Object Classification with Adaptable Regions

Hakan Bilen[†] Marco Pedersoli[†] Vinay P. Namboodiri[‡] Tinne Tuytelaars[†] Luc Van Gool[†]
[†] ESAT-PSI-VISICS/iMinds, KU Leuven, Belgium
firstname.lastname@esat.kuleuven.be
[‡] Dept. of Comp. Sci. & Eng., IIT Kanpur, Kanpur, India
vinaypn@iitk.ac.in

Abstract

In classification of objects substantial work has gone into improving the low level representation of an image by considering various aspects such as different features, a number of feature pooling and coding techniques and considering different kernels. Unlike these works, in this paper, we propose to enhance the semantic representation of an image. We aim to learn the most important visual components of an image and how they interact in order to classify the objects correctly. To achieve our objective, we propose a new latent SVM model for category level object classification. Starting from image-level annotations, we jointly learn the object class and its context in terms of spatial location (where) and appearance (what). Furthermore, to regularize the complexity of the model we learn the spatial and co-occurrence relations between adjacent regions, such that unlikely configurations are penalized. Experimental results demonstrate that the proposed method can consistently enhance results on the challenging Pascal VOC dataset in terms of classification and weakly supervised detection. We also show how semantic representation can be exploited for finding similar content.

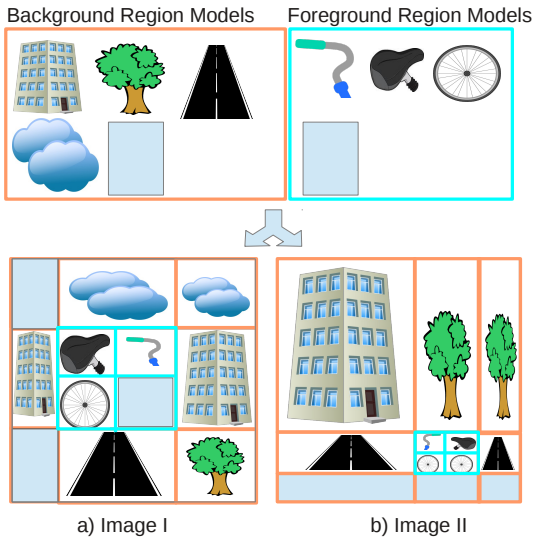

Figure 1. Overview of the proposed method. For our classification procedure we model an image as a composition of an object (foreground) enclosed in the cyan window box and the rest of the image (background). Both, foreground and background are represented by a pool of region models that are learned in a weakly supervised way. The empty blue rectangle models indicate occlusion models. For each region model and each location we learn unary and pairwise costs that reward likely configurations.

1. Introduction

In this paper, we classify objects (*e.g.* person or car) in the sense of PASCAL VOC [8], *i.e.* indicating their presence in an image, but not their spatial localization (the latter is referred to as detection in VOC parlance). There is a broad palette of classification methods, most of them focusing on a better feature representation [22] or a better feature encoding [24, 27] or a better mapping to a high-dimensional space [7]. However, very few methods have really improved the *semantic representation* of an object. We propose an improved semantic representation of an object by considering its spatial location in the scene and the multi-modality of its appearance (*i.e* intra-class variation such that instances of the same object class can vary in their shape, color, *etc.*). Also the background (context) is modeled explicitly in a multi-modal fashion. Note that the problems of representing spatial location and multi-modality of appearance especially arise in the Bag of Words (BoW) and Fisher Vector (FV) representation [19] that have been most successful for the classification problems.

At first sight, spatial location is accounted for by the BoW representation and FV through discriminative visual words (or Gaussians) that are learned and should fire only on the object. The rest of the image is ideally associated with non-discriminative words that should not contribute much to the final classification score. In the same way, multi-modal appearances should ideally be represented by the BoW and FV representation using more visual words (or Gaussians) (associating them with different appearances in the same object class).

In practice, the first problem is that the BoW and FV representation are quite sensitive to background visual noise such that visual words in the background that are not really correlated to the object, appear to be due to a limited number of samples. The second problem is that the recognition model is prone to false positives in the presence of high intra-class variation. For instance, if we want to recognize only perfectly yellow and red birds assuming those colors as our visual features, a BoW or FV representation with a linear classifier would also recognize the ones with both yellow and red color and over-generalize the bird class because they are the linear composition of the two.

Incorporating the spatial information of an object and its multi-modal appearance in a BoW representation is not straight-forward. Most of the state-of-the-art methods [27, 24, 19] still rely on a spatial pyramid (SP), which is a simple split of the image in a fixed grid of sub-regions. For each they then use a different BoW or FV model. However, this is clearly sub-optimal as it represents the image as a *static and uniformly distributed* collection of regions.

Blaschko *et al.* [3] propose a principled way to learn object localization when the locations of the objects are available in training for object detection. Kumar *et al.* [12] and Bilen *et al.* [2] use the object location as a latent variable and optimize the learning algorithm over a set of possible spatial configurations for classification tasks. This is a better representation than the SP, but it is still far from 'reality'. In real images the instances of an object class can have multiple and quite different appearances depending on the point of view, pose and specific object that are instantiated in the picture. Whereas, the method of [2] assumes that a single classification model can represent all the instances of a given object class. Furthermore, it has been shown that there is a strong correlation between the object and its background [20] (everything that is not the object of interest) and this can be used to better distinguish a certain class from the others.

