Weakly Supervised Object Localization with Large Fisher Vectors

Josip Krapac and Siniša Šegvić
Faculty of Electrical Engineering and Computing, University of Zagreb, Zagreb, Croatia

Keywords: Weakly Supervised Object Localization, Fisher Vectors, Sparse Classification Models.

Abstract: We propose a novel method for learning object localization models in a weakly supervised manner, by employing images annotated with object class labels but not with object locations. Given an image, the learned model predicts both the presence of the object class in the image and the bounding box that determines the object location. The main ingredients of our method are a large Fisher vector representation and a sparse classification model enabling efficient evaluation of patch scores. The method is able to reliably detect very small objects with some intra-class variation in reasonable time. Experimental validation has been performed on a public dataset and we report localization performance comparable to strongly supervised approaches.

1 INTRODUCTION

Detecting the presence of objects in images and recovering their locations are often jointly addressed by applying a trained binary classifier at many image locations, and by reporting objects where a positive response was obtained. Most successful representatives of this approach (Viola and Jones, 2004; Dalal and Triggs, 2005; Lampert et al., 2009; Felzenszwalb et al., 2010; Cinbis et al., 2013) employ strong supervision at the training stage. These methods require that each training image is annotated with information about the object location and class. However, annotating object locations is expensive due to significant human labeling effort involved, even if a simple location model is used (*e.g.* bounding box). This is especially the case in realistic scenarios when thousands of annotations are required to achieve the top performance (Munder and Gavrila, 2006). Annotation is particularly difficult when the objects of interest are small, since near to pixel-level annotation accuracy may be required for best results.

In order to alleviate the effort of full annotation, several recent papers have tried to solve the object localization problem in a weakly-supervised manner (Galleguillos et al., 2008; Siva and Xiang, 2011; Deselaers et al., 2012; Nguyen et al., 2014; Cinbis et al., 2014). In this setting, the training images are annotated only with class labels. The training procedure is supposed to discover the object locations and train the classifier at the same time. At the test time, however, bounding boxes have to be predicted for each learned object class as in the strongly supervised case. This can be useful even if the recovered object classifier is not particularly fast, since the recovered localization can be used to train a more efficient localization model in a strongly supervised fashion (Chum and Zisserman, 2007).

Weakly supervised training of object classifiers is a daunting task in most realistic scenarios. If we assume 1000 positive training images and 80000 patches per image, an exhaustive search for object locations would have to consider 80000^{1000} hypotheses. One way to decrease this complexity would be to avoid checking all patches in positive images by sampling (Crandall and Huttenlocher, 2006; Crowley and Zisserman, 2013), clustering (Chum and Zisserman, 2007) or employing bottom-up location proposals based on trained segmentation (Galleguillos et al., 2008; Cinbis et al., 2013; Cinbis et al., 2014) or objectness cues (Siva and Xiang, 2011; Deselaers et al., 2012). However all these approaches risk to miss some true object patches at the selection stage, which may invalidate all subsequent efforts.

A more conservative approach relies on classifiers able to detect the object presence in a larger image context. Such classifiers can be trained on positive images (Nguyen et al., 2014) or image regions (Galleguillos et al., 2008) and then subsequently applied to recover or gradually improve the object localization. Much of the previous work along these lines (Galleguillos et al., 2008; Nguyen et al., 2014) has been based on BoW histograms (Sivic and Zisserman, 2003; Csurka et al., 2004) which do not achieve state

Krapac J. and Šegvić S..
Weakly Supervised Object Localization with Large Fisher Vectors.
DOI: 10.5220/0005294900440053
In *Proceedings of the 10th International Conference on Computer Vision Theory and Applications* (VISAPP-2015), pages 44-53
ISBN: 978-989-758-090-1

of the art image classification performance (Sánchez et al., 2013), especially on datasets with small distinctive details (Gosselin et al., 2013). Recently, Cinbis et al. (Cinbis et al., 2014) have proposed an approach based on Fisher vector representation which still requires bottom-up location proposals in order to keep the computations tractable.

In this paper we present a novel weakly-supervised object localization method based on large Fisher vectors. The presented method does not require any bottom-up location proposals and succeeds to achieve a high localization performance in experiments on very small objects (traffic signs). We argue that a Fisher vector representation without non-linear normalizations (power, metric) (Sánchez et al., 2013) is especially well-suited for localizing small objects (the *needle in a haystack* scenario (Chum et al., 2009)) due to its ability to preserve unusual details (section 3). In order to alleviate computational complexity caused by a huge dimensionality of large Fisher vectors, we select a subset of Fisher vector representation capable to identify discriminative parts of the object class by training a sparse linear classification model (section 4). The resulting classifier is applied at all image locations and the spatial layout of highly scored patches is used to determine the bounding boxes for the detected objects (section 5). This is similar to the sliding window image traversal, but much more efficient due to optimizations which take advantage of the model sparsity (section 6). The proposed method is experimentally validated on a public dataset containing very small objects with some intra-class variation, in front of information-abundant background as illustrated in Figure 4. We demonstrate fair localization performance, comparable to strongly supervised approaches, by evaluating patch scores using only a fraction (64/1024) of Fisher vector representation of the patch (section 7).

2 RELATED WORK

Many previous approaches to weakly supervised localization adopt the following basic structure: i) bottom-up initialization of object locations in positive images, ii) iterative successive improving of classification and localization models. The second step typically optimizes a criterion that at least one (Galleguillos et al., 2008; Crowley and Zisserman, 2013) (or exactly one (Chum and Zisserman, 2007; Siva and Xiang, 2011; Deselaers et al., 2012; Nguyen et al., 2014)) object is found in each positive image and that no objects are found in negative images. This optimization can be viewed as a kind of multiple-instance learning (Auer, 1997; Andrews et al., 2002) (MIL). In general, MIL implies training a binary classifier on *bags* of instances, such that positive bags contain at least one positive feature while negative bags contain all negative features.

