wang and Mahadevan, 2008] Wang, C. and Mahadevan, S. (2008). Manifold alignment using procrustes analysis. *Proceedings of the 25th International Conference on Machine Learning*, pages 1120–1127.

[Wang et al., 2009] Wang, H., Ullah, M., Kläser, A., Laptev, I., and Schmid, C. (2009). Evaluation of local spatio-temporal features for action recognition. *Proceedings of the 20th British Machine Vision Conference*.

[Wang et al., 2006] Wang, J., Fleet, D., and Hertzmann, A. (2006). Gaussian process dynamical models. *Advances in Neural Information Processing Systems*, 18:1441–1448.

[Wang et al., 2008] Wang, J., Fleet, D., and Hertzmann, A. (2008). Gaussian process dynamical models for human motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):283–298.

[Wang et al., 2005] Wang, J., Zhang, Z., and Zha, H. (2005). Adaptive manifold learning. *Advances in Neural Information Processing Systems*, 17:1473–1480.

[Wang and Suter, 2007a] Wang, L. and Suter, D. (2007a). Learning and matching of dynamic shape manifolds for human action recognition. *IEEE Transactions on Image Processing*, 16(6):1646–1661.

[Wang and Suter, 2007b] Wang, L. and Suter, D. (2007b). Recognizing human activities from silhouettes: Motion subspace and factorial discriminative graphical model. *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 1–8.

[Wang and Suter, 2008] Wang, L. and Suter, D. (2008). Visual learning and recognition of sequential data manifolds with applications to human movement analysis. *Computer Vision and Image Understanding*, 110(2):153–172.

[Wang and Rehg, 2006] Wang, P. and Rehg, J. (2006). A modular approach to the analysis and evaluation of particle filters for figure tracking. *International Conference on Computer Vision and Pattern Recognition*, 1:790–797.

[Wang and Li, 2009] Wang, Q. and Li, J. (2009). Combining local and global information for nonlinear dimensionality reduction. *Neurocomputing*, 72(10-12).

[Wang et al., 2010] Wang, X., Gao, X., Yuan, Y., Tao, D., and Li, J. (2010). Semi-supervised gaussian process latent variable model with pairwise constraints. *Neurocomputing*, 73(10-12).

[Warshall, 1962] Warshall, S. (1962). A theorem on boolean matrices. *Journal of the ACM*, 9(1):11–12.

[Weber et al., 1998] Weber, R., Schek, H.-J., and Blott, S. (1998). A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces. *Proceedings of the 24rd International Conference on Very Large Data Bases*, pages 194–205.

[Wei et al., 2008] Wei, J., Peng, H., Lin, Y., Huang, Z., and Wang, J. (2008). Adaptive neighborhood selection for manifold learning. *Proceedings of the 17th International Conference on Machine Learning and Cybernetics*, 1:380–384.

[Weinberger and Saul, 2005] Weinberger, K. and Saul, L. (2005). Unsupervised learning of image manifolds by semidefinite programming. *International Journal of Computer Vision*, 70(1):77–90.

[Weinland et al., 2007] Weinland, D., Boyer, E., and Ronfard, R. (2007). Action recognition from arbitrary views using 3d exemplars. *Proceedings of the 11th International Conference on Computer Vision*, 5(7):8.

[Weinland et al., 2006a] Weinland, D., Ronfard, R., and Boyer, E. (2006a). Automatic discovery of action taxonomies from multiple views. *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 2:1639–1645.

[Weinland et al., 2006b] Weinland, D., Ronfard, R., and Boyer, E. (2006b). Free viewpoint action recognition using motion history volumes. *Computer Vision and Image Understanding*, 104(2-3):249–257.

[Weinland et al., 2010a] Weinland, D., Ronfard, R., and Boyer, E. (2010a). A survey of vision-based methods for action representation, segmentation and recognition. *Computer Vision and Image Understanding*.

[Weinland et al., 2010b] Weinland, D., Özuysal, M., and Fua, P. (2010b). Making action recognition robust to occlusions and viewpoint changes. *Proceedings of the 11th European Conference on Computer Vision*.

[Wen et al., 2007] Wen, G., Jiang, L., and Shadbolt, N. (2007). Using graph algebra to optimize neighborhood for isometric mapping. *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, pages 2398–2403.

[Wen et al., 2008] Wen, G., Jiang, L., and Wen, J. (2008). Using locally estimated geodesic distance to optimize neighborhood graph for isometric data embedding. *Pattern Recognition*, 41(7).

[Weng and Shen, 2008a] Weng, X. and Shen, J. (2008a). Classification of multivariate time series using locality preserving projections. *Journal Knowledge-Based Systems*, 21:581–587.

[Weng and Shen, 2008b] Weng, X. and Shen, J. (2008b). Classification of multivariate time series using two-dimensional singular value decomposition. *Knowledge-Based Systems*, 21(7):535–539.

[Whitney, 1971] Whitney, A. (1971). A direct method of nonparametric measurement selection. *IEEE Transactions on Computers*, 20(9):1100–1103.

[Willems et al., 2008] Willems, G., Tuytelaars, T., and Gool, L. (2008). An efficient dense and scale-invariant spatio-temporal interest point detector. *Proceedings of the 10th European Conference on Computer Vision*, pages 650–663.

[Williams, 1998] Williams, C. (1998). Prediction with gaussian processes: From linear regression to linear prediction and beyond. *Learning in Graphical Models*, 89:599–621.

[Williams, 2002] Williams, C. (2002). On a connection between kernel pca and metric multidimensional scaling. *Machine Learning*, 46(1):11–19.