In this sense, we propose an object classification method that better handles the complexity of real images by jointly learning and *localizing not only the object, but also its constituent parts as well as the background*. Similar to the 'Reconfigurable Bag of Words'(RBOW) [18], where a scene is modeled as a composition of multiple constituent parts, in this work we consider the object of interest as a composition of parts that can be placed together to better model its visual appearance (see Fig.1). It should be noted that the term 'part' does not necessarily correspond to a functional unit of an object such as wheel, handle bar of a bicycle. Furthermore, once the object (or foreground) is localized we also model the background as a composition of constituent parts. Finally, to enforce coherence in the models and better cope with appearance noise, we also learn *pairwise relationships between adjacent parts* (*e.g.* wheels next to each other for a bicycle). This permits us to avoid unlikely part configurations and therefore to avoid false positives due to 'hallucinated' recognitions.

In spite of the seemingly high complexity of the model that needs to be learned from weak supervision, we show that we can formulate the problem as an instance of a latent SVM (LSVM) [26]. It is known that learning a classifier with latent parameters is a non-convex optimization problem and thus it is quite sensitive to initialization. Considering that only the image label is given (no bounding box nor segmentation) in combination with numerous latent parameters of our expressive model, the initialization has a crucial role to proceed to an effective learning. In this paper, we propose a novel strategy to initialize the latent parameters that finds the most discriminative background groupings and improves the classification.

In summary, the main contributions of the paper are (i) an expressive object image representation that models the relationship between foreground and background regions for enhanced classification and (ii) a novel strategy to initialize the latent parameters that enables an effective learning in a weakly supervised setting. We empirically show through several experiments on PASCAL VOC 2007 [8] that the learned models improve the previous state-of-the-art and therefore may very well be a better representation of real images.

The remainder of the paper is structured as follows. Section 2 relates our method to previous work. Section 3 formulates the inference of the latent variables and the learning procedure of the latent SVM model. Section 4 discusses different initialization strategies. Section 5 describes and discusses the results on the VOC 2007 dataset [8] and section 6 concludes the paper.

2. Related Work

In the literature, numerous works [13, 16, 2, 17, 20] have explored the idea of using spatial information for object classification. Spatial pyramids [13] make use of the spatial information by dividing images into uniform regions and describe each region with a bag of words (BoW). Kumar *et al.* [12] and Bilen *et al.* [2] have shown that the choice of subregions in spatial pyramids can be further customized and optimized on image level to have better classification performance. Russakovsky *et al.* [20] propose a complementary object centric background model to boost the classification performance by using context information around the foreground. However, these approaches have limited power to deal with significant variability in appearances and views within the same object class.

Work closely related to ours is the reconfigurable bag of words (RBOW) approach [18] that models a scene as a composition of multiple constituent parts. Similarly, we also consider the object of interest and the background regions

as a composition of parts that can be placed together to better model visual appearance. Whereas the RBOW method focuses on scene classification, we tailor our method for object classification tasks. The fundamental difference between the two tasks is that in the latter one the foreground (object itself) usually has less variability in terms of appearance and includes more discriminative features than the background. Using background regions still helps to improve object classification performance but they need to be modeled separately from the foreground. We validate this claim experimentally in Section 5. This issue does not arise in scene classification, since there is no clear distinction between regions as being foreground or background. For our object classification task we propose an object centric approach. In this approach the position of the foreground regions automatically defines the background regions around it. Moreover, we enforce coherence between the adjacent foreground-foreground, background-background and foreground-background regions by learning their pairwise relationship and show that it is more robust against appearance noise. Similar to the RBOW method, Yakhnenko *et al.* [25] represent images as a collection of regions. Yet they use only two region labels to represent foreground and background, and assume that all the parts of the objects can be represented by one foreground region model, while we solve the challenging task of capturing multiple foreground and background appearances.

Context has also been used in [10, 5, 6, 1] for object detection. In [5, 6], the authors exploit the spatial interactions between object instances and in [10] the foreground-background relation is explored. Alexe *et al.* [1] propose to use context information to reduce the number of candidate object windows. These methods require bounding box annotations of the training images however, while our approach asks only for image-level class labels.

3. Inference and Learning

We want to learn a binary classifier (class *vs.* non-class) that estimates for each image the location of the foreground as well as the constituent parts of the foreground and background. The training happens in a weakly supervised setting, as only the class-label of an image is given. Only a single object of the target class and therefore a single foreground window is supposed to be present in the image.

