Object and Action Classification with Latent Variables

Hakan Bilen[1]
hakan.bilen@esat.kuleuven.be
Vinay P. Namboodiri[1]
vinay.namboodiri@esat.kuleuven.be
Luc J. Van Gool[1,2]
luc.vangool@esat.kuleuven.be

[1] ESAT-PSI/IBBT
VISICS/K.U. Leuven
Leuven, Belgium

[2] Computer Vision Laboratory
BIWI/ETH Zürich,
Zürich, Switzerland

Abstract

In this paper we propose a generic framework to incorporate unobserved auxiliary information for classifying objects and actions. This framework allows us to explicitly account for localisation and alignment of representations for generic object and action classes as latent variables. We approach this problem in the discriminative setting as learning a max-margin classifier that infers the class label along with the latent variables. Through this paper we make the following contributions a) We provide a method for incorporating latent variables into object and action classification b) We specifically account for the presence of an explicit class related subregion which can include foreground and/or background. c) We explore a way to learn a better classifier by iterative expansion of the latent parameter space.

We demonstrate the performance of our approach by rigorous experimental evaluation on a number of standard object and action recognition datasets.

1 Introduction

In object detection, which includes the localisation of object classes, people have trained their systems by giving bounding boxes around exemplars of a given class label. Here we show that the classification of object classes, i.e. the flagging of their presence without their localisation, also benefits from the estimation of bounding boxes, even when these are not supplied as part of the training. The approach can also be interpreted as exploiting non-uniform pyramidal schemes. As a matter of fact, we demonstrate that similar schemes are also helpful for action class recognition.

In this paper we address the problem of classification for objects (e.g. person or car) and actions (e.g. hugging or eating) [1–3] in sense of Pascal VOC [4], i.e. indicating the presence but not spatial/temporal localisation (the latter is referred to as detection in VOC parlance). The more successful methods are based on a uniform pyramidal representation built on a visual word vocabulary [3, 10, 20]. In this paper, we augment the classification by adding more flexible spatial information. This will be formulated more generally as inferring additional unobserved or 'latent' dependent parameters. In particular, we focus on two such types of parameters:

BMVC 2011 http://dx.doi.org/10.5244/C.25.17

- The first type specifies a cropping operation. This determines a bounding box in the image. This box serves to eliminate non-representative object parts and background.

- The second type specifies a splitting operation. It corresponds to a *non-uniform* image decomposition into 4 quadrants or temporal decomposition of a spatio-temporal volume into 2 video sequences.

Apart from using these operations separately, we also study the effect of applying and jointly learning both these types of latent parameters, resulting in a bounding box which is also split. In any case, uniform grid subdivisions are replaced by more flexible operations.

While it is possible to learn the latent variables by using a separate routine [14], we adopt a principled max-margin method that jointly infers latent variables and class label. This we solve using a latent structured support vector machine (LSVM) [21]. We also explore an extension of the LSVM by initially limiting the latent variable parameter space and iteratively growing it. Those measures were observed to improve the classification results.

Our work can be seen as complementary to several alternative refinements to the bag-of-words principle. As a matter of fact, it could be combined with such work. For instance, improvements have also been obtained by considering multiple kernels of different features [6, 18]. Another refinement has been based on varying the pyramidal representation step by considering maximal pooling over sparse continuous features [3, 20].

The research related to action classification has mainly followed a bag of words approach as well. Early work towards classification of actions using space-time interest points (STIP) [8] was proposed by Schüldt *et al.* [15]. A detailed evaluation of various features was carried out lately by Wang *et al.* [19].

In this paper we use the latent variable estimation proposed by Yu and Joachims [21]. Self-paced learning has recently been proposed as a further extension for the improved learning of latent SVMs [7], but was not used here.

Related recent work uses latent variables for object detection [2, 17]. In object detection, the loss function differs from the zero/one classification loss. Moreover, in our classification framework, the delineation of objects or actions need not be as strict. Thus, the typical overlap criterion of detection is not valid.

The main contributions of this paper are threefold, a) the introduction of latent variables for enhanced classification and the identification of relevant such parameters, b) a principled technique for estimating them in the case of object and action classification, and c) the avoidance of local optima through an iteratively widened parameter space.

