Classification with Global, Local and Shared Features

Hakan Bilen[1], Vinay P. Namboodiri[2], Luc J. Van Gool[1,3]

[1]ESAT-PSI/IBBT,VISICS/KU Leuven, Belgium
[2]Alcatel-Lucent Bell Labs, Antwerp, Belgium
[3]Computer Vision Laboratory, BIWI/ETH Zürich, Switzerland

Abstract. We present a framework that jointly learns and then uses multiple image windows for improved classification. Apart from using the entire image content as context, class-specific windows are added, as well as windows that target class pairs. The location and extent of the windows are set automatically by handling the window parameters as latent variables. This framework makes the following contributions: a) the addition of localized information through the class-specific windows improves classification, b) windows introduced for the classification of class pairs further improve the results, c) the windows and classification parameters can be effectively learnt using a discriminative max-margin approach with latent variables, and d) the same framework is suited for multiple visual tasks such as classifying objects, scenes and actions. Experiments demonstrate the aforementioned claims.

1 Introduction

In this paper, we consider the classification problem of deciding whether one of a number of pre-specified object classes, e.g. bicycle, motorbike, or person, is present in an image. We show that also learning pairwise relations between classes improves such classification: when having to tell whether or not a specific *target class* is present, sharing knowledge about other, *auxiliary classes* supports this decision.

Our method stands in contrast to standard classification approaches that only exploit global [11] or local information [15, 1] about the target class. In particular, we propose a framework that combines target class-specific global and local information with information learnt for pairs of the target class and each of a number of auxiliary classes. The advantage of adding such pairwise information is that it aids generalization. The common context for a class pair helps it being discriminated against other classes. For instance, similar classes like 'bicycle' and 'motorbike' share features that enable to discriminate both from other classes. The target class-specific parts of the models for 'bicycle' and 'motorbike' rather focus on specific nuances that are needed to discriminate between the pair. Even if in this paper we often formulate the approach in terms of object classification, the very same framework will be demonstrated for scene and action classification just the same.

In summary, our target class model combines information about:

1. global image appearance, using a spatial pyramid over the image, thereby providing context information;
2. local appearance, based on a target class-specific window, loosely corresponding to a bounding box;
3. shared appearances, based on a series of windows, each jointly defined for the target class and one of the auxiliary classes with which there are visual commonalities.

We show that all components of this combined representation can be learnt jointly, with as only supervision the class label for the training images (i.e. which target class appears in the images without any information on its location).

We have evaluated our approach for object, scene and action classification tasks using standard benchmarks, after such joint learning of the global, local, and shared components. We have experimentally evaluated each of these components individually and jointly for solving these various problems. The results show that adding the shared component is beneficial in all cases.

2 Related Work

Object classification is a well studied problem. A detailed survey has been presented by Pinz [20]. One of the more successful techniques is the use of spatial pyramid representations [11] over a bag of visual words [6, 27, 1, 2]. One variation has been the use of multiple feature families [6, 26], the organization of the spatial pyramid through sparse representations [27] or joint coding [2]. The above methods all assume a global representation for the whole image. We also integrate locally extracted feature [27] in our work.

The use of local feature representations has recently been considered for the classification of objects [15, 1] and actions [7]. Bilen *et al.* [1] consider the localisation – in the form of a window – as a latent variable that is learnt jointly with other classification model parameters. Our framework generalizes feature localization as we show that instead of using one window, using multiple is beneficial for classification.

The use of multiple appearance contexts has been considered for recognizing scenes [21]. However, that work relies on using different features to capture those different appearance contexts. Recent work by Pandey and Lazebnik [18], follows this line of thought and combines global GIST features with local HOG features. Our work is complementary to these ideas. We focus on obtaining different contexts from a single feature type. Yet, our framework is not restricted to a single feature.

