A Simple High Performance Approach to Semantic Segmentation

Gabriela Csurka and Florent Perronnin
Xerox Research centre Europe
6, chemin de Maupertuis
Firstname.LastName@xrce.xerox.com

Abstract

We propose a simple approach to semantic image segmentation. Our system scores low-level patches according to their class relevance, propagates these posterior probabilities to pixels and uses low-level segmentation to guide the semantic segmentation. The two main contributions of this paper are as follows. First, for the patch scoring, we describe each patch with a high-level descriptor based on the Fisher kernel and use a set of linear classifiers. While the Fisher kernel methodology was shown to lead to high accuracy for image classification, it has not been applied to the segmentation problem. Second, we use global image classifiers to take into account the context of the objects to be segmented. If an image as a whole is unlikely to contain an object class, then the corresponding class is not considered in the segmentation pipeline. This increases the classification accuracy and reduces the computational cost. We will show that despite its apparent simplicity, this system provides above state-of-the-art performance on the PASCAL VOC 2007 dataset and state-of-the-art performance on the MSRC 21 dataset.

1 Introduction

We are interested in the problem of semantic segmentation, *i.e.* assigning each pixel in an image to one of several semantic classes. This is a supervised learning problem in contrast to "classical" unsupervised segmentation which groups pixels into homogeneous regions based on low-level features such as the color or texture. Note that object localization is a particular instance of the semantic segmentation problem where the two classes are foreground and background.

One of the first approaches to simultaneous object recognition and localization is [11]. Images patches are extracted and matched to a set of codewords learned during a training phase. Each activated codeword then votes for possible positions of the object center.

Several authors have proposed to combine low-level segmentation with high-level representations. [2] computes a pixel probability map using a fragment-based approach and a multi-scale segmentation. The pixel labeling takes into account the fact that pixels within homogeneous regions are likely to be segmented together. [14] and [21] perform respectively normalized cuts and mean-shift segmentation and compute bags-of-keypoints at the region level. [3] uses Latent Dirichlet Allocation (LDA) at the region level to

BMVC 2008 doi:10.5244/C.22.22

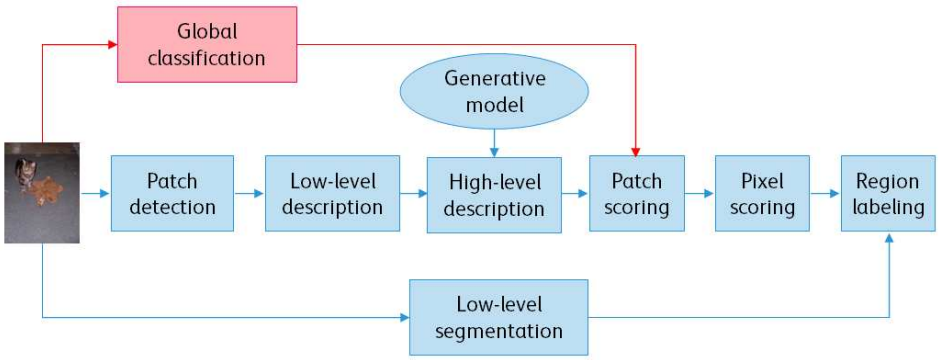

Figure 1: Proposed system overview.

perform segmentation and classification and enforce the pixels within a homogeneous region to share the same latent topic.

It is also possible to rely on low-level cues to improve the semantic segmentation without the need to perform explicit low-level segmentation. The different cues are generally incorporated in a random field model, such as a Markov random field (MRF). As local interactions are insufficient to generate satisfying results, global supervision is incorporated in the MRF. In the LOCUS algorithm [19] this takes the form of prototypical class mask which can undergo deformation. In the OBJ CUT algorithm [9] and in [17], it takes the form of a latent model.

While the MRF is generative in nature, the conditional random field (CRF) models directly the conditional probability of labels given images. [6] incorporates region and global label features to model shape and context. [10] proposed a two-layer hierarchical CRF which encodes both short- and long-range interactions. Textonboost [15] is a discriminative model which is able to merge appearance, shape and context information. [20] proposed the layout consistent random field, an enhanced version of the CRF which can deal explicitly with partial occlusion. [18] addresses the case of partially labeled images.