Crandall et al. (Crandall and Huttenlocher, 2006) take a random sample of patch descriptors (n=100000) from training images and initializes the training with the most discriminative subset (n=10000). Several part-based models are then initialized from descriptor pairs and subsequently optimized through EM. Crowley et al. (Crowley and Zisserman, 2013) search for an initial set of similar descriptors with one-shot classifiers trained on random patches, while further refinement is performed through MIL. Chum et al. (Chum and Zisserman, 2007) avoid random initialization by starting from visual words of a BoW representation and proceed in the MIL fashion. Random initialization can also be avoided by filtering patch candidates in positive images. Galleguillos et al. (Galleguillos et al., 2008) propose to consider regions obtained by multiple bottom-up segmentations. Each region is represented as a BoW histogram, and a boosted classifier is constructed by repeatedly minimizing the classification loss in a MIL fashion. Deselaers et al. (Deselaers et al., 2012) apply a trained generic object detector (Alexe et al., 2010) to guide initialization of 100 random samples in each training image. By assuming that there is only one object in each positive image they train a CRF which simultaneously optimizes object locations and the classification model. Siva et al. (Siva and Xiang, 2011) propose a related approach which focuses on capturing multi-modality of object appearance. Although some of these approaches are more advanced than the others, all of them may completely miss small objects at the initialization step. Due to that, MIL refinement may easily get trapped in a local optimum (as confirmed by our preliminary experiments), and the training is likely to fail. Additionally, MIL optimization is computationally very intensive, so that training on large datasets is not feasible.

Several approaches (Pandey and Lazebnik, 2011; Nguyen et al., 2014; Cinbis et al., 2014) initialize positive object locations to entire (or almost entire) positive images and then attempt to gradually zoom into correct locations through iteration. One way to formulate this iteration is to represent object locations as latent variables in a deformable part model framework (Pandey and Lazebnik, 2011). Another approach would be to construct an integral image of the patch scores and to rely on branch-and-bound techniques in order to find regions which maximize score for the current classification model (Nguyen et al., 2014). Both of these approaches do not require bottom-up location proposals, however they are prone to convergence issues, while not being able to handle training images with multiple objects. Finally, this iteration can also be expressed in terms of bottom-up location proposals as proposed in (Cinbis et al., 2014). In their approach, the first classification model is trained on Fisher vectors of entire positive and negative images. In each subsequent iteration the negative locations are chosen as (false) positives of the current classification model on the negative training dataset. On the other hand, the positive locations are set to the top-scored bottom-up location proposals. A care has been taken to avoid a bias towards the locations from the last iteration by performing the training and selection steps on different folds of the training set of positive images (this procedure is called multi-fold MIL learning). This approach currently achieves state-of-the-art mAP PASCAL visual object classes (VOC) 2007 localization challenge performance of 22.4%.

The method proposed in this paper also harnesses the Fisher vector representation for weakly supervised object localization. However, unlike (Cinbis et al., 2014), our method identifies the most distinctive parts of the object class by directly applying a sparse classification model at the patch level[1]. The main advantage with respect to the majority of other weakly supervised localization approaches is that we do not require bottom-up location proposals. Our method is therefore able to target object classes which do not receive sufficiently accurate (e.g. 50% IoU) bottom-up location proposals, which may happen due to small size or cluttered environment. Additionally, the capability to be applied at the patch level also entails a potential to achieve high detection performance (over 80% AP on our dataset).

In comparison with previous approaches (Nguyen et al., 2014; Chum and Zisserman, 2007) which also avoid bottom-up proposals, our method is based on Fisher vectors as a superior image representation model (Sánchez et al., 2013). Thus, our method may succeed even when the number of BoW components is not large enough to ensure that distinctive patches get represented by a dedicated component (cf. Figure 1). Additionally, we do not use efficient subwindow search (Lampert et al., 2009) to localize windows which maximize overall patch score (Nguyen et al., 2014), since the background clutter often obstructs that approach to the point of producing bounding boxes several times larger than the object. A major obstacle towards making our method feasible was to keep the computational complexity tractable with respect to the dimensionality of image representation (in our case the Fisher vectors are about 165 times larger than BoW histograms for the same number of visual words). We succeeded to achieve that by reinforcing a sparse patch classification model.

3 FISHER VECTOR IMAGE REPRESENTATION

Fisher vectors can be viewed as a way to embed data points (*e.g.* patch feature vectors) into a higher-dimensional vector space. This embedding has a desirable property that the data points which are related *w.r.t.* the generative process become close in the embedded space. Thus one can build advanced discriminative models which achieve improved performance thanks to the knowledge of the data distribution (Jaakkola and Haussler, 1998).

