Practical Generation of Video Textures using the Auto-Regressive Process

Neill Campbell, Colin Dalton, David Gibson and Barry Thomas
Department of Computer Science
University of Bristol
Bristol, BS8 1UB
Neill.Campbell@bristol.ac.uk

Abstract

Recently, there have been several attempts at creating 'video textures', that is, synthesising new (potentially infinitely long) video clips based on existing ones. One way to do this is to transform each frame of the video into an eigenspace using Principal Components Analysis so that the original sequence can be viewed as a signature through this low-dimensional space. A new sequence can be generated by moving through this space and creating 'similar' signatures. These signatures may be derived using an auto-regressive process. Such an auto-regressive process assumes that the signature has Gaussian statistics. For many sequences this assumption is valid, however, some sequences are strongly non-linearly correlated, in which case their statistical properties are non-Gaussian. We show two methods by which such non-linearities may be overcome. The first is by modelling the non-linearity automatically using a spline, and the second using a combined appearance model. New sequences created using these approaches can contain images never present in the original sequence and are very convincing.

1 Introduction

Recently, there have been several attempts at creating 'video textures', that is, synthesising new video clips based on existing ones. Such a technique has great appeal since, once a short video is obtained, new sequences (potentially infinitely long) can be synthesised based upon it. Animators are interested in this, since many of the background shots in computer-generated movies are difficult and time-consuming to produce. It is our belief that in the future many of these shots will be created automatically. It is not just film making that requires such technology. Virtual environments and computer games all require large numbers of actors and special effects, all behaving in believable ways but not necessarily generated using individual polygonal models. Using a flame sequence to act as a light source to provide flickering light can lead to a very unconvincing look-and-feel, especially if a simple loop of video is used and the user recognises this loop.

It is not enough to simply generate new frames of a video in some random manner, the new sequence must maintain the feeling and impression of the original. In [12] each frame of the original video is projected into an eigenspace using Principal Components

BMVC 2002 doi:10.5244/C.16.41

Analysis (PCA) so that the original sequence can be viewed as a signature through this low-dimensional space. A new sequence can be generated by moving through this space and creating 'similar' signatures. The creation of similar signatures is achieved using an auto-regressive process (ARP). Such an auto-regressive process assumes that the signature has Gaussian statistics. For many sequences this assumption is valid, however, some sequences are strongly non-linearly correlated, in which case their statistical properties are non-Gaussian. We show two methods by which such non-linearities may be overcome. The first is by modelling the non-linearity automatically using a spline, and the second using a joint shape and texture model. New sequences created using these approaches can contain images never present in the original sequence and are very convincing.

In section 2 we review previous work on creating video textures and also the auto-regressive process. In Section 3 we present the basic algorithm and demonstrate its use on a campfire sequence. In Section 4 we show how additional advanced techniques are required for other sequences containing non-linear characteristics that violate the Gaussian distribution assumption of the ARP. We remap the signatures, in the case of a horse gait using a spline-fitting technique, and in the case of the laughing man use a joint model of shape and appearance.

2 Previous Work

Schodl *et al.* [3] showed new video clips by carefully choosing sub-loops of an original video sequence that could be replayed. The work computed 'transition points', frames in the original sequence that were similar to others elsewhere in the sequence. The difficulty here is to select transitions that do not end up at 'dead-ends', that is, places in the sequence from which there are no graceful exits. The results were extremely effective, but could only replay already existing frames, and would struggle with a sequence which has no similar frames that are well spaced temporally.

Fitzgibbon [12] extended this work by creating video textures by projecting the images into a low-dimensional eigenspace, and modelling them using a moving average ARP. The approach was used in the context of registering images, and demonstrated on nowhere-static scenes. Some of the initial eigenvector responses (that represent non-periodic motions, such as pans) are removed manually. Reissell and Pai [15] showed a similar technique for graphical models, where auto-regressive models are used to model candles and leaves being moved by air currents. Inputs such as wind speed are fed as extra inputs into the extended AR model.