Our system bears some similarity with those of [2, 14, 21, 3] as we use low-level segmentation to guide the semantic segmentation.

In a nutshell, given an image the proposed approach works as follows (c.f. figure 1). First, patches are detected and low-level descriptors are computed for each patch. Given a low-level descriptor and a generative model, each patch is described by a high-level representation. These high-level patch descriptors are scored with respect to each class and the patch scores are propagated to the pixels, (*i.e.* one pixel probability map is computed per class). Finally, low-level segmentation is performed and a voting is done at the region level (*i.e.* one label is assigned to each region).

This is similar to the approach which consists in aggregating patch-level representations at the region level and then scoring the region-level representations as done in [14, 21, 3]. However, we believe that using the intermediate class probability map as in [2] leads to a more general framework as one can easily extend it to use a MRF or a CRF to enforce local consistency instead of low-level segmentation.

The two main contributions of this paper are as follows:

- Instead of using the traditional bag-of-words, we propose to use the Fisher vector

[7] as high-level representation of our patches at step 3. The Fisher kernel was successfully applied to image classification [13] and we will show that it leads to superior performance with respect to the bag-of-words for the segmentation problem.

- We refine the previous system by introducing a fast rejection step at the image level (shown in red on figure 1): if the probability that the image contains an instance of a given class is low, the given class is not considered in steps 4 to 6 of the algorithm. This is a simple and very general approach to use the object context and to enforce the global consistency of the labeling. It significantly speeds up the segmentation while improving the accuracy. Also we will show that by focusing only on those images which are likely to contain the objects of interest, we can learn a more accurate patch classifier.

In section 2, we present in more detail the proposed system, emphasizing the main novelties: the Fisher kernel representation and the use of a global rejection mechanism to take into account the context. In section 3, we show experimental results on the PASCAL VOC 2007 database and the MSRC 21 dataset before drawing conclusions.

2 Proposed Approach

We now describe in more detail the different steps of the proposed algorithm. Note that we will not elaborate on the first two components of our system as it can accommodate virtually any patch detector and low-level descriptor.

2.1 High-level Patch Description

In the image classification literature, the traditional approach to transform low-level features into high-level representations is the bag-of-visual-words (BOV) [16, 4]. The BOV is based on an intermediate representation, the visual vocabulary. In the case of a generative approach, the visual vocabulary is a probability density function (pdf) – denoted p – which models the emission of the low-level descriptors in the image. We model the visual vocabulary with a Gaussian mixture model (GMM) where each Gaussian corresponds to a visual word. Let λ be the set of parameters of p. $\lambda = \{w_i, \mu_i, \Sigma_i, i = 1...N\}$ where w_i, μ_i and Σ_i denote respectively the weight, mean vector and covariance matrix of Gaussian i and where N denotes the number of Gaussians. Let p_i be the component i of the GMM so that we have $p(x) = \sum_{i=1}^{N} w_i p_i(x)$. Finally, let $\gamma_i(x_t)$ be the probability that the low-level descriptor x_t is assigned to Gaussian i. This quantity can be computed using Bayes formula:

$$\gamma_i(x_t) = \frac{w_i p_i(x_t)}{\sum_{j=1}^{N} w_j p_j(x_t)}. \quad (1)$$

In the bag-of-words representation, the low-level descriptor x_t is transformed into the high-level descriptor f_t as follows: $f_t = [\gamma_1(x_t), \gamma_2(x_t), \ldots, \gamma_N(x_t)]$.

