SIMULTANEOUS DETECTION AND SEGMENTATION FOR GENERIC OBJECTS

Albert Torrent[1], *Xavier Lladó*[1], *Jordi Freixenet*[1] *and Antonio Torralba*[2]

[1]University of Girona, ATC, Girona, Spain
[2]Massachusetts Institute of Technology, CSAIL, Cambridge, Massachusetts

ABSTRACT

Numerous approaches to object detection and segmentation have been proposed so far. However, these methods are prone to fail in some general situations due to the proper object nature. For instance, classical approaches of object detection and segmentation obtain good results for some specific object classes (i.e. detection of pedestrians or segmentation of cars). However, these methods have troubles when detecting or segmenting object classes with different distinctive characteristics (i.e. cars and horses versus sky and road). In this paper, we propose a general framework to simultaneously perform object detection and segmentation on objects of different nature. Our approach is based on a boosting procedure which automatically decides - according to the object properties - whether is better to give more weight to the detection or segmentation process to improve both results. We validate our approach using different object classes from LabelMe, TUD and Weizmann databases, obtaining competitive detection and segmentation results.

Index Terms— Simultaneous detection and segmentation, boosting classifier

1. INTRODUCTION

Recognizing and classifying objects of interest in images is a challenging and important problem nowadays. Object recognition in images is typically divided in two different strategies, the object detection and the object segmentation. Classical approaches to deal with object detection are based on learning information from a given object model. These kind of approaches have been successfully used to detect rigid objects such as faces and cars, and also articulated objects such as pedestrians [1]. Other works have used a strategy to detect object parts, assembling them into a whole object [2]. However, these methods have problems to detect objects where their shape properties are not discriminative (i.e. sky or road).

Some other works have been focused on the object segmentation. For instance, Aldavert et al. [3] proposed a method based on the popular Bag of Features for a pixel level classification. Another example is the method proposed by Carreira et al. [4], in which are generated a set of figure-ground segmentation hypothesis to get the final object segmentation.

More recent approaches have introduced the idea to perform simultaneous object detection and segmentation.For instance, Wang et al. [5] propose to use the detection process to help the segmentation, while Ramanan [6] presented an approach for object recognition that uses the segmentation results to refine the detection ones. They compute a figure-ground segmentation at each hypothetized detection and learn from training data those segmentations that are consistent with the true positives. Finally, Wu and Nevatia [7] also proposed to simultaneously detect and segment objects. Their approach is based in a boosting process which selects features, by weighting the detection and segmentation errors.

Analyzing the literature results, we have seen that there are object classes that have reported better results in detection than in segmentation (i.e. people or car), while others (i.e. sky or road) have the opposite behavior. In this paper we propose a novel approach for simultaneous object detection and segmentation, with the idea that good detections can help the segmentation process and viceversa. Our approach is able to detect and segment any generic object and produces competitive results for object classes like cars, horses, sky and road. The main idea is to give more relevance to the process that provides better results (detection or segmentation) depending on the characteristics of the object. The results presented in this paper using different image object classes extracted from the LabelMe, the TUD and the Weizmann databases illustrate the validity of our approach, showing also the benefits of doing the simultaneous detection and segmentation.

2. OUR FRAMEWORK

Our approach is based on the work of Torralba et al. [8] for object detection using local patches and their relative position to the object center. Their proposal is adapted here to extract also segmentation features. The feature extraction is divided into two steps: 1) the generation of a dictionary, and 2) the feature extraction using this dictionary.

The first task is the building of the feature dictionary. Note that the definition of this dictionary contains the words which we used to extract the features for training and testing. This dictionary is for an object class, so a dictionary for each object has to be build. First of all, we randomly select a set of images to create the dictionary. These images are different from the

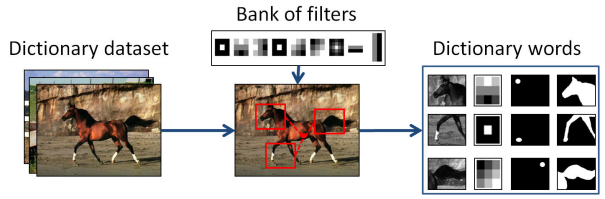

Fig. 1. Generation of a dictionary word. Every word contains the filtered local patch, the filter used, the relative position of the patch with respect to the object center and the ground truth segmentation of the patch.

ones used to train and test the system. We then convolve the images with a bank of filters.