Let the parametric generative model be given with θ, and let the pdf of a data point $\boldsymbol{x}$ *w.r.t.* to generative model be $p(\boldsymbol{x}|\theta)$. Consider now the *score function* (gradient of the log-likelihood) given with:

$$\mathrm{U}(\boldsymbol{x}|\theta) = \nabla_\theta \log p(\boldsymbol{x}|\theta)\,. \tag{1}$$

The score $\mathrm{U}(\theta,\boldsymbol{x})$ succinctly describes the relation of the data point *w.r.t.* the parameters of the generative model. Consequently, the data embedded in the score space may be easier to separate using a linear classifier, since dot product in score space corresponds to a non-linear kernel in the original space. By decorrelating components of the score, we obtain the Fisher vector $\Phi(\boldsymbol{x}|\theta)$ of the data point $\boldsymbol{x}$ given the generative model θ:

$$\begin{aligned} \Phi(\boldsymbol{x}|\theta) =& \mathbf{F}(\theta)^{-0.5} \cdot \mathrm{U}(\boldsymbol{x}|\theta)\,, \\ & \mathbf{F}(\theta) = \mathrm{E}_{\boldsymbol{x}}[\mathrm{U}(\boldsymbol{x}|\theta)\mathrm{U}^\top(\boldsymbol{x}|\theta)]\,. \end{aligned} \tag{2}$$

The covariance matrix $\mathbf{F}(\theta)$ is often referred to as the Fisher information matrix. Multiplying $\mathrm{U}(\theta,\boldsymbol{x})$ by $\mathbf{F}(\theta)^{-0.5}$ is also known as linear normalization. Fisher vectors have the following properties:

1. vanishing expectation: $\mathrm{E}_{\boldsymbol{x}}[\Phi(\boldsymbol{x}|\theta)] = 0$,
2. unit covariance: $\mathrm{E}_{\boldsymbol{x}}[\Phi(\boldsymbol{x}|\theta)\Phi^\top(\boldsymbol{x}|\theta)] = \mathbf{I}$,
3. additivity for $\boldsymbol{x}_i$ i.i.d.: $\Phi(\{\boldsymbol{x}_i\}|\theta) = \sum_i \Phi(\boldsymbol{x}_i|\theta)$.

Most classification approaches represent images with a set of i.i.d. D-dimensional patch descriptors (Lowe, 2004) which code the patch appearance. Assume that a generative model for these descriptors is given as $\theta = \{\alpha_i, \boldsymbol{\mu}_i, \boldsymbol{\sigma}_i\}_{i=1}^K$, that is, as a Gaussian mixture model (GMM) of K components with diagonal

[1] A similar idea has been explored in (Chen et al., 2013), however they address a strongly supervised localization and do not consider sparse classification models.

covariance matrices (Sánchez et al., 2013). Then the associated pdf is given by:

$$p(\boldsymbol{x}|\theta) = \sum_{i=1}^{K} w_i \cdot p(\boldsymbol{x}|\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i)\,, \quad w_i = \frac{e^{\alpha_i}}{\sum_j e^{\alpha_j}}\,. \tag{3}$$

The *responsibility* of i-th GMM component for generating the data point $\boldsymbol{x}$ is now given by:

$$\mathrm{P}(i|\boldsymbol{x}) = \frac{\mathrm{P}(i) \cdot p(\boldsymbol{x}|i)}{p(\boldsymbol{x})} = \frac{w_i \cdot p(\boldsymbol{x}|\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i)}{p(\boldsymbol{x}|\theta)}\,. \tag{4}$$

Finally, the Fisher vector elements corresponding to the i-th GMM component are (Sánchez et al., 2013):

$$\Phi_{\alpha_i}(\boldsymbol{x}|\theta) = \frac{\mathrm{P}(i|\boldsymbol{x}) - w_i}{\sqrt{w_i}}\,, \tag{5}$$

$$\Phi_{\boldsymbol{\mu}_i}(\boldsymbol{x}|\theta) = \frac{\mathrm{P}(i|\boldsymbol{x})}{\sqrt{w_i}} \cdot \frac{\boldsymbol{x} - \boldsymbol{\mu}_i}{\boldsymbol{\sigma}_i}\,, \tag{6}$$

$$\Phi_{\boldsymbol{\sigma}_i}(\boldsymbol{x}|\theta) = \frac{\mathrm{P}(i|\boldsymbol{x})}{\sqrt{2w_i}} \cdot \left[\frac{(\boldsymbol{x} - \boldsymbol{\mu}_i)^2}{\boldsymbol{\sigma}_i^2} - 1\right]. \tag{7}$$

If the GMM parameters $\boldsymbol{\mu}_i$ and $\boldsymbol{\sigma}_i$ have D dimensions, then the Fisher vectors $\Phi(\boldsymbol{x}|\theta)$ will have $K \cdot (1 + 2 \cdot D)$ elements. Due to additivity, the Fisher vector of an image $\boldsymbol{X}$ corresponds to the sum of the Fisher vectors of patches $\boldsymbol{x_i}$.

$$\Phi(\boldsymbol{X}|\theta) = \sum_i \Phi(\boldsymbol{x}_i|\theta)\,. \tag{8}$$

The main quality of this image representation model is that the contribution of ordinary patches (*i.e.* patches which are well-represented by the generative model) cancels out due to vanishing expectation. Consequently, ordinary image content is attenuated, and different portions of the Fisher vector reflect various kinds of extraordinary image regions. Other image representation models such as BoW (Sivic and Zisserman, 2003) are unable to amplify extraordinary features, and this is the main reason why Fisher vectors achieve state of the art results in recognition of small but distinctive image content.

We try to illustrate these points in Figure 1. Due to large variety of traffic scenes, patches at triangular traffic signs are typically not represented by a dedicated component of the corresponding visual dictionary. Hence, their cluster is located at the periphery of a larger GMM component in the high-dimensional feature space. These patches generate large and characteristic contributions to the gradients (6) and (7) with respect to the closest GMM component. These contributions result in characteristic deviations in the Fisher vector of the whole image (8) which can be identified by a sparse linear classification model. Our experiments clearly show that the learned classification model can be employed at the patch level to distinguish object patches from the background.