In Gibson *et al.* [2], a computer model was animated using information extracted from an image sequence. In the work, a small section of an image sequence was manually labelled by the user, and a feed-forwards neural network used to make the non-linear associations between the responses of each frame of the sequence and what the user required the computer model to do. In this manner, a few frames of a video where labelled with information such as the position of an actors ear-lobe, and then the script to drive a model was derived automatically for the entire sequence.

PCA has also been used in a similar manner to that proposed here by Devin and Hogg [5], in which a synthetic talking head is generated using a model of joint behaviour. The generative response of the head to a users interaction is created using a Hidden Markov Model and a leaky neural network.

As well as principal components analysis our work uses the auto-regressive model (see Ljung [8]), in a form proposed by Blake and Isard [1]. Their work involved using the auto-regressive model to solve the problem of predicting where objects would move to next, in order to solve problems of tracking in computer vision.

An auto-regressive process is able to model a pattern of points in a particular space having a temporal component. Each point, at time t_k, is represented by $X(t_k)$. The mean of the points is denoted as $\bar{X}$. A second-order ARP for a series of points in a d-dimensional space may be modelled by :

$$\chi(t_k) - \bar{\chi} = A(\chi(t_{k-1}) - \bar{\chi}) + Bw_k \tag{1}$$

where w_k is a random number sampled from a zero mean, unit variance Gaussian distribution and the last two time-steps are used :

$$\chi(t_k) = \begin{pmatrix} X(t_{k-1}) \\ X(t_k) \end{pmatrix} \tag{2}$$

and

$$\bar{\chi} = \begin{pmatrix} \bar{X} \\ \bar{X} \end{pmatrix} \tag{3}$$

The unknown parameters that need to be solved for (and are unique to each signature), are A and B each of which are $2d \times 2d$ matrices. There are many methods for learning them, including that of Yule-Walker [8], or that used here, due to Reynard et al [9].

Once A and B are known, a new point in the sequence can be iteratively generated using Equation 1.

3 The Basic Algorithm

Here we present the basic algorithm for generating new video clips, based on an example sequence.

The first step is to perform principal component analysis on the sequence. Figure 1 shows the result of transforming a 211 image clip of a campfire sequence into a two-dimensional eigenspace, that is plotting the response of each frame to the two eigenvectors having the largest eigenvalues. The 'signature' of the clip is what gives the sequence its particular look-and-feel; it is not simply a random walk-through of the space. If we use the auto-regressive process to model the signature (which includes information about the position of the previous two images in the space), a truly convincing result is obtained. One signature generated via the ARP is shown on the right of Figure 1.

Once such a signature is created, it is now straightforward to project from the signature back into the image space. In the above example, a two-dimensional space has been used for the sake of clarity. In practice if only two eigenvectors are used, then the reconstructions are too blurred to be useful. There are two obvious ways around this :

- Use more eigenvectors for the reconstruction. For the campfire reconstructions shown in Figure 2, nine eigenvectors have been used and give clear, sharp images.
- Use a small number of eigenvectors and, after recreating the images, find the closest frame from the original sequence using some distance metric. While such an approach is appealing in that it guarantees sharp images, it violates our desire to produce images that were never part of the original sequence.