The central contribution of this paper is the use of appearance properties that are shared by pairs of classes. The issue of sharing has so far been explored more for object detection [23, 5, 17] than for object classification, where this is more intricate to implement. In the case of classification we cannot assume that training images come with the locations of objects. Sharing for classification is therefore more challenging. It has been considered in the large margin framework based on a pre-specified hierarchy [3]. We do not rely on such restriction. Sharing has also been implemented by relying on auxiliary information such as text [22] or by constructing hierarchies from WordNet [14]. An interesting recent approach for sharing used other detector information as cues [12]. As a matter of fact, there has also been work that uses the output of classifiers to learn sharing between classes [24]. In contrast to that approach, we learn the sharing together with the classier itself. Moreover, we learn not only to share at the level of a class pair, but also adapt the sharing window to the individual instance of the target class (*i.e.* the window is not at a fixed relative position for the entire class).

3 Model Definition

To build our classifiers, we make use of the structural SVM formulation with latent parameters [28]. In our model, input $x \in \mathcal{X}$, output $y \in \mathcal{Y} = \{c_1, \cdots, c_k\}$ and latent parameters $h \in \mathcal{H}$ correspond to the image, its label, and a set of bounding boxes, respectively. We use discriminant functions of the form $f_\theta : \mathcal{X} \times \mathcal{Y} \times \mathcal{H} \rightarrow \mathcal{R}$ which scores triplets of (x, y, h) for a learnt vector θ of the structural SVM model as

$$f_\theta\left(x, y, h\right) = \theta^y \cdot \Psi^y\left(x, y, h\right) \tag{1}$$

where $\Psi^y\left(x, y, h\right)$ is a joint feature vector that describes the relation among x, y and h. In our model, each $\Psi^y\left(x, y, h\right)$ concatenates histograms which are obtained from multiple rectangular windows with the bag of words (BoW) representation [27]. We use different windows to encode the 3 information channels, *i.e.* global, local, and shared. We can write our feature vector for class y as $\Psi^y\left(x, y, h\right) = \left(\Psi^y_{gl}, \Psi^y_{loc}, \Psi^y_{sh,c_1}, \cdots, \Psi^y_{sh,c_k}\right)$, where the components – again exemplified for object classification – are:

Global Features: $\Psi^y_{gl} = \phi\left(x\right)$ is a histogram vector given image x, more specifically a histogram of quantized densely sampled SIFT descriptors [13] over the whole image x by using the spatial pyramid (SP) representation [27]. For the global features, we use three levels $(1 \times 1, 2 \times 2, 4 \times 4)$ for the SP.

Local Features: $\Psi^y_{loc} = \phi\left(x, h^y_{loc}\right)$ is a histogram over an image part selected with window h^y_{loc}, which roughly corresponds to a bounding box h^y_{loc} around the instance of the target class. We use a two-level SP $(1 \times 1, 2 \times 2)$ over SIFT descriptors for the local feature vector $\phi\left(x, y, h^y_{loc}\right)$.

Shared Features: $\Psi^y_{sh,\hat{y}} = K_{\mathcal{S}}(y, \hat{y})\phi\big(x, h^y_{sh,\hat{y}}\big)$ is a histogram over a window $h^y_{sh,\hat{y}}$. It is a two-level SP over SIFT descriptors. Suppose $\mathcal{S}$ is the set of all class pairs of on the one hand the target class y and on the other hand each one of the auxiliary classes with which the target class is supposed to share information. $K_{\mathcal{S}}(y, \hat{y})$ is an indicator function that outputs 1, if the label pair $(y, \hat{y}) \in \mathcal{S}$, and else is 0. Note that $K_{\mathcal{S}}(y, \hat{y}) = K_{\mathcal{S}}(\hat{y}, y)$. We explain the procedure to obtain $\mathcal{S}$ in section 5.