The remainder of the paper is structured as follows. Section 2 introduces some general concepts and methods which we use. Section 3 describes the latent parameter operations and how they are included in the overall classification framework. Section 4 introduces an iterative learning approach for these latent variables. Section 5 describes the results on standard object and action classification benchmarks. Section 6 concludes the paper.

2 Background to the Method

2.1 Feature Representation

Let P be a set of M D-dimensional descriptors $[p_1, ..., p_M]^T \in \mathbb{R}^{M \times D}$ extracted from an image/video. We apply the K-means clustering algorithm to the extracted descriptors from the

training samples to form a codebook $V = [v_1, ..., v_K]^T$ with K cluster centres. There are different ways of coding each descriptor p_i into a K dimensional vector q_i to generate the final image/video representation. We denote the set of codes as $Q = [q_1, ..., q_M]^T$ for the input P. The traditional method is the hard vector coding (VQ) for the image/video classification datasets. It assigns each descriptor to the nearest neighbour in V and leads up to a q vector with a single 1 and 0s otherwise. We denote the set of codes computed with VQ by Q^{VQ}. A recent coding scheme [20] which significantly improves the image representation for image classification applications is the locally linear coding (LLC). It relaxes the cardinality restriction on q and generates a locally smooth sparse representation by incorporating the locality constraint. We denote the LLC code set by Q^{LLC}.

Given the coding for the descriptors, we can have different image/video representations. One of the widely used representations is *bag-of-features* (BoF) which represents an image/video with a histogram of local features. A histogram with VQ coding and BoF representation is

$$z = \frac{1}{M} \sum_{m=1}^{M} q_m^{VQ}. \tag{1}$$

On the other hand, one uses maximal pooling to compute histograms for LLC coding [20].

$$z = \max\{q_1^{LLC}, q_2^{LLC}, \ldots, q_m^{LLC}\} \tag{2}$$

where the *max* operator selects the maximum value for each component among the different vectors. We denote the final image/video representations computed with the BoF method for the VQ and LLC schemes by Φ_{BoF}^{VQ} and Φ_{BoF}^{LLC} resp. The BoF representation discards the spatial/temporal layout of the image/video structure since it uses an unordered set of descriptors. A more extensive representation is spatial pyramidal matching (SPM) [10] which incorporates spatial information into the features by using a pyramidal representation. The method partitions the image into $2^l \times 2^l$ equal sized subregions at different scales l and computes the individual histograms for each subregion and generates the final representation by concatenating them. Similarly to the BoF method, individual histograms within each subregion can be computed by using Eq.(1,2) for the VQ and LLC coding. We present the feature vector represented with the SPM method for the VQ and LLC schemes by Φ_{SPM}^{VQ} and Φ_{SPM}^{LLC} resp.

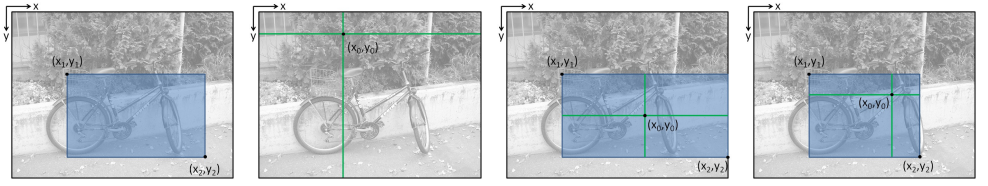
2.2 Structured Learning with Latent Parameters

Suppose we are given a training set $\mathscr{S} = \{(x_1, y_1), ..., (x_n, y_n)\}$ where $x_i \in \mathscr{X}$ are the input images/videos and $y_i \in \mathscr{Y}$ are their class labels. We want to learn a discriminant function $g : \mathscr{X} \rightarrow \mathscr{Y}$ which predicts the class label of unseen examples. In our applications input-output pairs also depend on unobserved latent variables $h \in \mathscr{H}$. Therefore we learn the mapping in the structured learning framework of [21],

$$g(x) = argmax_{(y,h) \in \mathscr{Y} \times \mathscr{H}} f(x, y, h). \tag{3}$$

where $f(x, y, h)$ is a discriminative function that measures the matching quality between input x and output y.