We propose as an alternative to the bag-of-words at the patch-level the Fisher representation. The Fisher vector describes in which direction the parameters of the model should be modified to best fit the data. In this case, the high-level descriptor f_t is given by $f_t = \nabla_\lambda \log p(x_t|\lambda)$. We follow [13] and consider only the gradient with respect to the

mean and standard deviation as it was shown that the gradient with respect to the mixture weights does not contain significant information. In the following, the superscript d denotes the d-th dimension of a vector. We have the following formulas for the partial derivatives:

$$\frac{\partial \log p(x_t|\lambda)}{\partial \mu_i^d} = \gamma_t(i)\left[\frac{x_t^d - \mu_i^d}{(\sigma_i^d)^2}\right], \quad (2)$$

$$\frac{\partial \log p(x_t|\lambda)}{\partial \sigma_i^d} = \gamma_t(i)\left[\frac{(x_t^d - \mu_i^d)^2}{(\sigma_i^d)^3} - \frac{1}{\sigma_i^d}\right]. \quad (3)$$

The Fisher gradient vector is just the concatenation of these partial derivatives. These vectors are subsequently whitened using the normalization technique described in [13]. While both the bag-of-words and Fisher representations are very high dimensional, they are also very sparse as only a very small number of components i (typically < 5) have a non-negligible value $\gamma_i(x_t)$ for a given t. This makes the storage and processing of these high-level patch representations manageable. Note that the typical image-level BOV or Fisher representations are simply the average of these patch representations.

2.2 Patch scoring

These high-level descriptors are subsequently scored according to their class relevance. Any discriminative classifier may be used and we chose Sparse Logistic Regression (SLR) [8]. The relevance of f_t with respect to class c is:

$$p(c|f_t) = \frac{1}{1+\exp(-(w_c' f_t + b_c))}, \quad (4)$$

where w_c and b_c are respectively the learned separating hyperplane and offset for class c. One of the advantages of SLR is that w_c is typically very sparse which means that SLR performs simultaneous classification and feature selection. This speeds-up the computation of $p(c|f_t)$. Note that, instead of learning each patch classifier independently, one could have learned them jointly using for instance Sparse Multinomial Logistic Regression (SMLR) [8].

In the following, we assume that the location of the objects in the training set if provided. The location may be a bounding box, a more complex polygon or a pixel mask (in the following, we will use the generic term "mask"). For each image and each class, we assume that there is an object mask (which can be empty) which will be referred to as positive mask and its complementary which will be referred to as negative mask. If several instances of an object are present in the image the mask refers to their union. The discriminative classifier can be learned at different levels:

- *Patch level*: we use as positive (resp. negative) samples the high-level descriptors corresponding to the patches within (or significantly overlapping with) the positive (resp. negative) masks of this class. As the number of training samples can be very large (several millions), we use a sub-sample of the whole training set.

- *Mask level*: we use as samples the averages of the high-level vectors over the masks. The main advantage of this approach compared to the previous one is the smaller number of training samples (3 orders magnitude less in our experiments if we were

to use all the patch-level representations) and thus the reduced computational cost at training time. The downside is the decrease in classifier accuracy.

2.3 Pixel scoring

We compute the class posteriors at the pixel level as a weighted average of the patch posteriors $p(c|f_t)$. For a pixel z and a class c we get:

$$p(c|z) = \frac{\sum_{t=1}^{T} p(c|f_t)\omega_{t,z}}{\sum_{t=1}^{T} \omega_{t,z}}. \quad (5)$$

The weights $\omega_{t,z}$ are given by the Gaussian Kernel $\mathcal{N}(z|m_t, C_t)$. m_t is the geometrical center of patch t and C_t a 2×2 isotropic covariance matrix with values $(\alpha s_t)^2$ on the diagonal where s_t is the size of patch t and α is a parameter (equal to 0.6 in the experiments). The isotropic covariance assumption corresponds to round patches. We thus obtain one probability map per class.

2.4 Region labeling

As labeling each pixel individually would lead to poor performance, we combine the probability maps with low level segmentation. Each image is segmented in a set of homogeneous regions according to some low-level features. Class probabilities are averaged over each region and each region $\mathcal{R}$ as a whole is assigned the most likely label: $c^*(\mathcal{R} = \arg\max_c \sum z \in \mathcal{R} p(c|z)$ We also include a rejection threshold and assign an "unknown" label to regions with low probabilities.