We can now rewrite the discriminant function (1) by including these feature vectors:

$$f_\theta(x,y,h) = \theta^y_{gl}\cdot\phi(x) + \theta^y_{loc}\cdot\phi(x,h^y_{loc}) + \sum_{\hat{y}\in\mathcal{Y}} K_{\mathcal{S}}(y,\hat{y})\,\theta^y_{sh,\hat{y}}\cdot\phi\left(x,h^y_{sh,\hat{y}}\right) \quad (2)$$

where θ^y_{gl} ,θ^y_{loc}, $\theta^y_{sh,\hat{y}}$ denote the parts of θ^y that correspond to the global, local, and shared parameter vectors resp, *i.e.* we define $\theta^y = \left(\theta^y_{gl},\theta^y_{loc},\theta^y_{sh,c_1},\cdots,\theta^y_{sh,c_k}\right)$ and $\theta = (\theta^{c_1},\cdots,\theta^{c_k})^{\mathrm{T}}$. The set of latent parameters can similarly be written as $h^y = \left(h^y_{loc},h^y_{sh,c_1},\cdots,h^y_{sh,c_k}\right)$ and $h = (h^{c_1},\cdots,h^{c_k})^{\mathrm{T}}$.

We use a common or *shared* parameter vector $\theta^y_{sh,\hat{y}}$ to encode the similarity between the labels y and $\hat{y}$. The equality $\theta^y_{sh,\hat{y}} = \theta^{\hat{y}}_{sh,y}$ means that the classes y and $\hat{y}$ share a common parameter vector. Not adopting that equality renders the model heavier while experiments in section 6.4 show a drop in performance. A graphical illustration of our model for a toy object classification task is shown in Fig.1. The images x_1, x_2 are labeled as c_1, c_2 resp. While there are separate class-specific parameter vectors for the global $\theta^{c_1}_{gl}$, $\theta^{c_2}_{gl}$ and local $\theta^{c_1}_{loc}$, $\theta^{c_2}_{loc}$ channels, an identical parameter vector $\theta^{c_1}_{sh,c_2}$ is shared between the labels c_1 and c_2. The latent parameters are used to learn instance specific shared, rectangular windows $h^{c_1}_{sh,c_2}$ and $h^{c_2}_{sh,c_1}$ as well as the target class-specific rectangular windows $h^{c_1}_{loc}$ and $h^{c_2}_{loc}$.

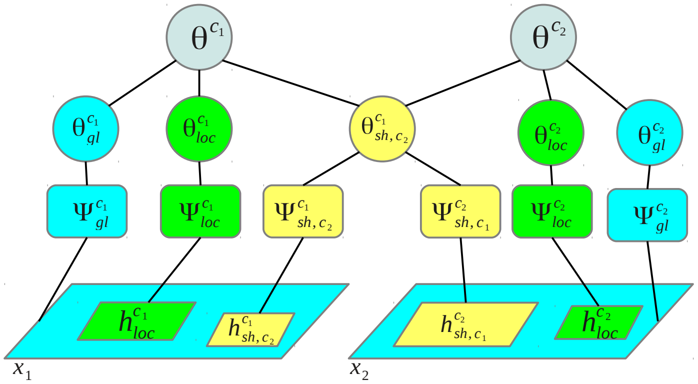

Fig. 1. Graphical illustration of our model for two images containing one target class each. Different types of features are denoted in different colors. Best viewed in color.

4 Inference and Learning

4.1 Inference

The inference problem corresponds to finding a prediction rule that infers a class label and a set of latent parameters for an unseen image. Formally speaking, the prediction rule $g_\theta(x)$ maximizes eq.(1) over y and h given the parameter vector θ and the image x:

$$g_\theta(x) = \arg\max_{y\in\mathcal{Y},h\in\mathcal{H}} f_\theta(x,y,h) \tag{3}$$

Since the windows corresponding to the global, local, and shared models do not depend on each other, the inference can be efficiently solved as follows:

$$\begin{aligned} g_\theta(x) = \arg\max_{y\in\mathcal{Y}} \Bigg[& \theta^y_{gl}\cdot\phi(x) + \arg\max_{h^y_{loc}\in\mathcal{H}} [\theta^y_{loc}\cdot\phi(x,h^y_{loc})] \\ & + \sum_{\hat{y}\in\mathcal{Y},\hat{y}\neq y} \arg\max_{h^y_{sh,\hat{y}}\in\mathcal{H}} \left[K_{\mathcal{S}}(y,\hat{y})\,\theta^y_{sh,\hat{y}}\cdot\phi\left(x,h^y_{sh,\hat{y}}\right)\right]\Bigg] \end{aligned} \tag{4}$$