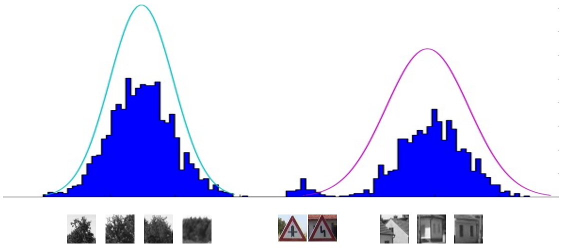

Figure 1: Traffic signs typically do not get represented by a dedicated component of a generative GMM since they are very small with respect to the dimensions of typical images acquired from the driver's perspective. Therefore, their patches produce large gradients (6) with respect to the closest GMM component, and generate a characteristic contribution to the Fisher vector of the image (8).

4 SPARSE CLASSIFICATION MODEL

We consider weakly-supervised localization of object classes with small intra-class variation and assume that discriminative object parts are contained in small parts of the Fisher vector space. To select these discriminative parts of image representation we learn a sparse linear model $\boldsymbol{w}$, *i.e.* a model in which the majority of coefficients is zero. The model $\boldsymbol{w}$ is learned by minimizing a regularized loss function on a set of N training images, where each image $\boldsymbol{X}_i$ is annotated with the corresponding label y_i:

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} \sum_{i=1}^{N} \ell(\boldsymbol{w}, \Phi(\boldsymbol{X}_i), y_i) + \lambda \cdot \mathcal{R}(\boldsymbol{w}) \tag{9}$$

The choice of loss function $\ell(\cdot,\cdot,\cdot)$ and model regularizer $\mathcal{R}(\cdot)$ determines the model class, while λ regulates the trade-off between the loss and the regularizer. By supplying the logistic loss and L_1-regularization, we obtain a sparse logistic regression model in which the sparsity is determined by λ:

$$\ell(\boldsymbol{w}, \Phi(\boldsymbol{X}_i), y_i) = \log(1 + \exp(-y \cdot \boldsymbol{w}^\top \Phi(\boldsymbol{X}_i)))\,,$$
$$\mathcal{R}(\boldsymbol{w}) = ||\boldsymbol{w}||_1\,. \tag{10}$$

We avoid non-linear normalizations of the image representation $\Phi(\boldsymbol{X})$ in order to preserve additivity of the learned model[2], so the image score s can be expressed as a sum of patch scores s_i:

$$s = \boldsymbol{w}^\top \Phi(\boldsymbol{X}) = \boldsymbol{w}^\top \sum_i \Phi(\boldsymbol{x}_i) = \sum_i \boldsymbol{w}^\top \Phi(\boldsymbol{x}_i)$$
$$= \sum_i s_i\,. \tag{11}$$

[2] In particular, power and metric normalizations (Perronnin et al., 2010) would imply $\Phi(\boldsymbol{X}) \neq \sum_i \Phi(\boldsymbol{x}_i)$.

Therefore, the model $\boldsymbol{w}$ can be directly applied to score image patches, although it has been learned on Fisher vectors of entire images.

This procedure resembles MIL. Our Fisher vector representation embeds a set of patches into a vector space, and we learn the discriminative model assuming that some patches in positive images have a positive label, while all patches in the negative images have negative labels. However, rather than explicitly removing or relabeling the patches in positive images as in multiple instance SVM approaches (Andrews et al., 2002) (which would be computationally prohibitive), our sparse model instead performs selection and weighting of Fisher vector coefficients and thus induces the ranking for each image and each patch.

The classification model that performs well is able to select image patches which are relevant for the class, and therefore the top ranked patches according to the model scores are likely to belong to the object.

5 FROM PATCH SCORES TO OBJECTS

We wish to recover locations of multiple objects by analyzing model scores of image patches. To achieve that goal, we select top T ranked patches in the input image and use their spatial layout to define bounding boxes. Specifically, we form a spatial graph of top ranked patches, where nodes correspond to patches while the connectivity is determined via patch overlap. We extract connected components from the spatial graph, and discard components with small number of patches. Each of the remaining components determines a bounding box as a union of the associated patches, while the bounding box score is set to the average patch score.

6 EFFICIENT PATCH SCORE COMPUTATION

We reduce the complexity of patch score computation by exploiting two kinds of sparsity. The soft-assign sparsity refers to the fact that the GMM posterior (4) is very sparse: a majority of patches are dominantly assigned to only one GMM component. This effect is especially pronounced for large Fisher vectors which we intend to use. The model sparsity indicates that, our classification model (due to L1 regularization) typically does not reference all parts of the image representation vector. Although we use GMM with $K = 1024$ components, non-zero model coefficients correspond to only 479 GMM components and only a few of them dominantly contribute to the patch score.

To obtain the score for a patch we have to i) compute its Fisher vector and ii) project it onto the learned discriminative model $\boldsymbol{w}$. However, due to soft-assign sparsity most of the Fisher vector elements will be zero. Thus, for each patch we first compute all soft-assigns (4) and subsequently compute the Fisher vector elements only for the top K_{SA} GMM components (*cf.* Figure 2, top). Hence, we compute the score by evaluating only a fraction of the full generative model, and consequently achieve a speedup of K/K_{SA} in score computation. We call this procedure the *soft-assign sparsity* optimization.

Since we are interested only in top ranked patches we can additionally exploit the sparsity of the discriminative model. We select top K_W components of the GMM, according to the L_1 norm of the model portion that corresponds to a GMM component (*cf.* Figure 2, bottom). This enables us to efficiently discard patches which are not likely to be among the top ones, because a patch can only have a high score if it is dominantly assigned to some of the top GMM components. We call this procedure the *model sparsity* optimization. For images in which many patches are dominantly assigned to GMM components that do not belong to a set of top K_W components this approximation results in considerable speedup, since the score needs to be computed for only a fraction of patches in these images.