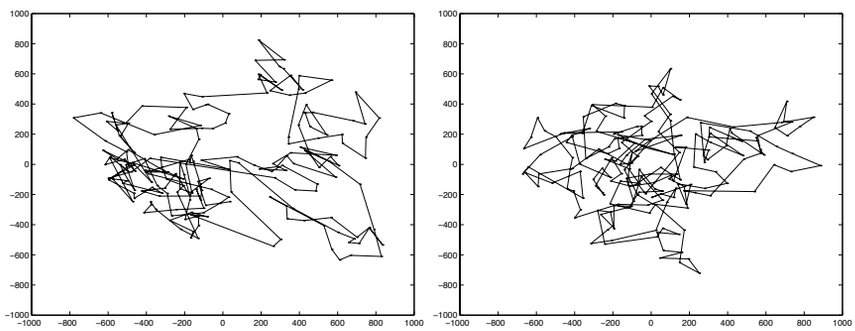

Figure 1: (Left) The original campfire sequence projected into a two-dimensional eigenspace for clarity. It is this 'signature' that we aim to model automatically. The axes show response to the dominant eigenvectors of the campfire images. (Right) One reconstruction of the campfire signature automatically generated using an auto-regressive model.

Figure 2: Three of the resulting frames from the new sequence of the campfire. All frames are synthesised, never having appeared in the original clip.

The approach has been tried on many different sequences and works well for some cases. However, many sequences have non-linearities after performing PCA that violate the assumption that the distribution of the signature is Gaussian. We overcome these limitations in two different ways as explained in the next section.

4 Advanced Algorithms

4.1 Periodic Motions

In many cases the resulting eigenspaces of image sequences are correlated non-linearly and exhibit distinctly non-Gaussian statistical properties. Figure 3 shows four frames from a 136 frame sequence of a horse walking on a treadmill. The Figure also shows a plot of the response of the sequence to the three principal modes of the corresponding eigenspace. The responses are very non-linear, lying on a 'shell' in the three-dimensional space. This is unsurprising since the first eigenvectors deal with the periodic movement of the horses legs and body, and have a $\sin()$ vs. $\cos()$ pattern [13], that forms a 'saddle' shape in three dimensions.

A Gaussian representation of this data can be generated if this space can be transformed in an appropriate manner. To this end, a spline model was used to model the

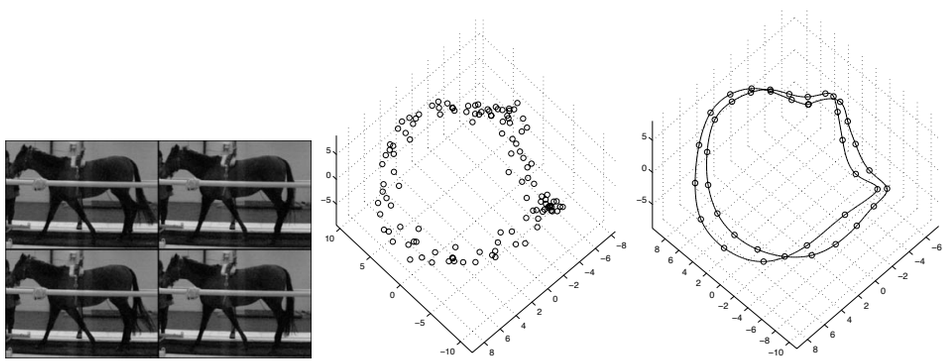

Figure 3: (Left) Four frames from a sequence of a horse walking on a treadmill. (Middle) A three-dimensional plot of the first three principal modes of the corresponding eigenspace for the horse sequence. (Right) A three-dimensional plot of the fitted spline model. The circles show the centres of the GMM trained on the original sequence.

trajectory through the eigenspace over time. Because of the symmetric and periodic nature of the animals shape and gait, the trajectory through the eigenspace is non-trivial. For each cycle of gait the trajectory loops back on itself as well as containing two cusps corresponding to the two maximal extensions of the animals gait. To overcome these difficulties, time, as a progression through each gait cycle, is included in a clustering process in order to generate a continuous n dimensional spline model of a gait cycle. The image responses to the eigenspace are modelled as a mixture of 40, d-dimensional spherical Gaussians (see Bishop [10] for details of the Gaussian Mixture Model).