4.2 Learning

Suppose we are given a set of training samples $\{(x_1,y_1,h_1),\ldots,(x_n,y_n,h_n)\}$ and we want to learn a model θ to predict the class label of an unseen example. Here we assume that each input x_i has only one label y_i. When the set of windows h_i are labeled for the training set, the standard structural SVM [25] solves the following optimization problem:

$$\min_\theta \frac{1}{2}\|\theta\|^2 + C\sum_{i=1}^{n}\left[\max_{y,h^y}[\theta^y\cdot\varPsi^y(x_i,y,h^y) + \varDelta(y_i,y,h_i^{y_i},h^y)] - \theta^{y_i}\cdot\varPsi^{y_i}(x_i,y_i,h_i^{y_i})\right] \tag{5}$$

where C is the penalty parameter and $\varDelta(y_i,y,h_i^{y_i},h^y)$ is the loss function. The loss is taken to be $\varDelta(y_i,y,h_i^{y_i},h^y) = 1$ if $y_i = y$, 0 else. Yet, as the window labels are actually not available for training the classification model, we treat them as latent parameters. To solve the optimization problem in eq.(5) without the labeled windows, we follow the latent SVM formulation of [28]:

$$\min_\theta \frac{1}{2}\|\theta\|^2 + C\sum_{i=1}^{n}\left[\max_{y,h^y}[\theta^y\cdot\varPsi^y(x_i,y,h^y) + \varDelta(y_i,y,h^y)] - \max_h[\theta\cdot\varPsi(x_i,y_i,h^{y_i})]\right] \tag{6}$$

Note that we remove $h_i^{y_i}$ from $\varDelta$ since it is not given.

The above formulation yields a non-convex problem and can be solved by using the Concave-Convex Procedure (CCCP) [29]. Our problem of learning the target class-specific $\theta^y_{gl},\theta^y_{loc}$ and shared $\theta^y_{sh,\hat{y}}$ model parameters is compatible with the latent SVM formulation because the class labels and latent parameters can be optimized for each image individually.

5 Choosing Shared Label Pairs

We have introduced the indicator function $K_{\mathcal{S}}(y,\hat{y})$ to allow for sharing only between the class label pairs which are included in the set $\mathcal{S}$, *i.e.* $K_{\mathcal{S}}(y,\hat{y})$ is 1 if $(y,\hat{y})\in\mathcal{S}$, else it is 0. $\mathcal{S}$ can be designed in various ways. One can include all class pairs in $\mathcal{S}$ and let the learning algorithm determine the weights $\theta^{y}_{sh,\hat{y}}$. However, this approach may lead to a non-optimal solution since sharing between visually very different classes can degrade the classification performance. Including all the class pairs also leads to a computational complexity that is quadratic in the number of classes. Alternatively, one can introduce additional binary latent variables to learn which class pairs should be included in $\mathcal{S}$. However, naively minimizing the loss in eq. (6) with respect to those latent parameters will always result in including all the pairs.

In our experiments, we assume that the classes that are often confused with the target class in classification share enough visual similarities with the target to turn them into good candidates to build the class pairs. We thus only activate the pairwise features for such pairs. We learn a single threshold to obtain $\mathcal{S}$ from the confusion tables of the validation sets. The super-threshold class pairs extracted from the confusion table are symmetric but not necessarily transitive. For example, if the 'bicycle' class shares with 'motorbike' then also vice-versa. However, it may be that 'bicycle' shares with the class 'motorbike' and not 'bus', but 'motorbike' shares with both classes 'bicycle' and 'bus'.

6 Experiments

6.1 Datasets

We evaluate our method on the PASCAL VOC 2006 [4], Oxford Flowers17 [16], Scene15 [11], and TV Human Interactions (Interactions) benchmarks: [19].

VOC2006: This dataset consists of 5,304 images with 10 object categories. We extract dense SIFT features [13] at every fourth pixel at a single scale and quantize them by using a 1024 words dictionary. We use the same training, validation and testing splits as in [1].