For training the discriminant function, we follow the generalized support vector machine



(a) crop (b) split (c) crop-uniform split (d) crop-split

Figure 1: Illustrative Figure for Latent Models - Images

in margin rescaling formulation [21],

$$\begin{aligned} &\min_{\omega,\varepsilon_i\geq 0} \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{n}\xi_i, \\ &\text{subject to} \max_{h_i\in\mathscr{H}} \omega\cdot\left(\Phi\left(x_i,y_i,h_i\right)-\Phi\left(x_i,\hat{y}_i,\hat{h}_i\right)\right)\geq\Delta\left(y_i,\hat{y}_i\right)-\xi_i, \\ &\forall\hat{y}_i\in\mathscr{Y},\forall\hat{h}_i\in\mathscr{H},i=1,\ldots,n \end{aligned} \tag{4}$$

where $f(x_i,\hat{y}_i,\hat{h}_i)=\omega\cdot\Phi\left(x_i,\hat{y}_i,\hat{h}_i\right)$, ω is a parameter vector and $\Phi\left(x_i,\hat{y}_i,\hat{h}_i\right)$ is a joint feature vector. $\Delta\left(y_i,\hat{y}\right)$ is the loss function that penalizes misclassification. Since our applications require multiclass classification, we design our loss function as

$\Delta\left(y_i,\hat{y}\right)=100\left[y\neq\hat{y}\right]$, with $[\,]$ the Iverson brackets and our feature vector as

$$\Phi_{multi}(x,y,h)=\begin{pmatrix} 0 & \ldots & 0 & \Phi(x,y,h) & 0 & \ldots & 0 \end{pmatrix}^T \tag{5}$$

where the feature vector $\Phi(x,y,h)$ is concatenated into position y. It should be noted that the problem reduces to the Standard Structural SVM formulation [16] in the absence of latent variables. It is used as the learning tool for the baseline approach.

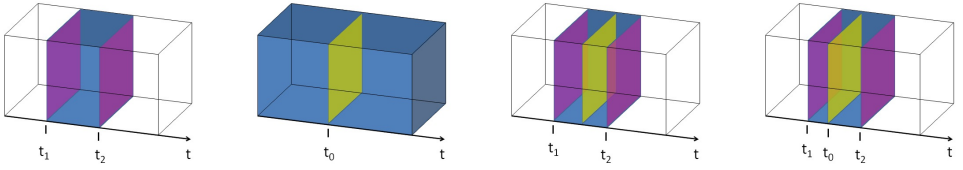
The solution to the optimisation problem in Eq.(4) cannot be posed as a convex energy function due to the dependency of h_i on ω. However, Yu *et al.* [21] adopted an alternating optimisation scheme, the Concave-Convex Procedure (CCCP) for the LSVM to find a local minimum. In section 4 we suggest an iterative method for improving the exploration of the latent space of variables.

3 Latent Models

In this section we explain the use of different selections of latent parameters to explore the spatial and temporal decomposition of images and video sequences resp. We show illustrative figures for our latent models in Fig.1 and Fig.2 and some representative classification examples from the Graz-02 dataset in Fig.3-6. We now discuss the two basic operators represented by our latent variables, cropping and splitting, in turn.

3.1 Crop

Our first latent model is motivated by the consideration that including the class related content and discarding the irrelevant and confusing parts should provide a better discriminant function for classification. Therefore we use a rectangular bounding box to separate two



(a) crop (b) split (c) crop-uniform split (d) crop-split

Figure 2: Illustrative Figure for Latent Models - Videos

(a) (b) (c) (d) (e)

Figure 3: Representative crop examples from the Graz-02 dataset

parts. The bounding box is represented by two points for both spatial and temporal cropping. We denote the latent parameter set with $h_{crop} = \{x_1, y_1, x_2, y_2\}$ and $h_{crop} = \{t_1, t_2\}$ for images and video sequences resp. An illustrative figure for each latent model is shown in Fig.1.(a) and Fig.2.(a).