For the low-level mean-shift segmentation, we used the Edge Detection and Image Segmentation (EDISON) System [1]. The 5 dimensional pixel representations contain the *Lab* information and the pixel coordinates. The parameters of the segmentation are chosen so that we mostly over-segment the objects. The reason is that it is more penalizing to have two objects ending up in the same region than a single object being split in several regions. Connected regions with similar labels can be subsequently merged.

2.5 Global classification

Taking into account the context of an object generally improves the categorization performance. It was even shown at [5] on the PASCAL VOC 2007 database that state-of-the-art approaches to object categorization actually seem to use more the appearance of the context than the object appearance itself.

We thus propose the following approach to improve the segmentation accuracy. We train for each category a set of classifiers – one per class – at the image level. The global classifiers may be similar to the classifiers used at the patch level (*e.g.* in our case both patch and image classifiers are based on the Fisher representation). While this is not a requirement, it can reduce the computational cost. We thus get for each image the class posterior probabilities. If the score of a class is above a given threshold, then one computes the patch score and probability maps for that class and the corresponding probability map is used for the region labeling. If the score of a class is below the threshold, then this class is not considered in the remainder of the processing pipeline.

This simple modification offers the following advantages:

- It increases the segmentation accuracy by considering the objects context. Indeed, if we do not use this filter, each region is classified individually which may result in incorrect region labeling, especially in the case of small regions.

- It significantly decreases the computational cost as, generally, only few classes pass the global score test. In our experiments, we set a threshold value of 0.5 on the posterior class probabilities. as a trade off between recall and precision. On the average, 1.4 classes pass the test per image on the PASCAL VOC 2007 dataset. This value is comparable to the average number of classes per image, as estimated on the trainval set (approximately 1.46).

- Weakly labeled data, *i.e.* images where the labels are assigned to the whole image, is easier and less costly to collect than strongly labeled data, *i.e.* images where the objects have been segmented. While training the patch classifiers requires strongly labeled data, training the global image classifier only requires weakly labeled data. Hence, our algorithm can make efficient use of weakly labeled images.

The main drawback of this rejection mechanism is that, if an object appears in an unusual context (*e.g.* a cow in an urban setting), the global classifier might prevent the segmentation stage to discover the object.

We can also use this fast rejection to improve the quality of the patch classifier. As the only images which pass the global rejection test are those which are likely to contain the object, we can train the patch classifier to segment specifically an object from its usual background/context. When training the classifier at the patch level, we use as negative samples all the patches which significantly overlap with negative masks located in images which have a high posterior probability (most of which should be images containing the considered object class). When training the classifier at the mask level, we use as negative samples all the " negative" masks which are in images which have a high posterior probability.

3 Experimental Results

We first describe our experimental setup. We then report the results of our evaluations on two publicly available datasets: PASCAL VOC 2007 and MSRC21. Figure 2 shows example segmentations for both databases.

3.1 Experimental setup

Our system extracts patches on grids at multiple scales and use SIFT-like features [12] as well as simple color features. The dimensionality of these features is subsequently reduced to 50. For the high-level patch representations, we used respectively a visual vocabulary of $1,024$ Gaussians for the BOV and 64 Gaussians for Fisher.

As for the image-level rejection mechanism, we used the Fisher representation and the SLR classifier.

Brookes	INRIA best	MPI best	TKK	UoCTTI
8.5 / 5.6	23.5 / 5.0	27.8 / 8.6	30.4 / 7.4	21.2 / 5.5

Table 1: Results of the main competitors (pixel / union measures) on the PASCAL VOC 2007 segmentation challenge.

	BOV		Fisher	
	Patch	Mask	Patch	Mask
no GR	30.4 / 11.6	17.0 / 10.3	39.5 / 15.0	27.7 / 15.9
GR1	36.0 / 19.3	16.3 / 11.9	38.7 / 21.0	26.5 / 18.8
GR2	39.8 / 24.2	21.4 / 15.9	39.4 / 25.8	36.6 / 24.0

Table 2: Results (pixel / union measures) for the systems based on the BOV and Fisher representations. We consider the cases where the patch classifiers are learned at the patch and mask level. We also consider the cases where we do not use global rejection (no GR), where we use global rejection (GR1) and in the case where we use the global rejection and learn the patch classifiers on the non-rejected images (GR2).