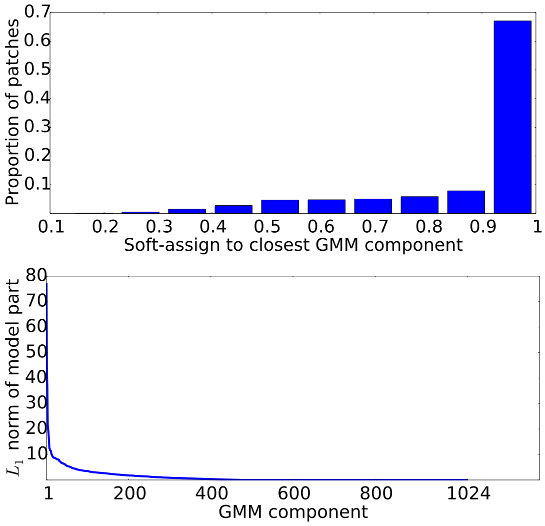

Figure 2: Two kinds of sparsity which we exploit in efficient computation of the patch scores. *Top*: the soft-assign sparsity. *Bottom*: the model sparsity.

7 EXPERIMENTAL EVALUATION

Dataset. We use the dataset MASTIF TS2010[3] which contains 3296 images extracted from a video sequence recorded from a moving vehicle for the purpose of traffic sign inspection. Each image contains at least one traffic sign, and each traffic sign is annotated with a groundtruth bounding box and a class label. Images are also annotated with track labels which denote their temporal position in video. The dataset is divided into the train and the test split in a way that images of particular physical signs are always assigned to the same split.

We evaluate the proposed approach on European triangular warning signs. This superclass includes around thirty individual classes such as "road hump" (*cf.* Figure 4, top left) or "pedestrian crossing" (*cf.* Figure 4, bottom left). There are 1705 images in the train split, 453 of which contain warning signs. The test split consists of 1591 images, including 379 images with one warning sign and 60 images with two traffic signs. Thus the test split contains a total of 499 warning sign instances.

Our dataset is considerably different from most popular object localization datasets. First, our objects are small compared to the image size. The average size of the traffic sign bounding boxes is 49.66×48.34 (±23.08×22.82). Since the resolution of all images is 720×576, our objects usually cover less than 1% of image pixels. Second, the context is not very informative for classification and localization: positive and negative images have almost identical backgrounds. We therefore believe that this localization problem, especially in the weakly-supervised setting, deserves attention from the vision community despite the relative simplicity of the object class.

Performance Measure. We evaluate the performance by using the precision-recall curve and average precision (AP), as proposed in Pascal VOC (Everingham et al., 2010). The localization accuracy is defined in terms of overlap with the groundtruth bounding box measured as intersection over union. We attempt to remove multiple detections of the same object by accepting only the bounding box with the highest score among the ones that overlap more than 50%. A detected bounding box is counted as a positive detection if it overlaps with the ground truth bounding box more than 50%. This is a quite high threshold, considering weakly-supervised localization and especially the small size of the objects.

Implementation Details. We extract dense SIFT features using the VLFeat library (Vedaldi and Fulkerson, 2008) from patches with square spatial bins of sizes 4, 6, 8 and 10 pixels. The stride is set to the half of the spatial bin size. Features with a small L_2 norm with respect to the default SIFT threshold are discarded, while the remaining ones are L_2 normalized. This very dense sampling is necessary to localize the objects which cover a rather small part of the image, however it results in around 80×10^3 patches per image.

We reduce the dimensionality of local descriptors from 128 to D=80 by projecting them onto the global PCA subspace. The subspace is learned from 10^6 patches uniformly sampled from the training images. A GMM with K=1024 components is learned by the EM implementation from Yael (Douze and Jégou, 2009). The dimensionality of Fisher vectors computed *w.r.t.* the learned GMM is therefore 164864. For computation of patch Fisher vectors we use fixed K_{SA}=4.

We learn a sparse logistic regression model by employing the stochastic gradient descent (SGD) (Bottou, 1991) implementation from scikit-learn (Pedregosa et al., 2011) and SPAMS (Bach et al., 2012). Preliminary experiments have shown that sparse logistic regression slightly outperforms sparse support vector machines in terms of image classification performance, so sparse logistic regression was used in all experiments. The number of epochs in SGD (*i.e.* the number of iterations over all images in the train split) is set to 50. The parameter λ that controls the model sparsity is determined by cross-validation on the train split in the range $\lambda \in \left[10^{-7}, 10^{-4}\right]$.

In the detection phase, we build the spatial graph by connecting the top ranked patches which overlap more than 25%, and remove connected components that contain less than 10 patches. We form one spatial graph per patch size, and therefore may have multiple detections per object. This prevents highly ranked background patches of different sizes to form connected components, and allows to accurately determine the bounding boxes by adding the margin corresponding to the half of the patch size.

Classification Results. In sections 3 and 4 we have showed that a slightly modified Fisher vector pipeline for image classification can be employed for weakly supervised localization. The two required modifications are i) avoiding non-linear normalizations in order to preserve additivity, and ii) employ L1 regularization in the classification model (instead of the L2 regularization employed in SVM) in order to induce sparsity of the classification model. Here we wish to evaluate the influence of these two decisions to the

[3] URL http://mastif.zemris.fer.hr/datasets.shtml

image classification accuracy. We learn the image classification model on the train split of our dataset, and report the average precision on both splits (train and test) as well as the achieved overall sparsity. The results are shown in Table 1.

Table 1: Influence of regularization and non-linear normalization to the average classification precision.