We illustrate cropping samples with blue drawn bounding boxes for the object classes from the Graz-02 dataset in Fig.3. Differently from object detection methods, our method does not require to localize objects accurately. Instead it can discard non-representative object parts like the handlebar or the seat of bikes, which may not be quite representative for their class due to variations in their appearance (Fig.3(a)). It can also include more than one object in a bounding box (Fig.3.(b-c)). Moreover, it can include additional context information related to the object, for instance a car extended with some road background (Fig.3.(d)).

3.2 Split

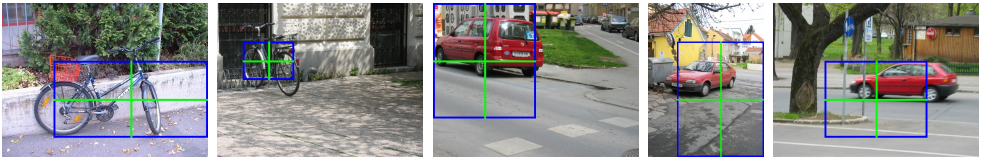
It is known that using pyramidal subdivisions of images or videos improves the classification of objects and actions [9, 10]. Therefore, it stands to reason to consider a pyramid-type subdivision, but with added flexibility. Rather than splitting an image uniformly into equal quadrants, we consider splitting operations that divide into unequal quadrants. In the same vein, we allow a video fragment to be temporally split into two parts, which are not halves. Indeed, a uniform split would probably not keep all object or action evidence within the same subdivision.

Note that in this paper we only consider a single layer of subdivision of the pyramid, the extension to full multi-layers pyramids is not covered yet. Hence, such split is fully characterised by one point. We denote the latent variable set with $h_{split} = \{x_0, y_0\}$ (Fig.1.(b)) and $h_{split} = \{t_0\}$ (Fig.2.(b)) for images and videos resp.

We show splitting samples for bikes with green crossing lines in Fig.4. We observe that bikes are often located in the left and right bottom cells, but also that they are not purely segregated into a single 'quadrant'. This is not crucial for our classification task, however. Cropping can do a better job at object or action segmentation, as it has more degrees of

(a) (b) (c) (d) (e)

Figure 4: Representative split examples for the bike class from the Graz-02 dataset



(a) (b) (c) (d) (e)

Figure 5: Representative crop-uniform split examples from the Graz-02 dataset

freedom at its disposal, yet this very aspect renders good croppings more difficult to learn. Thus, it is not a foregone conclusion that cropping will perform better than splitting.

3.3 Crop - Uniform Split

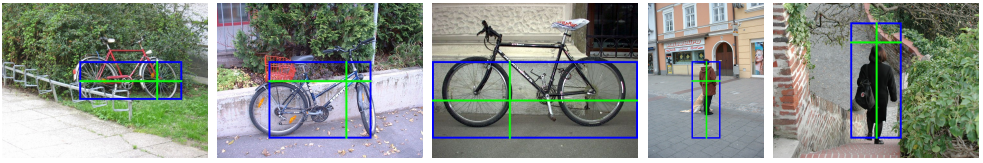
Our Crop - Uniform Split model learns a cropped region, which is subdivided further into equal parts, in order to enrich the representation in pyramid-style. The latent parameter set is that of the cropping. The model is illustrated in Fig.1.(c) and Fig.2.(c). We illustrate crop-uniform splitting examples with blue cropping boxes and green uniform splits in Fig.5. Fig.5 heralds more effective model learning than through uniform splitting only. The richer representation of cropping and uniform splitting will in section 5 be seen to outperform pure cropping.

3.4 Crop-Split

The Crop-Split model comes with the highest dimensional latent parameter set. It learns a cropping box and non-uniform subdivision thereof. Its latent parameter set is a combination of the Crop and Split models, $h_{crop+split} = \{x_0, y_0, x_1, y_1, x_2, y_2\}$ for images and $h_{crop+split} = \{t_0, t_1, t_2\}$. The latent models are illustrated in Fig.1.(d) and Fig.2.(d) resp. We illustrate crop-split examples with blue cropping boxes and green splits in Fig.6. This figure already suggests that the crop-split model is able to locate objects, although we do not use any ground truth bounding box locations in training.