Fig. 2 illustrates the image representation that is used in this paper. We represent each image $\mathbf{x}$ as a collection of foreground (drawn in green) and background regions (drawn in orange) (r_i). We uniformly split the region inside the given foreground window (o) (drawn in cyan) into four foreground regions $\{r_1, r_2, r_3, r_4\}$. The foreground window provides a natural split for the eight background regions $\{r_5, \ldots, r_{12}\}$ such as bottom-left, top-right, *etc.* relative to o. The spatial arrangement of all foreground and background regions can thus be parameterized through the specification of the single foreground window $o \in \mathcal{O}$, where $\mathcal{O}$ is the set of possible image windows in an image. Each region r_i is represented by a *region label* from a pool of appearances or models. We use l_i (from a discrete label set $\mathcal{L}$) to specify the selected label for the region r_i. We have two independent pools of region labels for foreground and background as illustrated in Fig. 1 and we learn a *region model* for each region label.

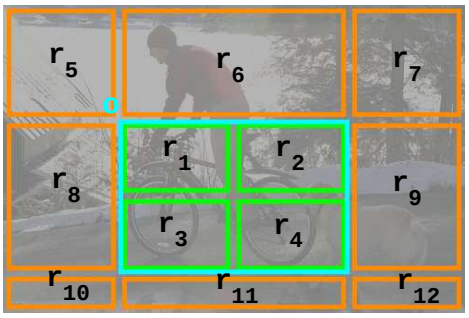

Figure 2. Diagram of a possible spatial configuration: It depicts the full configuration with all the foreground $r_1, \ldots, r_4$ and the background regions $r_5, \ldots, r_{12}$. The foreground window o, drawn in cyan, separates the foreground and background regions.

We formulate the learning problem in two steps. The first step is inference, which finds the configuration of the foreground and background regions that maximizes a scoring function. The second is the learning step, which trains a model given a set of images and their class labels. We detail the two procedures in the following sections. Note that, as we are using a discriminative setting, learning will make use of the inference step.

3.1. Inference

The inference problem of our method is to find a prediction rule that infers a class label $y \in \{-1, +1\}$ for a previously unseen image $\mathbf{x}$ using a learned discriminatively trained model parameter vector $\boldsymbol{w}$:

$$y^* = \arg\max_y f_{\boldsymbol{w}}(\mathbf{x}, y), \tag{1}$$

where $f_{\boldsymbol{w}}(\mathbf{x}, y)$ is the discriminant function trained to give a high score if the image $\mathbf{x}$ belongs to class y. Moreover, we use an image window (o) to divide an image into foreground and background regions (r_i) (Fig. 2). We also assign a region label (l_i) to each region r_i. These parameters (location of o and region labels l_i) define the configuration of the image and are considered as latent variables, because the ground truth annotation of them is not available. As we use a linear model, the discriminant function $f_{\boldsymbol{w}}(\mathbf{x}, y)$ with the latent variables $\mathbf{h}$ can be rewritten as:

$$f_{\boldsymbol{w}}(\mathbf{x}, y) = \max_{\mathbf{h}} \boldsymbol{w}^{\mathsf{T}} \boldsymbol{\psi}(\mathbf{x}, y, \mathbf{h}), \tag{2}$$

where $\psi(x, y, h)$ is the joint feature vector in the LSVM formulation [26] and $h = (o, l_1, \ldots, l_M)$ contains the configuration of latent variables. M denotes the total number of foreground and background regions in an image and is 12 in our case. For $y = -1$ we define $f_w(x, y = -1) = 0$ as in [3]. For $y = +1$, the window o and region labels l_i are obtained as the best configuration of foreground and background regions:

$$f_w(x, y = +1) = \max_{o} \sum_{i=1,\ldots,4} \max_{l_i} \left(A^{fg}_{r_i,l_i} + B^{fg\,T}_{l_i} \phi(x, o, r_i) \right) + \sum_{i=5,\ldots,12} \max_{l_i} \left(A^{bg}_{r_i,l_i} + B^{bg\,T}_{l_i} \phi(x, o, r_i) \right), \tag{3}$$

where $A^{fg}_{r_i,l_i}$ and $A^{bg}_{r_i,l_i}$ are biases that tell us how compatible the region label l_i is with the region r_i for foreground and background respectively. In the same way, $B^{fg}_{l_i}$ and $B^{bg}_{l_i}$ are the appearance parameters associated with the feature map $\phi(x, o, r_i)$ for the region label l_i for foreground and background. As the best label can be selected for each region independently, the optimization is fast and it can be done for each window location o. Note that we also include an auxiliary region label for occlusion. This label is assigned to a region, when the highest scoring label is lower than a corresponding bias.

Now, we introduce pairwise costs C_{r_i,r_j,l_i,l_j} that define the compatibility between the chosen labels l_i, l_j of adjacent regions r_i, r_j with $(i, j) \in \varepsilon$, the set of connected region pairs. The new discriminant function is:

$$f_w(x, y = +1) = \max_{o,l} \Big(\sum_{i=1,\ldots,4} A^{fg}_{r_i,l_i} + B^{fg\,T}_{l_i} \phi(x, o, r_i) + \sum_{i=5,\ldots,12} A^{bg}_{r_i,l_i} + B^{bg\,T}_{l_i} \phi(x, o, r_i) + \sum_{(i,j)\in\varepsilon} C_{r_i,r_j,l_i,l_j} \Big). \tag{4}$$

In this case, for each possible window o the selection of a region label for a certain region depends also on its neighbors. Thus, whereas in Eq.(3) we could select the label for each region independently, now the scoring function needs a global optimization over $l = \{l_1, l_2, \ldots, l_{12}\}$. To do that, we use a conditional random field (CRF) optimization for each window location o based on re-weighted tree belief propagation [11], where the CRF nodes are defined regions r_i and labels are region labels l_i. As the numbers of regions and labels are relatively small, this optimization is still quite fast.