[Wu et al., 2009] Wu, X., Liang, W., and Jia, Y. (2009). Tracking articulated objects by learning intrinsic structure of motion. *Pattern Recognition Letters*, 30(3):267–274.

[Xu et al., 1995] Xu, L., Jordan, M., and Hinton, G. (1995). An alternative model for mixtures of experts. *Advances in Neural Information Processing Systems*, pages 633–640.

[Xu et al., 1993] Xu, L., Krzyzak, A., and Oja, E. (1993). Rival penalized competitive learning for clustering analysis, rbfnet, and curve detection. *IEEE Transactions on Neural Networks*, 4(4):636–649.

[Yan et al., 2008] Yan, P., Khan, S., and Shah, M. (2008). Learning 4d action feature models for arbitrary view action recognition. *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 12.

[Yang and Honavar, 1998] Yang, J. and Honavar, V. (1998). Feature subset selection using a genetic algorithm. *IEEE Intelligent Systems*, 13(2):44–49.

[Yang and Shahabi, 2007] Yang, K. and Shahabi, C. (2007). An efficient k nearest neighbor search for multivariate time series. *Journal Information and Computation*, 205:65–98.

[Yang, 2003] Yang, M. (2003). Discriminant isometric mapping for face recognition. *Lecture Notes in Computer Science*, 2626:470–480.

[Yang and Pedersen, 1997] Yang, Y. and Pedersen, J. (1997). A comparative study on feature selection in text categorization. *Proceedings of the 14th International Conference on Machine Learning*, pages 412–420.

[Yeffet and Wolf, 2009] Yeffet, L. and Wolf, L. (2009). Local trinary patterns for human action recognition. *Proceedings of the 12th International Conference on Computer Vision*.

[Yi et al., 1998]Yi, B.-K., Jagadish, H., and Faloutsos, C. (1998). Efficient retrieval of similar time sequences under time warping. *Proceedings of the 14th International Conference on Data Engineering*, pages 201–208.

[Yilmaz and Shah, 2005] Yilmaz, A. and Shah, M. (2005). Recognizing human actions in videos acquired by uncalibrated moving cameras. *Proceedings of the 10th International Conference on Computer Vision*, 1:150–157.

[Yin et al., 2008a] Yin, F., Makris, D., and Velastin, S. (2008a). Time efficient ghost removal for motion detection in visual surveillance systems. *Electronics Letters*, 44:1351–1353.

[Yin et al., 2008b] Yin, J., Hu, D., and Zhou, Z. (2008b). Growing locally linear embedding for manifold learning. *Journal of Pattern Recognition Research*, 2(1):1–16.

[Yu and Yuan, 1993] Yu, B. and Yuan, B. (1993). A more efficient branch and bound algorithm for feature selection. *Pattern Recognition*, 26(6):883–889.

[Yu and Liu, 2003]Yu, L. and Liu, H. (2003). Feature selection for high-dimensional data: A fast correlation-based filter solution. *Proceedings of the 20th International Conference on Machine Learning*, 20(2):856.

[Zatsiorsky, 2002] Zatsiorsky, V. (2002). *Kinetics of Human Motion*. Human Kinetics.

[Zhan et al., 2009a] Zhan, Y., Yin, J., Liu, X., and Zhang, G. (2009a). Adaptive neighborhood select based on local linearity for nonlinear dimensionality reduction. *Proceedings of the 4th International Symposium on Advances in Computation and Intelligence*, pages 337–348.

[Zhan et al., 2009b] Zhan, Y., Yin, J., and Long, J. (2009b). Dynamic neighborhood selection for nonlinear dimensionality reduction. *Proceedings of the 6th International Conference on Modeling Decisions for Artificial Intelligence*, pages 327–337.

[Zhang and Gong, 2010] Zhang, J. and Gong, S. (2010). Action categorization with modified hidden conditional random field. *Pattern Recognition*, 43(1):197–203.

[Zhang et al., 2010] Zhang, K., Sch\"olkopf, B., and Janzing, D. (2010). Invariant gaussian process latent variable models and application in causal discovery. *Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence*, pages 1–8.

[Zhang et al., 2008] Zhang, Y., Ding, C., and Li, T. (2008). Gene selection algorithm by combining relieff and mrmr. *BMC Genomics*, 9(Suppl 2).

[Zhang and Wang, 2007] Zhang, Z. and Wang, J. (2007). Mlle: Modified locally linear embedding using multiple weights. *Advances in Neural Information Processing Systems*, 19:1593–1600.

[Zhang and Zha, 2005] Zhang, Z. and Zha, H. (2005). Principal manifolds and nonlinear dimensionality reduction via tangent space alignment. *SIAM Journal on Scientific Computing*, 26(1).

[Zhao et al., 2005] Zhao, J., Li, L., and Keong, K. (2005). 3d posture reconstruction and human animation from 2d feature points. *Computer Graphics Forum*, 24(4):759–771.

[Zhao and Elgammal, 2008] Zhao, Z. and Elgammal, A. (2008). Information theoretic key frame selection for action recognition. *Proceedings of the 19th British Machine Vision Conference*, pages 1095–1104.

[Zheng et al., 2008] Zheng, F., Chen, N., and Li, L. (2008). Semi-supervised laplacian eigenmaps for dimensionality reduction. *Wavelet Analysis and Pattern Recognition*, 2:843–849.

[Zheng, 2008] Zheng, Y. (2008). Manifold learning algorithms and their mathematical foundations. Cps-234, Duke University Technical Report.

[Zhou et al., 2003] Zhou, D., Weston, J., Gretton, A., Bousquet, O., and Scholkopf, B. (2003). Ranking on data manifolds. *Advances in Neural Information Processing Systems*.