Flowers17: The dataset contains 17 flower categories and 80 images from each flower species. We use densely sampled Lab color values and quantize them using an 800 words dictionary. The dataset has three predefined splits including 40/20/20 training-validation-testing images per class. The ground truth pixel-wise segmentation is also available for some images but it is not used in this paper.

Scene15: The dataset contains images from 15 scene categories, covering a wide range, from natural scenes to man-made environments. We extract dense SIFT features [13] at every fourth pixel at a single scale; and quantize them by using a 1024 words dictionary. We apply the same experimental set-up as in [11] and randomly sample 100 images 10 times for training and use the rest for testing. Additionally we randomly pick 30 images for each class from the existing training sets to validate the best threshold for sharing and use the 100 image training splits to train our classifiers.

Interactions: This dataset contains video sequences containing four human interaction types: hand shakes, high fives, hugs, kisses and an additional background class. The videos are collected from over 20 different TV shows. We describe the videos by a set of HOF and HOG descriptors [10] located at the detected Harris3D interest points [9] and quantize them using a 1024 words vocabulary. We use the same training and testing sets as [19]. We randomly pick 40% of the original training set and use them to validate the selection of the best threshold for sharing and report the performance of our method on the original split.

6.2 Implementation Details

We use a sparse encoding of the BoW feature representation in [27] for all 3 of $\Psi^y_{gl}, \Psi^y_{loc}, \Psi^y_{sh,\hat{y}}$, with 5 nearest neighbors and the respective SPs of $(1 \times 1, 2 \times 2, 4 \times 4)$,$(1 \times 1, 2 \times 2)$ and $(1 \times 1, 2 \times 2)$ in these three cases for the images, and $(1, 2, 4)$, $(1, 2)$ and $(1, 2)$ combinations of frames for the videos. Moreover, we adopt a coarse discretization of the latent space $\mathcal{H}$ by forcing the corners to lie on an 8×8 spatial grid and at the boundaries of 32 equal temporal intervals in the case of videos. Our inference and learning algorithms scale linearly with the number of possible windows, thus this discretization significantly shortens the computation times. As our experiments have shown, defining $\mathcal{H}$ at pixel resolution did not substantially improve the classification performance.

6.3 Baselines

In order to evaluate the contribution of the global (gl), local (loc) and shared (sh) features, we report the classification results for each of these feature types individually, and also for their combinations, *i.e.* gl+loc, gl+sh, loc+sh and gl+loc+sh. We refer to gl and loc as the baselines, corresponding to the work by [27] and [1], resp.

6.4 Results

The results for the baselines and the proposed methods are depicted in Table 1. It shows that the best feature selection always includes the shared features. Some examples of inferred local and shared windows are illustrated in Fig. 2. We provide further details about the selected class pairs and corresponding confusion matrices in the supplementary material.

VOC2006: We can observe from Table 1 that using the shared features is always useful. We obtain the best classification accuracy for the configuration 'gl+sh'. This setting improves the baseline method by 3.39%. We compare our algorithm with three additional baselines as shown in Table 2 with the respective 'gl+asym sh', 'sh with gl' and 'gl+full sh'. For the first one, we do not enforce the symmetry constraint $\theta^y_{sh,\hat{y}} = \theta^{\hat{y}}_{sh,y}$. Although this model has a parameter vector with higher dimension, 'gl+sh' still performs better. For the second baseline, in addition to the global features, we use the whole image for sharing by setting all h_{sh} to

		VOC2006 [4]	Flowers17 [16]	Scene15 [11]	Interactions [19]
	gl[27]	53.83	65.58±4.33	75.93±1.95	34.40
Baselines	loc [1]	54.82	63.14±4.01	74.42±1.54	35.20
	gl+loc	54.55	68.72±3.15	77.32±1.92	37.20
Ours	loc+sh	55.16	65.19±5.06	75.16±2.53	37.60
	gl+sh	**58.21**	66.08±3.95	76.47±1.65	**40.00**
	gl+loc+sh	57.59	**71.08±0.68**	**77.45±1.54**	**40.00**

Table 1. Classification results. The results are given as the classification accuracy averaged over the different target classes, in percentages. For the Flowers17 and Scene15 datasets the standard deviation of the accuracy is also given.

gl+sh	gl+asym sh	sh with gl	gl+full sh
58.21	57.64	49.62	58.06

Table 2. The results for three additional baselines.

the entire image size. The result obtained from the second baseline shows that sharing information through smaller learnt windows is beneficial. For the third one, 'gl+full sh', we use all the class pairs to share, *i.e.* $K_{\mathcal{S}}(y, \hat{y}) = 1$ for all $(y, \hat{y})$ pairs with $\hat{y} \neq y$. The result shows that sharing with all the label pairs results in inferior performance.