$\mathcal{R}(\boldsymbol{w})$	normalization	train	sparsity	test
L1	none	1.00	99.7%	0.72
L1	power + metric	0.91	99.6%	0.80
L2	none	1.00	0%	0.64
L2	power + metric	1.00	0%	0.73

Columns of the table correspond to the employed regularizer (L1 or L2), applied regularization (either only linear or linear, power and metric), AP on the train split, the achieved sparsity of the classification model, as well as the AP on the test split. The table shows that L1 regularization succeeds to induce classification models which employ only 0.3% (linear normalization) and 0.4% (all normalizations) of the Fisher vector representation. Even more interesting is the fact that these sparse models outperform the standard dense models induced by L2 regularization by 8 (linear normalization) and 7 (all normalizations) percentage points of AP accuracy. Thus, it turns out that by employing L1 normalization we are able to gain both accuracy and efficiency. The table also shows that some performance is lost by omitting nonlinear normalizations: 8 (L1) and 9 (L2) percentage points. However, this loss is not critical: we shall see that the linear Fisher vector representation paired with the sparse classification model shall achieve the weakly supervised localization AP larger than the best classification AP from the Table 1.

Localization Baseline. In order to provide insight into the difficulty of our problem, we show the results of a popular strongly supervised localization approach. We extract positive HOG descriptors (Dalal and Triggs, 2005) at all groundtruth locations in the train split, as well as about 25 times more negative HOG descriptors at random locations in a special dataset containing many images without any traffic signs. We used HOG implementation from OpenCV library (Bradski, 2000). The parameters of the HOG descriptor were: window_size=(24,24), block_size=(4,4), block_stride=(2,2), cell_size=(2,2) and n_bins=9. We weight the training samples in order to achieve class balance and learn a L2 regularized logistic regression model. We apply the classification model in the sliding window starting from the size 24×24 at 64 scales, with a scale factor of 1.03 and achieve the results summarized in Table 2.

Table 2: Baseline localization results with a strongly supervised linear classifier applied to HOG descriptors in the sliding window.

	AP train	AP test	processing time
results	0.968	0.944	10 s

Localization Results. We first evaluate the localization performance when the patch scores are obtained by using only soft-assign sparsity optimization. The results are presented in the top panel of Figure 3. The method performs remarkably well, given the difficulty of the problem. The best performance is achieved when bounding boxes are derived from a smaller number of top ranked patches T. However, considering only a small number of highly ranked patches increases the number of missed objects. Including more top ranked patches decreases the number of missed objects, but also deteriorates the detection performance.

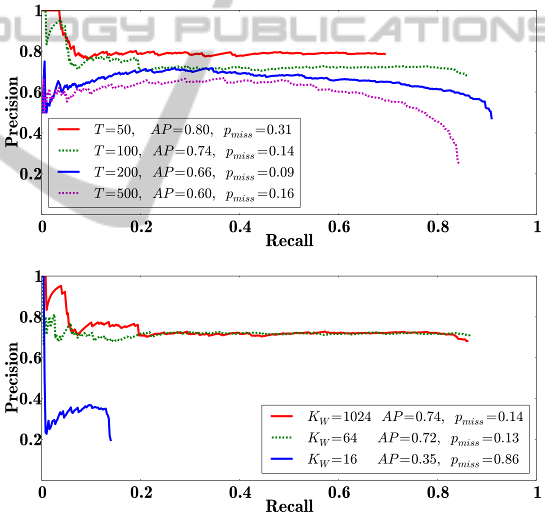

Figure 3: Precision-Recall curves displaying localization performance on the test set. *Top*: exploiting only soft-assign sparsity with fixed K_{SA}=4, full classification model (K_{W}=1024), and different numbers of top patches T to obtain the bounding box candidates. *Bottom*: exploiting both soft-assign sparsity (K_{SA}=4) and model sparsity while varying the number (K_{W}) of GMM components used by the classification model and keeping the number of top patches fixed (T=100). AP denotes the average precision and $\mathrm{p}_{\mathrm{miss}}$ corresponds to the proportion of missed objects (1-recall at the rightmost datapoint).

Effects of the model sparsity optimization are illustrated in the bottom panel of Figure 3. Our discriminative model is very sparse: it has only 1277 non-zero coefficients, which corresponds to only 0.77% of all image features. Consequently, the performance drops only slightly when K_{W}=64

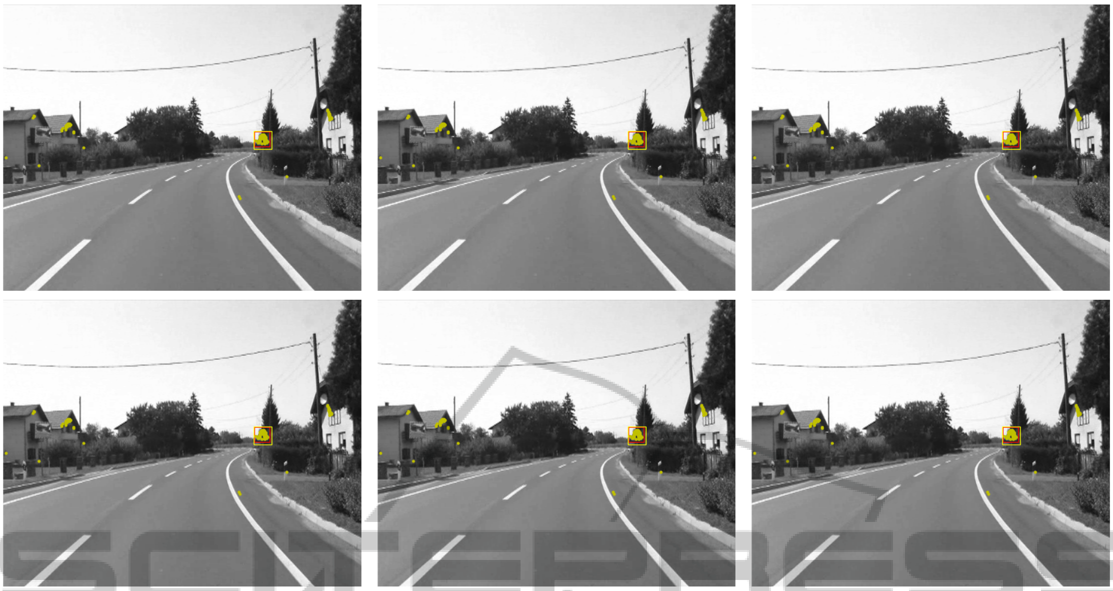

Figure 4: Successful operation: the method is able to detect multiple objects (*cf.* top-left) and to handle the complex background and small objects. Yellow dots indicate top ranked patches (rank$< T$), yellow rectangles show the detected bounding boxes, while red rectangles indicate the groundtruth locations (which are used exclusively for evaluation).