3.2 PASCAL VOC 2007

We first report results on the Pascal 2007 segmentation dataset [5]. This dataset contains 422 training images and 210 test images. During the 2007 evaluation campaign, only one system entered the challenge, Brookes. However, several systems which took part in the detection challenge were also automatically scored on the segmentation challenge. For the detection challenge, 5,011 training images were available for training. In our experiments, we used this enlarged dataset to train our segmenter. Hence, we have an unfair advantage with respect to Brookes.

The measure of accuracy which was used during the competition is the "pixel measure" which is the number of true positives divided by the number of ground truth positives. It was later found to reflect only imperfectly the segmentation accuracy as it does not take into account false positives. Therefore, we also report results using the so called "union" or "intersection/union" measure which is the number of true positives divided by the number of ground truth positives plus false positives [1]. It should be underlined that *this measure was not what the competition was being assessed on and so the participants were not necessarily optimizing for that measure.* Table 1 summarizes the results (in percentages) of the competitors in the challenge. It is clear from the differences in ranking that the pixel and union measures are significantly different.

Table 2 reports results for our system with different settings: without global rejection (no GR), with global rejection (GR1) and with global rejection and a patch classifier learned only using these images which pass the fast rejection step (GR2) (c.f. section 2.5). We also report results when the classifiers are learned at the patch and mask level (c.f. section 2.2). We can see that Fisher outperforms BOV systematically on the union measure and almost always on the pixel measure. especially when the patch classifiers are learned at the mask level. We recall that learning at the mask level is clearly advantageous from a computational standpoint. We note that in all cases, the fast rejection improves

[1] The "unofficial" results with this new measure can be found at:
https://www.comp.leeds.ac.uk/me/VOC/voc2007prelimresults/segunion.html

Figure 2: Sample segmentation results for the VOC07 and MSRC 21 databases.

	building	grass	tree	cow	sheep	sky	aeroplane	water	face	car	bicycle
[15]	62	**98**	86	58	50	83	60	53	74	63	75
[21]	63	**98**	**90**	66	54	86	63	71	83	**71**	80
[17]	52	87	68	**73**	**84**	**94**	**88**	73	70	68	74
Our best	**84**	95	81	67	78	89	72	**77**	**87**	**71**	**86**

	flower	sign	bird	book	chair	road	cat	dog	body	boat	average
[15]	63	35	19	**92**	15	86	54	19	**62**	7	58
[21]	71	38	23	88	23	88	33	34	43	**32**	62
[17]	**89**	33	19	78	**34**	**89**	46	**49**	54	31	64
Our best	66	**59**	**28**	85	19	68	**59**	47	35	9	**65**

Table 3: Per-class and average segmentation accuracy on the MSRC 21 dataset.

the classification accuracy and that learning the patch classifier only on the non-rejected images brings an additional improvement. The proposed system is above the state-of-the-art, both on the pixel (39.8 vs 30.4) and union (25.8 vs 8.6) measures. For this case, overall 73.5% pixels were correctly labelled.

As for the computational cost, the low-level segmentation clearly dominates. On a 2.4 GHz Opteron^{TM}machine the low-level segmentation takes approximately 30 s per image (using the implementation of mean shift provided at [1]) while the remainder of the processing (patch extraction, computation of Fisher representations, global image scoring, patch scoring, pixel scoring and region labeling) takes less than 1 s.