Flowers17: We obtain an improvement of 5.5% using the combined configuration of the 'gl+loc+sh' model. This is interesting as the dataset involves difficult, fine-grained (subclass) classification, suggesting that the sharing framework better exploits the subtle differences between classes. Adding the shared part of the model always came out to be beneficial and enhance the classification performance.

Scene15: In this case, we obtain an improvement of 1.52% in the mean classification accuracy. Yet, this improvement is smaller than for object classification. This may be because the classes are rather different from each other and sharing visual features therefore holds less promise.

Interactions: In this dataset we obtain an improvement of 4.8% over the baseline method. Again, the accuracy for classifying actions in these videos was improved by adding shared features. This is interesting as the nature of the dataset is quite different from the image classification datasets. The localization here is purely temporal.

7 Conclusion

This paper provides a method for improved visual classification by sharing localized features between selected pairs of classes. We proposed the combined use

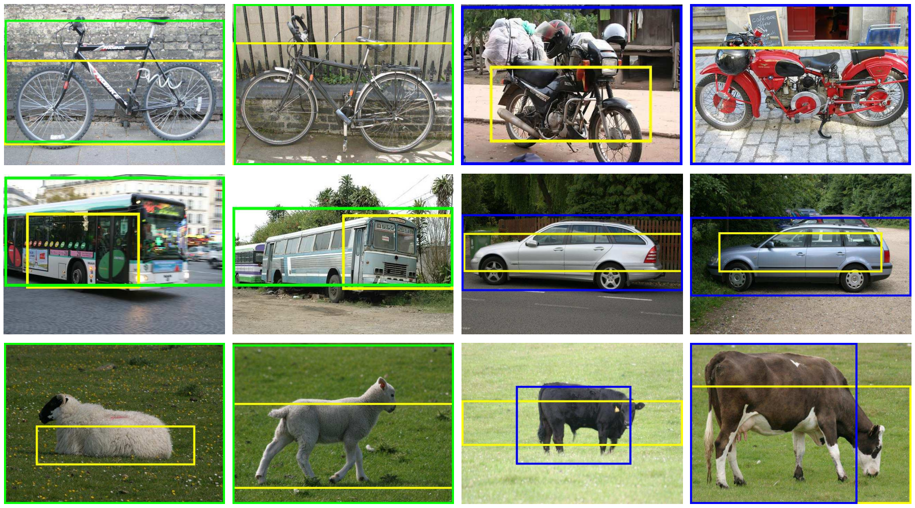

Fig. 2. Some inferred windows for images from VOC2006. Each row consists of two samples for a 'label 1' and a 'label 2' class. Green and blue windows correspond to h^1_{loc} and h^2_{loc} for the labels 1 and 2, resp. Yellow windows indicate those for the shared label pairs $h^1_{sh,2}$.

of global, local, and shared windows. The experimental evaluation has shown that this framework is applicable to a variety of visual classification tasks such as the classification of objects, scenes and actions. Though we have limited the approach to learning pairwise class relations in this paper, the idea could be extended to sharing among larger class groupings by exploiting hierarchical class taxonomies. In the future, we would like to explore this idea further. We also plan to allow for the presence of multiple target classes by considering the recently proposed multilabel structured output techniques [8].

Acknowledgments: This work was supported by the EU Project FP7 AXES ICT-269980 and IBBT iCocoon.