4 Iterative Learning of Latent Parameters

Learning the parameters of an LSVM model often requires solving a non-convex optimisation problem. Like every such problem, LSVM is also prone to getting stuck in local minima. Recent work [1] proposes an iterative approach to find better local minima within shorter convergence times for non-convex optimisation problems. It suggests to first train



(a) (b) (c) (d) (e)

Figure 6: Representative crop-split examples from the Graz-02 dataset

the learning algorithm with easy examples and to then gradually feed in more complex examples. This procedure is called curriculum learning. The main challenge of curriculum learning is to find a good measure to quantify the difficulty of samples. In this paper, we take the size of the parameter space as another indication of the complexity of the learning problem. Therefore, we run our learning algorithm within a limited latent subspace initially and gradually increase the latent parameter space rather than providing examples that get gradually harder.

5 Experiments

We evaluate our system on four publicly available computer vision benchmarks, the Graz-02, the PASCAL VOC 2006 and the Caltech 101 datasets for object classification, and the activities of daily living life dataset for action classification.

For the object classification experiments, we extract dense SIFT features [11]. These are then used to obtain feature representations using regular grids over images as described in 2.1. We use discretisations of 4×4 and 8×8 regular grids. The video sequences we subdivide into temporal cells with a step size of 5 frames. Each temporal cell is described by a set of the HoF descriptors [9] located at the detected Harris3D interest points [8]. We apply K-means to the randomly sampled 100,000 descriptors from the training images/videos to form the visual codebook. The dimension of the visual codebook for the images and videos are chosen as 1024 and 1000 resp.

We compare the performance of the proposed latent models, 'crop', 'split', 'crop-unisplit', 'crop-split' to the standard BoF and one level SPM representations. We provide the performances of those representations with different grid sizes and coding methods. Our implementation of the proposed latent learning makes use of the Latent SVM-Struct [21] algorithm. To obtain a fair comparison, we use the most compatible learning approach provided in the Multiclass SVM-Struct package [16] to get the baseline results. It should be noted that the performance criterion in the experiments is the average multiclass classification accuracy.

5.1 Graz-02 Dataset

The Graz-02 dataset contains 1096 natural real-world images in three object classes, bikes, cars and people. This database includes considerable amount of intra-class variation, varying illumination, occlusion, and clutter. We form 10 training and testing sets by randomly sampling 150 images from each object class for training and use the rest for testing. Note that we use the mean of classification accuracy from the 10 experiments for our evaluation.

Table 1 shows the multiclass classification results. The crop latent model improves the classification performance over the BoF representation around 4-5 percent for all the dif-

Grid Size	Coding	BoF	crop	SPM	split	crop-uni-split	crop-split
4x4	VQ	81.08	86.32	78.62	82.11	85.65	**85.67**
8x8	VQ	80.40	86.80	81.47	81.67	**86.89**	86.59
4x4	LLC	84.09	87.51	87.23	88.36	89.04	**89.07**
8x8	LLC	85.22	89.74	87.74	89.31	90.39	**90.74**

Table 1: Classification Accuracy for the Graz-02 Dataset

Grid Size	Coding	BoF	crop	SPM	split	crop-uni-split	crop-split
4x4	VQ	55.07	56.36	57.89	57.94	**57.70**	57.51
8x8	VQ	55.07	58.09	57.89	57.27	**59.38**	59.23
4x4	LLC	54.93	55.74	61.10	61.05	61.53	**61.63**
8x8	LLC	54.93	55.98	61.10	**62.30**	60.54	61.87

Table 2: Classification Accuracy for the VOC2006

ferent settings. The non-uniform split model also achieves better classification performance than the uniform split. The crop-split model has more degree of freedom than the crop-uni-split model. Therefore it usually outperforms the crop-uni-split. Moreover, the best classification, indicated by bold characters, is always performed by one of the proposed latent models. We also show that finer grid discretisation (8×8) enables the learning of more effective latent models.

5.2 PASCAL VOC 2006

The PASCAL VOC 2006 dataset consists of 5,304 images for 10 object classes and provides a train/validation and a test set. We refer to [4] for details. The dataset is mostly collected from the internet and poses a challenging test bed for detection and classification. It includes images with multiple object labels. Since we focus on classification rather than detection, we modify the dataset by removing 702 multi-labelled examples.