3.3 MSRC 21

We now report results on the Microsoft Research Cambridge database (MSRC 21). It contains 591 color images of 21 object classes such as building, grass, tree, cow, sheep,

etc. Several papers report results on this dataset with similar training / test conditions including [15, 21, 18]. We use the same protocol as [18], *i.e.* the training set contains 276 training images picked randomly and the test set the remaining 315 test images. *As the training and test splits are not exactly the same across the different papers, we should exercise caution when drawing conclusions.* The segmentation accuracy is reported in terms of pixel classification accuracy for each classes. The overall pixelwise segmentation accuracy is 77.1%. Due to space constraints, we report only our best results in Table 3. This is the system using Fisher representation and patch-classifiers learned at the patch level. However, this does not include the global rejection step as it actually decreased slightly the performance on this dataset (approximately 62%). We believe that this is due to the lack of per-class training material to train a good image classifier. Note that our results are comparable to the state-of-the-art although our system is much simpler than those of [15, 21, 18].

4 Conclusion

We used in this paper a simple framework to semantic segmentation. Our system scores low-level patches according to their class relevance, propagates these posterior probabilities to pixels and uses low-level segmentation to guide the semantic segmentation. Our first contribution was to use the Fisher kernel [7] to derive high-level descriptors to compute the patch level class-relevance. While the Fisher kernel had already been shown to lead to high accuracy for image classification [13], it had not been applied to the segmentation problem. We showed experimentally that it generally leads to higher performance compared to the BOV. The second contribution was to use classification at the image level to take into account the objects context. This step discards unlikely hypotheses and thus speeds up computation. We also showed that, when enough training material is available to train image-level classifiers, this fast rejection increases performance. Overall, it was shown on the PASCAL VOC 2007 and MSRC 21 datasets that, despite its apparent simplicity, this system provides state-of-the-art performance.

While we currently guide the semantic segmentation using low-level segmentation, our system could also be integrated with random field approaches. We are currently investigating this path.

References

[1] Edge detection and image segmentation (EDISON) system. http://www.caip.rutgers.edu/riul/research/code/EDISON/index.html, 2003.

[2] E. Borenstein, E. Sharon, and S. Ullman. Combining top-down and bottom-up segmentation. In *CVPR*, 2004.

[3] L. Cao and L. Fei-Fei. Spatially coherent latent topic model for concurrent segmentation and classification of objects and scenes. In *ICCV*, 2007.

[4] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *ECCV Workshop on Statistical Learning for Computer Vision*, 2004.

[5] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html.

[6] X. He, R. Zemel, and M. Carreira-Perpi nán. Multiscale conditional random fields for image labeling. In *CVPR*, 2004.

[7] T. Jaakkola and D. Haussler. Exploiting generative models in discriminative classifiers. In *NIPS*, 1999.

[8] B. Krishnapuram and A. J. Hartemink. Sparse multinomial logistic regression: Fast algorithms and generalization bounds. *PAMI*, 27(6), 2005.

[9] M. Pawan Kumar, P.H.S. Torr, and A. Zisserman. Obj cut. In *CVPR*, 2005.

[10] S. Kumar and M. Hebert. A hierarchical field framework for unified context-based classification. In *ICCV*, 2005.

[11] B. Leibe, A. Leonardis, and B. Schiele. Combined object categorization and segmentation with an implicit shape model. In *ECCV Workshop on Statistical Learning for Computer Vision*, 2004.

[12] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[13] F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *CVPR*, 2007.

[14] B. Russell, A. Efros, J. Sivic, W. Freeman, and A. Zisserman. Using multiple segmentations to discover objects and their extent in image collections. In *CVPR*, 2006.

[15] J. Shotton, J. Winn, C. Rother, and A. Criminisi. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *ECCV*, 2006.

[16] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *ICCV*, volume 2, pages 1470–1477, 2003.

[17] J. Verbeek and B. Triggs. Region classification with markov field aspects models. In *CVPR*, 2007.

[18] J. Verbeek and B. Triggs. Scene segmentation with crfs learned from partially labeled images. In *NIPS*, 2007.

[19] J. Winn and N. Jojic. Locus: Learning object classes with unsupervised segmentation. In *ICCV*, 2005.

[20] J. Winn and J. Shotton. The layout consistent random field for recognizing and segmenting partially occluded objects. In *CVPR*, 2006.

[21] L. Yang, P. Meer, and D.J. Foran. Multiple class segmentation using a unified framework over mean-shift patches. In *CVPR*, 2007.