References

1. Bilen, H., Namboodiri, V.P., Van Gool, L.J.: Object and Action Classification with Latent Variables. In: BMVC (2011)
2. Boureau, Y., Le Roux, N., Bach, F., Ponce, J., LeCun, Y.: Ask the locals: multi-way local pooling for image recognition. In: ICCV. IEEE (2011)
3. Dekel, O., Keshet, J., Singer, Y.: Large margin hierarchical classification. In: International Conference on Machine Learning (ICML). pp. 27–35 (2004)
4. Everingham, M., Zisserman, A., Williams, C.K.I., Van Gool, L.: The PASCAL Visual Object Classes Challenge 2006 (VOC2006) Results. http://www.pascal-network.org/challenges/VOC/voc2006/results.pdf

5. Fergus, R., Bernal, H., Weiss, Y., Torralba, A.: Semantic label sharing for learning with many categories. In: ECCV. pp. 762–775 (2010)
6. Gehler, P.V., Nowozin, S.: On feature combination for multiclass object classification. In: ICCV. pp. 221–228 (2009)
7. Hoai, M., Lan, Z.Z., De la Torre, F.: Joint segmentation and classification of human actions in video. In: CVPR (2011)
8. Lampert, C., Austria, I.: Maximum margin multi-label structured prediction (2011)
9. Laptev, I., Lindeberg, T.: Space-time interest points. In: ICCV. pp. 432–439 (2003)
10. Laptev, I., Marszałek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: CVPR (2008)
11. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: CVPR. pp. 2169–2178 (2006)
12. Li-Jia Li, Hao Su, E.P.X., Fei-Fei, L.: Object bank: A high-level image representation for scene classification & semantic feature sparsification. In: Advances in Neural Information Processing Systems (NIPS) (2010)
13. Lowe, D.: Object recognition from local scale-invariant features. In: ICCV. p. 1150 (1999)
14. Marszałek, M., Schmid, C.: Semantic hierarchies for visual object recognition. In: CVPR (2007)
15. Nguyen, M.H., Torresani, L., De la Torre, F., Rother, C.: Weakly supervised discriminative localization and classification: a joint learning process. In: ICCV (2009)
16. Nilsback, M.E., Zisserman, A.: A visual vocabulary for flower classification. In: CVPR. vol. 2, pp. 1447–1454 (2006)
17. Opelt, A., Pinz, A., Zisserman, A.: Incremental learning of object detectors using a visual shape alphabet. In: CVPR. pp. 3–10 (2006)
18. Pandey, M., Lazebnik, S.: Scene recognition and weakly supervised object localization with deformable part-based models. In: ICCV (2011)
19. Patron, A., Marszalek, M., Zisserman, A., Reid, I.D.: High five: Recognising human interactions in tv shows. In: BMVC. pp. 1–11 (2010)
20. Pinz, A.: Object categorization. Foundations and Trends in Computer Graphics and Vision 1(4) (2005)
21. Quattoni, A., Torralba, A.: Recognizing indoor scenes. In: CVPR (2009)
22. Sadeghi, M.A., Farhadi, A.: Recognition using visual phrases. In: CVPR (2011)
23. Salakhutdinov, R., Torralba, A., Tenenbaum, J.: Learning to share visual appearance for multiclass object detection. In: CVPR (2011)
24. Torresani, L., Szummer, M., Fitzgibbon, A.: Efficient object category recognition using classemes. In: ECCV. pp. 776–789 (2010)
25. Tsochantaridis, I., Hofmann, T., Joachims, T., Altun, Y.: Support vector machine learning for interdependent and structured output spaces. In: International Conference on Machine Learning (ICML). pp. 104–112 (2004)
26. Vedaldi, A., Gulshan, V., Varma, M., Zisserman, A.: Multiple kernels for object detection. In: ICCV. pp. 606–613 (2009)
27. Wang, J., Yang, J., Yu, K., Lv, F., Huang, T.S., Gong, Y.: Locality-constrained linear coding for image classification. In: CVPR. pp. 3360–3367 (2010)
28. Yu, C.N.J., Joachims, T.: Learning structural svms with latent variables. In: International Conference on Machine Learning (ICML). pp. 1169–1176. ACM (2009)
29. Yuille, A., Rangarajan, A.: The concave-convex procedure. Neural Computation 15(4), 915–936 (2003)