This model was used in order to smoothly model the density and detail of the data. Other clustering techniques, such as k-means would be too coarse and require a large k and thus loose the smoothness of the spline, whereas diagonal and fully covariant mixture models are too general given that there is no need to model n dimensional scale and rotation. In concept, the idea of decomposing the data using Gaussians is similar to the Mixture Density Network of Bishop[14]. On the right of Figure 3 a three-dimensional plot of the spline fitted to the data is shown.

Given such a model, the minimum distance of each training data point from the spline can be calculated. Each training point then exists on a plane that is perpendicular to the direction of the spline and its offset on this plane with respect to the corresponding point on the spline can be calculated. Figure 4 shows the offsets of the training data from the spline model (the offset signature) and also the temporal displacement along the spline (the speed signature). The above process is described in three dimensions for clarity and the mathematics is easily extended to higher dimensions. As can be seen, both the offset signature and the speed signature are now approximated well by a Gaussian distribution, and hence may be modelled via an ARP. These signatures are then used to reconstruct images from the original eigenspace to give new sequences that vary in realistic ways but are always slightly different from the original. One ARP reconstruction of the speed signature is shown in Figure 5, along with one image from the sequence reconstructed using both signatures.

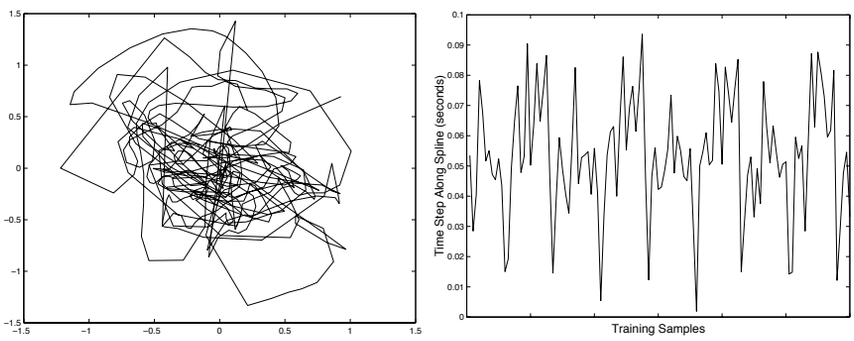

Figure 4: Once a spline is fitted to the data shown in Figure 3, it can be decomposed into an offset perpendicular to the spline (Left), and also distance along the spline, which sets the speed of the gait (Right). Both plots have Gaussian distributions.

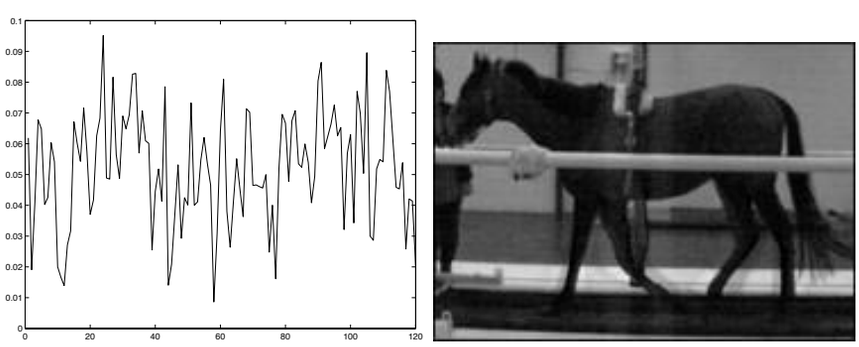

Figure 5: (Left) A plot of the reconstruction of the offset (speed) of the training data along the spline model. (Right) One frame from the new, synthetically generated, horse sequence.

4.2 Combined Appearance

The above non-linearity is very pronounced and could, perhaps, have been predicted *a priori*. However, in the case of the laughing man image sequence, some frames of which are shown in Figure 6, it would not have been so easy to predict the non-linearities obtained. Figure 7 shows the video clip projected into a two-dimensional space. The distribution of the data is clearly non-Gaussian, mostly lying around the edge of the cluster, not the inside. A spline-fitting technique is less appropriate here, since there are no tight clusters of data to fit a spline to.