The discriminant function (4) allows us to define the LSVM parameter vector w and the joint feature map $\psi(x, y, h)$ in Eq.(2). The learned parameter vector w is now a concatenation of the bias parameters $A^{fg}_{r_i,l_i}$, $A^{bg}_{r_i,l_i}$, the appearance parameters $B^{fg}_{l_i}$, $B^{bg}_{l_i}$ and the pairwise parameters C_{r_i,r_j,l_i,l_j}. Having the parameter vector w, we design the joint feature vector $\psi(x, h)$ for a given class y and configuration h as follows: When the class y is present $(y = 1)$ in the image, $\phi(x, o, r_i)$ is positioned at the corresponding location of the label l_i for each region r_i. When the class is not present $y = -1$ in the image x, we set all elements of the feature map vector $\psi(x, y, h)$ to zero. Note that this does not mean that the negative images are not used during training. As shown in the next section, our learning procedure enforces the highest response from a negative image to be lower than 0 and the one from a positive image to be greater than 0 with a margin.

3.2. Learning

Given a set of training samples $X = \{x_1, \ldots, x_n\}$ and their labels $Y = \{y_1, \ldots, y_n\}$, where each $y_i \in \{-1, 1\}$ $(i = 1, \ldots, n)$, we learn a linear SVM model w to predict the class label of an unseen example. We also use the latent parameters $H = \{h_1, \ldots, h_n\}$ to select the image windows o that specify the spatial configuration, and labels l that explain the resulting foreground and background regions best. The region labels l correspond to those introduced in Section 3.1. To jointly learn the SVM model and latent parameters, we follow the latent SVM formulation of [26]:

$$\min_{w} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{n} \Big[\max_{\hat{y}_i, \hat{h}_i} \left[w^T \psi(x_i, \hat{y}_i, \hat{h}_i) + \Delta(y_i, \hat{y}_i) \right] - \max_{\hat{h}_i} \left[w^T \psi(x_i, y_i, \hat{h}_i) \right] \Big] \tag{5}$$

where C is the regularization parameter and $\Delta(y_i, \hat{y}_i)$ is the $0-1$ loss function $\Delta(y_i, \hat{y}_i) = 1$ if $\hat{y}_i \neq y_i$, and else 0. We refer to [26] for more details.

4. Initialization of Latent Parameters

The success of our method relies on learning discriminative appearance models that can represent a wide range of variability in the appearance and spatial configuration of foreground and background regions. In other words, each foreground and background model should be distinctive and at the same time general enough to appear in a number of images. We use the LSVM framework to train those models and relations in a discriminative way. The BoW representation with a linear SVM model has many degrees of freedom and is thus usually able to learn a classifier with an arbitrary latent configuration and small training error by over-fitting on the training set. Here, the initialization of parameters for the optimization plays an important role to learn discriminative background and foreground models. One can set the parameter vector w or the latent parameters h_i for each sample i to initialize the optimization. In our experiments, we prefer to initialize the latent variables, since setting w with an arbitrary norm (*i.e.* $\|w\|$) may introduce stability problems or biases.

In practice we find that our optimization algorithm is more sensitive to different initialization strategies for background regions than for foreground ones. This can be explained by the fact that background regions carry more variation in appearance than foreground. Therefore, we focus on the initialization of background regions. For the foreground regions, we use a *fixed initialization* strategy by assigning a particular region label to each foreground region (*i.e.* $r_i \leftarrow l_i$ for $i \in \{1,2,3,4\}$). Moreover, we use the localization result of our model without any background regions (denoted as LOC) as in [2] to set our initial window (o) for each image. The regions inside and outside of those windows are considered as foreground and background regions respectively.

A naive initialization strategy is to assign a particular region label to each background region depending on its location, as done for the foreground regions. However, in our experiments, we found that this prevents changes in the latent parameters during the following optimization iterations. A better strategy is to train an exemplar classifier, such as exemplar SVM [15], for each background region in positive images. Then, we test the trained exemplar classifiers on the validation set, choose the most discriminating ones and use them to label background regions. However, training thousands of linear SVMs is computationally expensive. Therefore, we propose to use a simpler linear classification method, linear discriminant analysis (LDA). LDA has a shorter training time and comparable performance, as shown in [9].

In order to initialize the latent parameters in the training set, we use the following procedure:

1. Training an LDA classifier requires the computation of the covariance matrix $\mathbf{S_W}$ and mean $\mathbf{m_-}$ of negative background regions.

2. We run the LOC method on the training set to initialize foreground windows (o). This automatically defines the locations of foreground and background regions.