Table 2 shows the multiclass classification results for the dataset. This dataset has more variance in object and background appearance as well as pose than the Graz-02 dataset. The crop model is more consistently useful for this dataset than the split model. A case where splitting does work is for the 8×8 setting for LLC codebook representation where the latent variable was learnt more accurately.

5.3 Caltech-101 Dataset

The Caltech-101 dataset [5] contains images from 101 object classes and an additional background class, i.e. 102 classes in total. The number of images per class varies from 31 to 800. The dataset does not provide sufficient examples for some of the object classes to learn the enriched object models. Thus we sort the object classes in terms of their number of examples and pick the top 15 classes with the most images. Subsequently we form 10 training and testing sets by randomly sampling 50 images from each class of the reduced dataset.

Table 3 depicts the classification results for the Reduced Caltech 101 dataset. This dataset is clean. The objects are placed in the image centre. Therefore, the classification accuracy itself is quite high. We still obtain an improvement by the use of latent variables as the crop-split models achieve the highest performance. In this case, the non-uniform split operation

Grid Size	Coding	BoF	crop	SPM	split	crop-uni-split	crop-split
4x4	VQ	81.04	83.22	85.02	83.70	**87.38**	86.31
8x8	VQ	80.77	85.25	85.42	83.84	87.35	**88.25**
4x4	LLC	89.70	89.53	95.41	95.18	95.48	**95.50**
8x8	LLC	89.58	89.60	95.64	95.33	**95.82**	95.76

Table 3: Classification Accuracy for the Caltech 101 Reduced

coding	BoF	crop	SPM	split	crop-uni-split	crop-split
VQ	82.67	84.00	84.67	86.00	**86.67**	86.00
LLC	79.33	72.00	88.00	88.00	**90.67**	88.67

Table 4: Classification Accuracy for the Everyday Actions

does not serve classification as well as the uniform splitting because of the special nature of the dataset. The latent parameters for the non-uniform split yield over-fitting.

5.4 Everyday Actions Dataset

The activities of daily living dataset [12] contains ten different types of complex actions like answering a phone, writing a phone number on a whiteboard and eating food with silverware. These activities are performed three times by five people with different heights, genders, shapes and ethnicities. Videos are taken at high resolution (1280×720 pixels). A leave-one-out strategy is used for all subjects and the results are averaged as in [12].

Table 4 shows the results for action classification on this dataset. Satkin and Herbert [14] have recently explored cropping for classification of this dataset. They used BoF as well and reported an improvement of 0.67% (from 79.33 to 80). In comparison we obtain an improvement of 1.33% from cropping. As can be seen, the best results are obtained with the crop-uni-split model. We see a rare drop in performance for crop in LLC. We surmise that this is due to the small cluster vocabulary size and intend to explore this further. However, the method retains its performance for crop-split and crop-uni-split settings.

5.5 Iterative Learning

We show preliminary results for the iterative learning of latent models. We perform the iterative learning algorithm for the splitting operation. The settings used are LLC coding over an 8×8 grid. We initially constrain the latent search space for the split model to the centre of the images and expand it along the x and y directions by a step size 2 on the 8×8 grid at each iteration. Once the CCCP algorithm converges within the given latent space in the one iteration, we expand the latent search space again at the start of the next. The algorithm terminates when the entire search space is covered.

	Graz-02	VOC 2006	Caltech 101 (R)
LSVM	89.31	62.30	95.33
Iterative LSVM	**89.72**	**62.53**	**95.40**

Table 5: LLC 8x8 Split for LSVM and Iterative LSVM

Table 5 depicts the performance of the iterative splitting operation on the three object classification datasets. We show that the iterative method for LSVM consistently improves the classification accuracy over the original formulation of the LSVM. The preliminary results obtained here are encouraging.

6 Conclusion and future work

We have developed a method for classifying objects and actions with latent variables. We have specifically shown that learning latent variables for flexible spatial operations like 'crop' and 'split' are useful for inferring the class label. We have adopted the latent SVM method to jointly learn the latent variables and the class label. The evaluation of our principled approach yielded consistently good results on several standard object and action classification datasets. We have further improved the latent SVM by iteratively growing the latent parameter space to avoid local optima. In future, we are interested in extending the set of operations that may aid classification and in improving the learning of multiple parameters.