Once again we decompose the data into two components; this time using the position of the person, and their appearance regardless of position. Such an approach was first derived for face tracking in the combined appearance model of Cootes and Taylor [4].

The first step is to derive the 'shape space'. Landmark points are chosen on the person. These include points around the face, hair and body. The landmark points used for the laughing man example are shown in Figure 8. Each image could be labelled with these points manually, but we speed up the process by making an initial best guess, using a neu-

Figure 6: Four images from the 'laughing man' image sequence. In the video clip a man is laughing, talking and smiling in a very animated manner, throwing back his head and moving his eyebrows.

ral network, in an identical manner to that described previously by Gibson *et al*. [2]. The (x, y) position of these points are formed into one vector per image, and PCA performed on these. This leads to a 'shape space' showing the variation of the landmark points.

Now every image is transformed from its original position to a 'neutral pose' using an image warping algorithm. The algorithm used here is that proposed by Bookstein [11]. Every landmark point is transformed to its new location in the neutral position by deforming the texture map using a thin-plate spline technique. This leads to a sequence of images that have had all positional variation removed from them, but whose textures vary. Differences in facial expression are still clearly visible. Principal components analysis is performed on these images to create a second sub-space, the 'texture space'. One eigenvector derived from the texture space is shown in Figure 8. The eigenvector explains (amongst other variations) how open the mouth is when smiling.

The shape space and the texture space are now combined, once again using PCA (since the position of the face and its texture are strongly coupled) by concatenating the response for each frame to the two spaces. This leads to the combined appearance model which, in a low-dimensional space, models both the position and texture of the sequence. The response of each frame of the original sequence in this new space is shown on the right of Figure 7, which is now Gaussian.

It is in this combined space that we learn the ARP and, hence, synthesise a new sequence. Once the model is learned we need to project back from the combined appearance space into the shape and texture spaces. Each point leads to a texture which is then warped back from the neutral pose using the shape space information, leading to a new frame synthesised entirely from the eigenvector information. Figure 9 shows some of the new frames synthesised by this approach using 20 eigenvectors. The distribution of the original and the synthetic signatures are very similar. The corresponding synthesised video clip is very convincing, containing all the characteristics of the original clip.

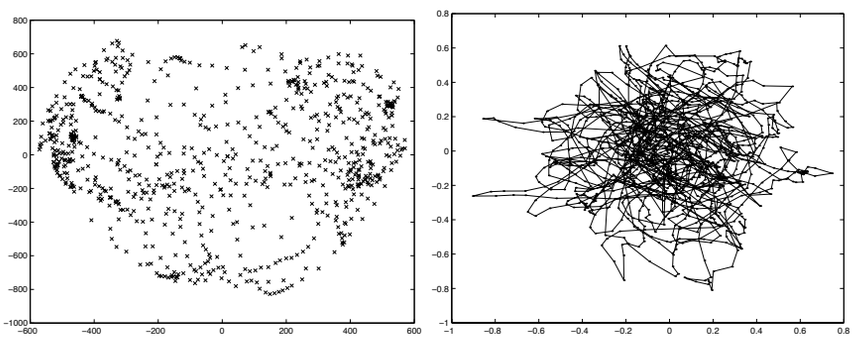

Figure 7: (Left) Signature of video clip of the laughing man in a two-dimensional eigenspace. The distribution of the data is clearly non-Gaussian; we have omitted joining the points together using lines to emphasise their nature. (Right) Signature of video clip of the laughing man in a combined appearance space.

5 Conclusions

We have demonstrated approaches suitable for reconstructing unique video clips based on an original sequence. In many cases, simply applying PCA then an ARP leads to unconvincing results. To overcome some the problems of non-linearities in the data, we decompose the 'signature' of the clip either using a spline fitting approach or a combined appearance model, and then model the nature of these signatures using a second-order auto-regressive process. It could be argued that simply using a non-linear ARP would be just as appropriate as the two approaches demonstrated here. In our experience, however, such techniques are fraught with problems, and require more user intervention than those described here.