3. We encode each background region (r_j) in each positive image (x_i) with LLC [24], denote it as ($\mathbf{p_{ij}}$) and learn a LDA classifier θ_{ij}:

$$\theta_{ij} \propto {\mathbf{S_W}}^{-1}(\mathbf{p_{ij}} - \mathbf{m_-}). \tag{6}$$

4. In order to prevent very similar background regions to be chosen, we compute the cosine similarity angle between learned LDA classifier pairs and remove similar ones with a threshold of 0.4.

5. To pick the best N LDA classifiers, we use an SVM with a L_1 regularization which encourages sparsity among the learned weights. Briefly, we describe each background region with a K-dimensional vector that contains scores of the K learned LDA classifiers. We concatenate these vectors and obtain a $8 \times K$ descriptor for each image. We train an SVM on these features that chooses the most discriminative and independent LDA classifiers and pick the LDA classifiers with the highest absolute values.

6. We test these classifiers on background regions of positive images and assign each region to the best scoring classifier.

5. Experiments

In this section, we first give the details of the experimental benchmark and implementation. We then show and discuss the effect of each component in our model.

Dataset. We evaluate on the challenging PASCAL VOC 2007 [8]. It contains 9,963 images split into training, validation and test sets. The images are labeled with twenty classes. We learn a one-vs.-rest classifier for each class and report the average precision for each class as well as the mean AP (mAP) which is the mean of AP values from each of the classifiers.

Implementation Details. We extract dense SIFT features [14] by using the `vl_phow` function from the VLFeat toolbox [23]. We apply K-means to 200,000 randomly sampled descriptors from the training images to form the visual codebook. The computed visual words are then used to build up the descriptors using LLC coding and max pooling [24]. The codebook size is 8192. The encoded feature vectors for foreground and background are normalized to have L_2 norm 1 and 0.1 respectively. This normalization strategy forces the SVM model parameters to be regularized more strictly for the ones that correspond to background. This gives more importance to the foreground representation and it has a positive impact on the final classifier accuracy.

Spatial Pyramid. Our baseline is a BoW implementation with a $1 \times 1, 2 \times 2$ spatial pyramid with LLC coding. In Table 1 this method is denoted by `(1)` and it obtains a mAP of 54.7%. Notice that the score is obtained by using a single feature and sparse coding. Using multiple features (*e.g.* LBP, HOG) and a better encoding (*e.g.* fisher kernels) should improve the baseline as well as any row of the table because our contributions are orthogonal to those.

Localization. This configuration is a re-implementation of [2] using the 'crop-uni-split' operation which employs a bounding box with a single layer of spatial pyramid. For each image, a latent window is used to localize the object of interest. We use a coarse 8×8 grid to spatially quantize the images and this produces 1296 unique configuration for the foreground window o. The regularization parameter C is set to 10^6 for training of all classifiers. As shown in Table 1 configuration `(2)`, the latent localization of the object of interest is a fruitful strategy, and improves over the baseline

	(1)	(2)	(3)	(4)	(5)	(6)
LOC		x	x	x		x
MFG			x	x	x	x
MBG				x	x	x
CRF					x	x
mean	54.7	56.5	57.2	58.2	55.8	**59.3**
aeroplane	70.0	70.1	72.5	**76.1**	75.0	74.2
bicycle	59.6	64.0	65.3	63.5	62.6	**65.5**
bird	45.4	45.9	46.2	49.1	42.6	**50.0**
boat	64.4	66.9	66.9	**67.7**	66.6	67.2
bottle	24.8	24.6	25.3	**27.7**	24.3	26.9
bus	60.4	64.0	63.7	63.6	59.4	**65.2**
car	75.3	77.0	76.9	79.0	75.7	**80.2**
cat	57.5	59.9	58.2	60.4	58.4	**63.4**
chair	53.5	55.1	**55.6**	54.7	50.2	53.9
cow	42.9	45.4	46.0	46.4	44.2	**49.5**
diningtable	46.9	46.9	47.6	51.3	48.8	**52.4**
dog	41.2	41.7	42.0	43.4	44.9	**47.6**
horse	71.4	74.7	74.6	76.6	75.3	**77.4**
motorbike	62.7	66.2	67.0	66.5	64.8	**68.5**
person	82.4	82.5	82.6	83.3	81.5	**83.7**
pottedplant	22.5	22.7	26.7	26.9	24.7	**27.2**
sheep	43.5	44.2	44.7	44.8	43.0	**46.6**
sofa	49.6	53.8	55.1	54.5	51.5	**55.6**
train	70.9	72.6	73.2	75.2	72.2	**75.6**
tv	50.0	53.1	53.8	53.8	49.5	**54.4**

Table 1. The classification results in terms of AP on PASCAL VOC 2007 for different configurations of our method. LOC, MFG, MBG and CRF denote localization, mixture of foreground models, mixture of background models and conditional random fields respectively.

SP in most of the categories. This method increases the mAP over the SP by around 2 points.

Multiple Appearances. In Table 1, configurations (3) and (4) correspond to our multiple models for foreground and background respectively. The introduction of multiple models for both foreground and background result in a similar improvement of around 1 point each. For background we use the initialization based on LDA as explained in Section 4. In our preliminary experiments we have noticed that the best performance is obtained by using the same number of models as the number regions. Thus we learn 4 foreground models and 8 background models.