Acknowledgements

We are grateful for financial support from the EU Project FP7 AXES ICT- 269980 and the IBBT iCocoon Project.

References

[1] Y. Bengio, J. Louradour, R. Collobert, and J. Weston. Curriculum learning. In *Proc. Int. Conf. on Machine Learning (ICML)*, pages 41–48. ACM, 2009.

[2] M. B. Blaschko, A. Vedaldi, and A. Zisserman. Simultaneous object detection and ranking with weak supervision. In *Proc. of Advances in Neural Information Processing Systems (NIPS)*, 2010.

[3] Y-Lan Boureau, Francis Bach, Yann LeCun, and Jean Ponce. Learning mid-level features for recognition. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, pages 2559–2566, 2010.

[4] M. Everingham, A. Zisserman, C. K. I. Williams, and L. Van Gool. The PASCAL Visual Object Classes Challenge 2006 (VOC2006) Results.

[5] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *IEEE. CVPR 2004, Workshop on Generative-Model Based Vision*, 2004.

[6] Peter V. Gehler and Sebastian Nowozin. On feature combination for multiclass object classification. In *Proc. of Int. Conf. on Computer Vision (ICCV)*, pages 221–228, 2009.

[7] M. Pawan Kumar, Benjamin Packer, and Daphne Koller. Self-paced learning for latent variable models. In *Proc. of Advances in Neural Information Processing Systems (NIPS)*, pages 1189–1197. 2010.

[8] Ivan Laptev and Tony Lindeberg. Space-time interest points. In *Proc. of Int. Conf. on Computer Vision (ICCV)*, pages 432–439, 2003.

[9] Ivan Laptev, Marcin Marszałek, Cordelia Schmid, and Benjamin Rozenfeld. Learning realistic human actions from movies. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, jun 2008.

[10] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, pages 2169–2178, 2006.

[11] D.G. Lowe. Object recognition from local scale-invariant features. In *Proc. of Int. Conf. on Computer Vision (ICCV)*, page 1150, 1999.

[12] Ross Messing, Chris Pal, and Henry Kautz. Activity recognition using the velocity histories of tracked keypoints. In *Proc. of Int. Conf. on Computer Vision (ICCV)*, Washington, DC, USA, 2009.

[13] Axel Pinz. Object categorization. *Foundations and Trends in Computer Graphics and Vision*, 1(4), 2005.

[14] Scott Satkin and Martial Hebert. Modeling the temporal extent of actions. In *Proc. of European Conf. Computer Vision (ECCV)*, pages 536–548, 2010.

[15] Christian Schüldt, Ivan Laptev, and Barbara Caputo. Recognizing human actions: A local svm approach. In *Int. Conf. on Pattern Recognition (ICPR)*, pages 32–36, 2004.

[16] I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun. Support vector machine learning for interdependent and structured output spaces. In *Proc. Int. Conf. on Machine Learning (ICML)*, page 104, 2004.

[17] A. Vedaldi and A. Zisserman. Structured output regression for detection with partial occulsion. In *Proc. of Advances in Neural Information Processing Systems (NIPS)*, 2009.

[18] Andrea Vedaldi, Varun Gulshan, Manik Varma, and Andrew Zisserman. Multiple kernels for object detection. In *Proc. of Int. Conf. on Computer Vision (ICCV)*, pages 606–613, 2009.

[19] Heng Wang, Muhammad Muneeb Ullah, Alexander Kläser, Ivan Laptev, and Cordelia Schmid. Evaluation of local spatio-temporal features for action recognition. In *Proc. of British Machine Vision Conf. (BMVC)*, page 127, sep 2009.

[20] Jinjun Wang, Jianchao Yang, Kai Yu, Fengjun Lv, Thomas S. Huang, and Yihong Gong. Locality-constrained linear coding for image classification. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, pages 3360–3367, 2010.

[21] Chun-Nam John Yu and Thorsten Joachims. Learning structural svms with latent variables. In *Proc. Int. Conf. on Machine Learning (ICML)*, pages 1169–1176, 2009.