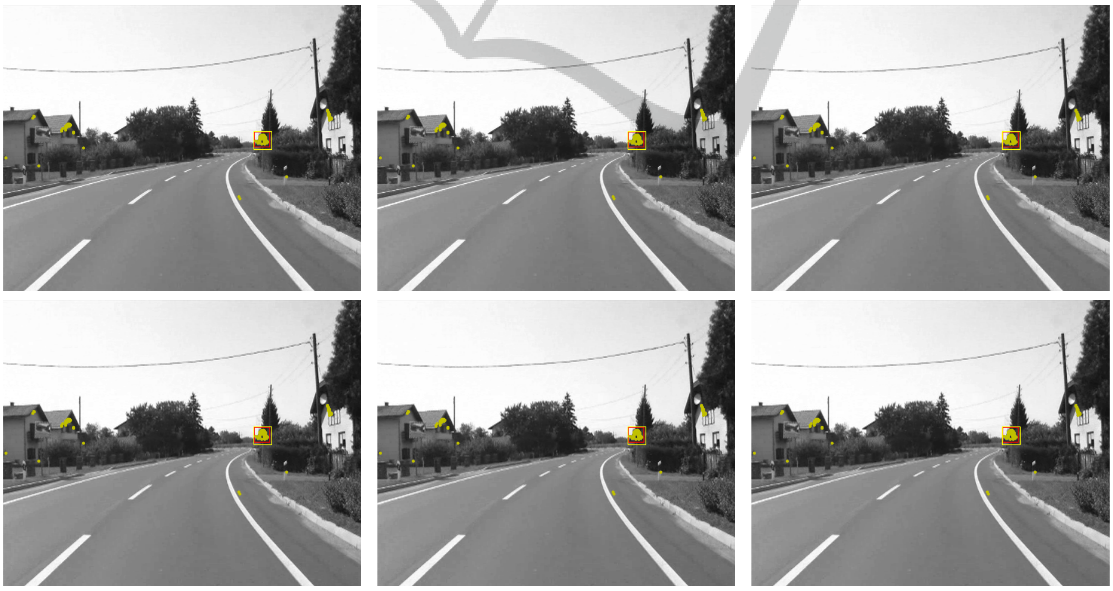

Figure 5: Detections counted as false alarms (top row), and missed detections (bottom row). The main causes of false alarms are insufficient overlap with the groundtruth location and background structures similar to warning signs (*e.g.* roofs). Most missed signs are either small or contain out-of-plane rotations which are seldom encountered in the train split.

top GMM components are employed. However, using as little as K_{W}=16 top GMM components affects performance significantly. The localization AP on the train split was consistently better for 5 percent points in experiments with K_{W}=1024, K_{SA}=4 and $T \in 50, 100, 200, 500$. This means that the obtained bounding boxes could be used to bootstrap the learning of strongly-supervised object detectors.

Figure 4 shows some examples of correctly detected objects. The detected bounding boxes (yellow) display very good overlap with the groundtruth location (red). Note that we use the groundtruth location exclusively for evaluation purposes, the training procedure knows only whether an image contains a warning sign or not.

Figure 5 shows two kinds of failure cases: bounding boxes with high scores that do not overlap with the true bounding box, and true bounding boxes missed by our detection method.

8 CONCLUSION AND PERSPECTIVES

We have proposed a novel weakly supervised localization method based on classification of image patches represented with large Fisher vectors. The main advantage of our method is fast evaluation due to a sparse linear classification model trained with L1 regularization. The sparsity of our model is determined by the parameter λ which regulates the trade-off between the loss and the regularizer. We set that parameter by cross-validation, which implies that our model outperforms less sparse models on the train split. Supplying a larger λ would lead to enhanced sparsity and faster evaluation at the expense of some performance loss.

To the best of our knowledge, this is the first account of patch Fisher vectors being used for weakly supervised object localization. The method has been experimentally validated on a challenging public dataset and the obtained performance (90% recall, 75% precision) is comparable with strongly supervised approaches. The most interesting qualities of the proposed approach include:

1. it is based on a slightly downgraded state-of-the-art image classification approach;
2. does not require ad-hoc or bottom-up initialization;
3. it is trainable on images of very small objects (less than 1% of the image content);
4. it is trainable on very large datasets (thousands of images) in reasonable time.

Our results suggest that Fisher vectors hold a great potential in the field of weakly supervised object localization. An interesting direction for future work would be to use a block-sparse model to directly enforce sparsity over GMM components. This would also help to improve soft-assign time, which is currently the bottleneck of the method (our unoptimized Python implementation takes around 20 s in the detection stage). To this end we shall explore cascade classifiers in the original feature space for quick rejection of the patches that can not contribute to the top scores. An interesting extension would be a more expressive spatial layout model for proposing bounding boxes. Finally, we would like to tackle weakly-supervised localization of fine-grained object classes, as this problem has many interesting applications.