Pairwise Compatibility. On top of the previous configuration we add the pairwise costs defined in Section 3 and denote this setting as (6) in Table 1. These additional costs enforce coherency between adjacent regions and therefore help to produce a more consistent representation of the scene. The overall benefit of the pairwise costs is 1.1 percent and certain classes show a substantial improvement (*e.g.* bicycle +2.0, cat +3.0, cow +3.1, dog +4.2, motorbike +2.0, sheep +1.8).

We also evaluate the effect of modeling foreground and background separately without localizing the foreground. A similar setting has been used in [18] for a scene classification task where there is no distinction between foreground and background. In practice, for this configuration (denoted as (5) in Table 1) we use a fixed window o at the center of the images to divide the image into 9 equal regions. We let each of these regions (r_i) to choose the best region label (l_i) considering also the pairwise constraints. In this case the mAP is 55.8% and the increment with respect to the SP is only 1.4, whereas with localization the increment is 5.2 points. This indicates that the localization of the object of interest also helps to produce better region models and it is therefore crucial for a good object classification system. We also visualize the estimated latent variables for the full configuration in Fig. 4 and show that we can obtain semantically meaningful results.

Latent Initialization. We compare the initialization strategy based on LDA, which is explained in Section 4, with the fixed initialization, where each region r_i depending on its index value (i) is assigned to a label l_i. In case of fixed initialization the number of models should be equal to the number of regions. Therefore, in order to provide a fair comparison, we also use 4 foreground and 8 background models (with the pairwise connections in both settings) for the LDA. We obtain 58.1% mAP for the fixed initialization, while the LDA based strategy achieves 59.3% (Table 1, configuration (6)) with a net improvement of 1.2.

To further analyze our initialization strategy and latent learning, we set an additional baseline experiment by initializing the foreground windows with the ground truth bounding boxes. We adapt the annotated boxes to an 8×8 grid by quantizing their coordinates. In the case of multiple instances of the same object in an image, we pick the one with the bigger area. Initializing the foreground window with the ground truth ones achieves 59.5% mAP and improves only 0.2% over the weakly supervised case. This shows that our classifiers achieve a comparably good performance and learn well with latent localization.

Comparison to Similar Methods. We also compare our best configuration, (6) to the reported results of the most related work [4] (LLC(25k)) and [20] (OCP) that also use a single type of local feature, SIFT and DHOG respectively. The first column (LLC(25k)) of Table 2 shows the result obtained by using a SP, LLC encoding with 25,000 visual words and approximated chi square kernel and a $1 \times 1, 2 \times 2, 3 \times 1$ spatial pyramid. Even though this setting uses a bigger codebook and a non-linear kernel, our model is still better. The second column (OCP) reports the results of [20]. This method also localizes the object of interest and represents the background. However, it is still 2.1 points below our best configuration. This shows that using multiple models and pairwise costs really helps to boost classification.

Image Retrieval with Semantic Similarity. In addition to inferring a class label, our method also divides the image into regions and assigns a label to each image region. We claim that these labels provide a coarse semantic level

method	mAP	plane	bicycle	bird	boat	bottle	bus	car	cat	chair	cow
LLC(25k) [4]	57.7	72.4	62.2	47.3	**68.9**	25.8	64.0	77.3	59.8	54.3	46.0
OCP [20]	57.2	**74.2**	63.1	45.1	65.9	**29.5**	64.7	79.2	61.4	51.0	45.0
Our Method (6)	**59.3**	**74.2**	**65.5**	**50.0**	67.2	26.9	**65.2**	**80.2**	**63.4**	**53.9**	**49.5**

method		table	dog	horse	mbike	person	plant	sheep	sofa	train	tv
LLC(25k)[4]		51.1	43.2	76.7	67.1	83.5	**27.7**	44.9	52.8	**76.0**	52.5
OCP[20]		**54.8**	45.4	76.3	67.1	**84.4**	21.8	44.3	48.8	70.7	51.7
Our Method (6)		52.4	**47.6**	**77.4**	**68.5**	83.7	27.2	**46.6**	**55.6**	75.6	**54.4**

Table 2. Comparison with the most related published results on the PASCAL VOC 2007, LLC (25k) [4] and OCP [20]. Our method outperforms other related methods in most of the classes and also in mean average precision.

representation of the image. In this part we test our method on retrieval of similar content for a given query image on VOC07. We show preliminary qualitative results in Fig. 3. We run our classifiers trained for the configuration (6) on the test images of VOC07 dataset and use the inferred foreground and background region labels to describe each image. We randomly pick a query image and compute the Hamming distance between the labels of the query and the test images and rank their distance to the query. As a baseline we evaluate a SP representation only on the images that are from class of the query. We use the cosine similarity between the two normalized histograms to rank related images. Fig. 3 shows that ranking based on SP yields a rather random selection of images from the class, while the ranking based on our method seems to make more sense.