The second-order ARP used here may not be the only suitable method of modelling time-series data of this type. Certainly it is known to result in data whose high-frequency component is over-emphasised and a large number of example frames are typically needed to obtain valid results. Higher-order models may be applicable, although we are happy with the results on the examples demonstrated here.

The results of the process are extremely impressive, resulting in new video clips that convey the characteristics of the movements in the original sequence, without simply reusing frames from that sequence.

Acknowledgements

The GMM code used here is based in part on Netlab:
`http://www.ncrg.aston.ac.uk/netlab`
We are grateful to F.L. Bookstein for making available his image warping code. Thanks also to the laughing man, Colin Jones.

Figure 8: (Left) The points manually labelled on the laughing man. Key features are chosen, such as the tip of the nose, the lips, eyes and shoulders. Any point selected in this manner will be transformed exactly onto its position in the neutral pose, and points nearby will be warped using a spline approximation. (Middle and Right) One texture eigenvector with the mean image added back in. This particular eigenvector describes (amongst other things) how open (-ve values) or closed (+ve values) the mouth is.

References

[1] A. Blake and M. Isard. *Active Contours*. Springer, 1998.

[2] D.P. Gibson, N.W. Campbell, C.J. Dalton, and B.T. Thomas 2000. Extraction of Motion-Based Information from Image Sequences. In *International Conference on Pattern Recognition*, IEEE Computer Society, A. Sanfeliu, J. J. Villanueva, M. Vanrell, R. Alquezar, T. Huang, and J. Serra, Eds., 893–896.

[3] Schödl, A., Szeliski, R., Salesin, D. H., and Essa, I. Video Textures. In *Siggraph 2000, Computer Graphics Proceedings*, ACM Press / ACM SIGGRAPH / Addison Wesley Longman, K. Akeley, Ed., Annual Conference Series, 489–498, 2000.

[4] T.F.Cootes, G. Edwards, and C.J.Taylor. Active Appearance Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **23**(6):681–685, 2001.

[5] V.E. Devin and D.C. Hogg. Reactive Memories : An Interactive Talking Head. *British Machine Vision Conference*, 603–612, 2001.

[6] R. Brunelli and T. Poggio. Face Recognition: Features versus Templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **15**(10):1042–1052, 1993.

[7] R.W. Picard and T. Kabir. Finding Similar Patterns in Large Image Databases. *IEEE ICASSP*, **5**:161–164, April 1993.

[8] L. Ljung. *System Identification : Theory for the User*. 2nd Edition, Prentice Hall PTR, 1999.

[9] D. Reynard, A. Wildenberg, A. Blake, and J. Marchant. Learning Dynamics of Complex Motions from Image Sequences. *Proc. European Conference on Computer Vision*, **1**:357–368, 1996.

Figure 9: Four frames from a new laughing man sequence. All of the images shown here are synthesised and were not present in the original sequence.

[10] C. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, New York, 1995.

[11] F.L. Bookstein. Principal Warps: Thin-Plate splines and the decomposition of deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**(6), June 1989

[12] A.W. Fitzgibbon. Stochastic rigidity: Image registration for nowhere-static scenes . *Proceedings of the Eighth International Conference On Computer Vision (ICCV-01)*, 662–669, July 2001.

[13] D.P. Gibson. *The Application of Computer Vision to Very Low Bit-Rate Communications*. PhD thesis, Department of Computer Science, University of Bristol, October 1999.

[14] C.M. Bishop. Mixture Density Networks. Technical report, Aston University, NCRG/94/004, 1994.

[15] L.-M. Reissell and D.K. Pai. Modeling Stochastic Dynamical Systems for Interactive Simulation. EG 2001 Proceedings, Computer Graphics Forum, **20**(3):339–348, 2001.