ACKNOWLEDGEMENTS

This work has been supported by the project VISTA - Computer Vision Innovations for Safe Traffic, IPA2007/HR/16IPO/001-040514 which is cofinanced by the European Union from the European Regional Development Fund.

Parts of this work have been performed while the first author was funded by the European Community Seventh Framework Programme under grant no. 285939 (ACROSS).

Parts of this work have been performed in the frame of the Croatian Science Foundation project I-2433-2014.

REFERENCES

Alexe, B., Deselaers, T., and Ferrari, V. (2010). What is an object? In *CVPR*, pages 73–80.

Andrews, S., Tsochantaridis, I., and Hofmann, T. (2002). ”Support Vector Machines for Multiple-Instance Learning”. In *NIPS*, pages 561–568.

Auer, P. (1997). ”On Learning From Multi-Instance Examples: Empirical Evaluation of a Theoretical Approach”. In *ICML*, pages 21–29.

Bach, F. R., Jenatton, R., Mairal, J., and Obozinski, G. (2012). Optimization with sparsity-inducing penalties. *Foundations and Trends in Machine Learning*, 4(1):1–106.

Bottou, L. (1991). ”Stochastic Gradient Learning in Neural Networks”. In *Neuro-Nîmes*.

Bradski, G. (2000). OpenCV library. *Dr. Dobb's Journal of Software Tools*.

Chen, Q., Song, Z., Feris, R., Datta, A., Cao, L., Huang, Z., and Yan, S. (2013). Efficient maximum appearance search for large-scale object detection. In *CVPR*, pages 3190–3197.

Chum, O., Perdoch, M., and Matas, J. (2009). Geometric min-hashing: Finding a (thick) needle in a haystack. In *CVPR*, pages 17–24.

Chum, O. and Zisserman, A. (2007). An exemplar model for learning object classes. In *CVPR*.

Cinbis, R. G., Verbeek, J. J., and Schmid, C. (2013). Segmentation driven object detection with fisher vectors. In *ICCV*, pages 2968–2975.

Cinbis, R. G., Verbeek, J. J., and Schmid, C. (2014). Multi-fold MIL training for weakly supervised object localization. In *CVPR*, pages 2409–2416.

Crandall, D. J. and Huttenlocher, D. P. (2006). ”Weakly Supervised Learning of Part-Based Spatial Models for Visual Object Recognition”. In *ECCV*, pages 16–29.

Crowley, E. J. and Zisserman, A. (2013). Of gods and goats: Weakly supervised learning of figurative art. In *BMVC*.

Csurka, G., Dance, C. R., Fan, L., Willamowski, J., and Bray, C. (2004). Visual categorization with bags of keypoints. In *ECCV workshop*, pages 1–22.

Dalal, N. and Triggs, B. (2005). ”Histograms of Oriented Gradients for Human Detection”. In *CVPR*.

Deselaers, T., Alexe, B., and Ferrari, V. (2012). ”Weakly Supervised Localization and Learning with Generic Knowledge”. *IJCV*, 100(3):275–293.

Douze, M. and Jégou, H. (2009). Yael library. https://gforge.inria.fr/projects/yael.

Everingham, M., Gool, L. J. V., Williams, C. K. I., Winn, J. M., and Zisserman, A. (2010). ”The Pascal Visual Object Classes (VOC) Challenge”. *IJCV*, 88(2):303–338.

Felzenszwalb, P. F., Girshick, R. B., McAllester, D. A., and Ramanan, D. (2010). ”Object Detection with Discriminatively Trained Part-Based Models”. *PAMI*, 32(9):1627–1645.

Galleguillos, C., Babenko, B., Rabinovich, A., and Belongie, S. J. (2008). ”Weakly Supervised Object Localization with Stable Segmentations”. In *ECCV*, pages 193–207.

Gosselin, P.-H., Murray, N., Jégou, H., and Perronnin, F. (2013). ”Inria+Xerox@FGcomp: Boosting the Fisher vector for fine-grained classification”. Technical report, INRIA.

Jaakkola, T. and Haussler, D. (1998). Exploiting generative models in discriminative classifiers. In *NIPS*, pages 487–493.

Lampert, C. H., Blaschko, M. B., and Hofmann, T. (2009). ”Efficient Subwindow Search: A Branch and Bound Framework for Object Localization”. *PAMI*, 31(12):2129–2142.

Lowe, D. G. (2004). ”Distinctive Image Features from Scale-Invariant Keypoints”. *IJCV*, 60(2):91–110.

Munder, S. and Gavrila, D. M. (2006). An experimental study on pedestrian classification. *PAMI*, 28(11):1863–1868.

Nguyen, M. H., Torresani, L., la Torre, F. D., and Rother, C. (2014). ”Learning discriminative localization from weakly labeled data”. *Pattern Recognition*, 47(3):1523–1534.

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

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Perronnin, F., Sánchez, J., and Mensink, T. (2010). ”Improving the Fisher Kernel for Large-Scale Image Classification”. In *ECCV*, pages 143–156.

Sánchez, J., Perronnin, F., Mensink, T., and Verbeek, J. J. (2013). ”Image Classification with the Fisher Vector: Theory and Practice”. *IJCV*, 105(3):222–245.

Siva, P. and Xiang, T. (2011). ”Weakly supervised object detector learning with model drift detection”. In *ICCV*, pages 343–350.

Sivic, J. and Zisserman, A. (2003). Video google: A text retrieval approach to object matching in videos. In *ICCV*, pages 1470–1477.

Vedaldi, A. and Fulkerson, B. (2008). VLFeat: An open and portable library of computer vision algorithms. http://www.vlfeat.org/.

Viola, P. A. and Jones, M. J. (2004). ”Robust Real-Time Face Detection”. *IJCV*, 57(2):137–154.