Weakly Supervised Detection. Our method outputs, together with the class label, the location of the latent window o that best describes the class. Could this additional information be used for detection? In our common configuration the classifiers use a rough localization (*i.e.* a grid with 8×8 cells with a minimum 2×2 foreground size) that ensures good classification results while being computationally efficient, but it is too coarse for accurate localization (*e.g.* 50% intersection over union). To arrive at more accurate localization, we replace the coarse windows generated by the grid with a set of windows based on [22]. We use a finer subdivision into 3×3 foreground regions since a finer grid leads to better localization.

We train our detectors on those windows for two settings, LOC and our full model (6). While LOC obtains 15.6% mAP on the testing split of the VOC07 dataset, config. (6) yields 16.6% mAP *i.e.* a net improvement of 1%. This shows that our method cannot only improve classification, but also detection. With these results, we also outperform [21, 20] who report weakly supervised detection results on Pascal VOC 2007 of 15.0% and 13.9% respectively.

Computational Cost. The running time of the LSVM experiments is dominated by inference of the best configuration for each image (*e.g.* including all possible windows and all possible background models). The training of each class-specific classifier for the full configuration (6) on the VOC 2007 dataset took 4 hours on a 12 CPU machine. The typical inference time for an image is 5 seconds.

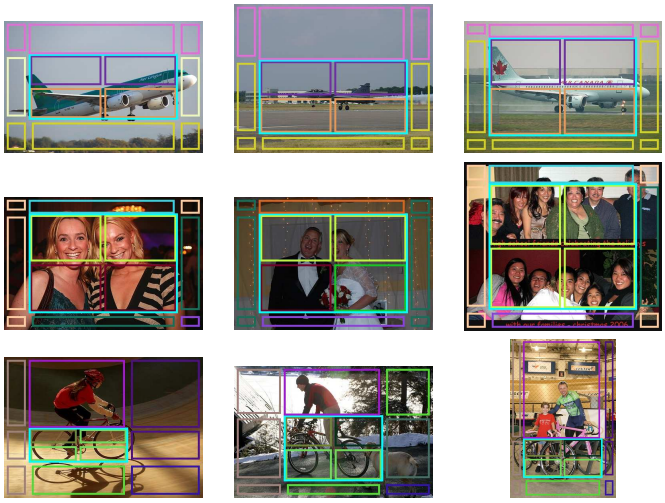

Figure 4. Examples of latent variables for different classes of VOC07. The cyan bounding box represents the localized object of interest. The other bounding boxes represent the different regions of the image for foreground (inside the object of interest) and background (outside the object of interest). For a certain class, the color of the bounding box represents the inferred region model. Thus, same color means same region model. The examples in the first row show that 'sky' and 'ground' background regions are consistently labeled with a particular model. In the second row, faces and upper body of people are assigned to different foreground models. In the last row, as 'bicycle' is the class of interest, people in the images are assigned to a background region label (l_i).

6. Conclusion

In this paper we have introduced a new semantic representation of an image based on latent variables that can improve the object classification accuracy without requiring any additional annotation. With an incremental evaluation of each characteristic of our model on the challenging Pascal VOC 2007, we have shown that localizing the object of interest in the image is important as well as properly representing the multi-modal appearance of the object and its background. Furthermore, additional accuracy can be obtained by learning and enforcing pairwise costs between the neighboring regions. Altogether our model is able to achieve a gain of 4.6 points over the standard SP without any additional images, low-level features or annotations.

Acknowledgments: This work is supported by the EU Project FP7 AXES ICT-269980.

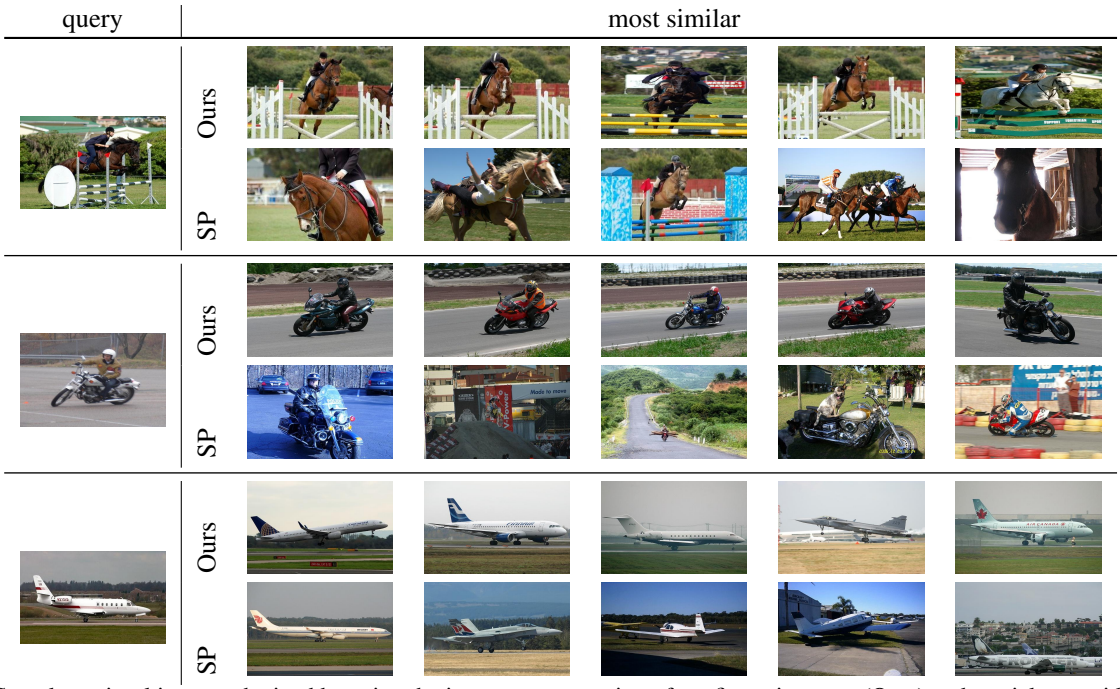

Figure 3. Sample retrieval images obtained by using the image representation of configuration (6) (Ours) and spatial pyramid (SP). Our method makes use of the latent labels assigned at inference to retrieve semantically similar images.

References

[1] B. Alexe, N. Heess, Y. W. Teh, and V. Ferrari. Searching for objects driven by context. In *NIPS*, pages 890–898, 2012. 3

[2] H. Bilen, V. P. Namboodiri, and L. J. Van Gool. Object and action classification with latent window parameters. *IJCV*, pages 1–15, 2013. 2, 5

[3] M. Blaschko and C. H. Lampert. Learning to localize objects with structured output regression. In *ECCV*, 2008. 2, 4

[4] K. Chatfield, V. Lempitsky, A. Vedaldi, and A. Zisserman. The devil is in the details: an evaluation of recent feature encoding methods. In *BMVC*, 2011. 6, 7

[5] M. J. Choi, J. J. Lim, A. Torralba, and A. S. Willsky. Exploiting hierarchical context on a large database of object categories. In *CVPR 2010*, pages 129–136, 2010. 3

[6] C. Desai, D. Ramanan, and C. Fowlkes. Discriminative models for multi-class object layout. In *ICCV 2009*, pages 229–236, 2009. 3

[7] O. Duchenne, A. Joulin, and J. Ponce. A graph-matching kernel for object categorization. In *ICCV 2011*, pages 1792–1799, 2011. 1

[8] M. Everingham, A. Zisserman, C. K. I. Williams, and L. Van Gool. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. 1, 2, 5

[9] B. Hariharan, J. Malik, and D. Ramanan. Discriminative decorrelation for clustering and classification. In *ECCV*, pages 459–472. 2012. 5

[10] G. Heitz and D. Koller. Learning spatial context: Using stuff to find things. In *ECCV 2008*, pages 30–43. 2008. 3

[11] V. Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *TPAMI*, 28(10):1568–1583, 2006. 4

[12] M. P. Kumar, B. Packer, and D. Koller. Self-paced learning for latent variable models. In *NIPS*, pages 1189–1197, 2010. 2

[13] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, pages 2169–2178, 2006. 2

[14] D. Lowe. Object recognition from local scale-invariant features. In *ICCV*, page 1150, 1999. 5

[15] T. Malisiewicz, A. Gupta, and A. A. Efros. Ensemble of exemplar-svms for object detection and beyond. In *ICCV*, 2011. 5

[16] M. H. Nguyen, L. Torresani, F. De la Torre, and C. Rother. Weakly supervised discriminative localization and classification: a joint learning process. In *ICCV*, 2009. 2

[17] M. Pandey and S. Lazebnik. Scene recognition and weakly supervised object localization with deformable part-based models. In *ICCV*, pages 1307–1314, 2011. 2

[18] S. N. Parizi, J. G. Oberlin, and P. F. Felzenszwalb. Reconfigurable models for scene recognition. In *CVPR*, pages 2775–2782, 2012. 2, 6

[19] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *ECCV*, pages 143–156, 2010. 1, 2

[20] O. Russakovsky, Y. Lin, K. Yu, and L. Fei-Fei. Object-centric spatial pooling for image classification. In *ECCV 2012*, pages 1–15. 2012. 2, 6, 7

[21] P. Siva and T. Xiang. Weakly supervised object detector learning with model drift detection. In *ICCV*, pages 343–350, 2011. 7

[22] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. Evaluating color descriptors for object and scene recognition. *TPAMI*, 32(9):1582–1596, 2010. 1, 7

[23] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. http://www.vlfeat.org/, 2008. 5

[24] J. Wang, J. Yang, K. Yu, F. Lv, T. S. Huang, and Y. Gong. Locality-constrained linear coding for image classification. In *CVPR*, pages 3360–3367, 2010. 1, 2, 5

[25] O. Yakhnenko, J. Verbeek, and C. Schmid. Region-Based Image Classification with a Latent SVM Model. Research Report RR-7665, INRIA, July 2011. 3

[26] C.-N. J. Yu and T. Joachims. Learning structural svms with latent variables. In *ICML*, pages 1169–1176, 2009. 2, 4

[27] X. Zhou, K. Yu, T. Zhang, and T. S. Huang. Image classification using super-vector coding of local image descriptors. In *ECCV 2010*, pages 141–154. 2010. 1, 2