After filtering the images we extract some different patches from them. We select a set of patches located in a random position around the object. Patches of different sizes are extracted from each selected location. Each filtered patch will become a word of the dictionary. As well as the patch, the filter used is also needed to extract the features of an image, since the patch is convolved with the filtered image. Moreover, for each word of the dictionary two more elements are needed, one to get the features of the detection and another one for the segmentation. Figure 1 illustrates the extraction of the dictionary words, composed by the local patch, the filter used, the relative position of the patch with respect to the center and the patch ground truth segmentation.

When the dictionary has been built, we can characterize the pixels of an image using the same equation for detection and segmentation as follows:

$$v = [(I * f) \otimes p] * g \tag{1}$$

where v is the characterized image, I the original image, f the filter, p the filtered patch and g the mask. We use two different masks, one to extract the detection features and another one to extract the segmentation features. In the case of detection, g is the relative location of the patch p respect to the object center, while in the case of segmentation it is the real segmentation of the patch. Therefore, we convolve the image with the filter and we then perform a normalized cross correlation with the patch. As a result we get a probability image with high values on the regions that are similar to the patch. Finally, we convolve this probability image with the g mask in order to get high values in the object center for the detection case, or in the pixels of the object for the segmentation case. Note that with this pixel-based process we characterize each pixel of the image.

In order to train the classifier we need to select training points. To reduce the high computational cost of using all the points of all the training images we decided to reduce the number of training samples. For the detection task we select the center of the object as the positive sample, and some points of the background as the negative ones. Note that we do not select any point of the object as negative, although they are not part of the object center, in order to avoid possible confusions. On the other hand, for the segmentation process the algorithm selects some random points of the object as positives and some random points of the background as negative. For each point we then extract a feature vector (one feature for each dictionary word).

3. SIMULTANEOUS DETECTION AND SEGMENTATION

The object detection and segmentation methods proposed in the previous section could be applied as independent processes. However, our idea is to develop a boosting framework for simultaneous object detection and segmentation which automatically gives more relevance to the detection or the segmentation process depending on the object properties. According to the global detection and segmentation cost, the best option is selected at each boosting round. The idea is to use good detections to help the segmentation process, as well as to use good segmentations to improve the detection task.

3.1. Crossing detection and segmentation information

One of the main contributions of our approach is to use the partial results obtained during the boosting training as input for the next rounds. In particular, we use the detection results as input for the segmentation, and viceversa. This means that the input features of the boosting training change every round.

To perform a crossing from segmentation to detection we use a simple idea. If we have a good enough object segmentation, its center will be located at the segmentation center, and the probability of finding it will decrease radially as further from the segmentation center. In particular, we generate a probability image with maximum value at the center of the segmentation result. Therefore we can improve the object detection on object classes in which good segmentation results are obtained due to their intrinsic properties.

On the other hand, if we have good detection results the crossing consists in using the detection image of the previous round to get the segmentation of the actual round. First, we apply a threshold to get the regions with high probability of being an object center. We then use the dictionary words to generate a segmentation from an object center. Afterwards, we convolve the detection with the relative position of a patch with respect to the center to get high values in the position of the image where we expect to find an specific patch. Finally, we convolve these results with the ground truth segmentation of the patch to obtain the object part segmentation.

3.2. Boosting

Boosting algorithms are based on the simple idea that the sum of weak classifiers can produce a strong classifier. In our system we use the gentleBoost algorithm proposed by Friedman et al. [9]. The weak classifiers h_t are simple regression stumps

with one of the features, so at each round the feature with less error is selected. The weak classifier function is

$$h_t(x) = a(x_i > th) + b \tag{2}$$

where x is the data, x_i is the i'th dimension (feature) of x, and th is a threshold that determines if the data belongs to the object class or not. Note that a and b are parameters selected to minimize the error function of Eq. 3 given the chosen feature x_i and threshold th.

$$\varepsilon = \sum_{s=1}^{S} (w_s(y_s - (a(x_{i,s} > th) + b))^2) \tag{3}$$

where y are the labels, w are the training data weights updated at each round, and S the number of training samples. Eq. 4 is then used to update the data weights.

$$w_{t+1} = w_t e^{y \cdot h_t(x)} \tag{4}$$

The final classifier $H(x)$ is the sign of the result of the weak classifiers sum ($H(x) = sign(\sum h_t(x))$).

We integrate the detection and segmentation classifiers in a single boosting training, so at each round we only perform a detection or a segmentation step. On each round of the boosting we select both the best weak rule for detection and segmentation. However, we finally select the one that minimizes the global cost of detection and segmentation, defining their global cost J as

$$J = \alpha J^d + \beta J^s \tag{5}$$

where J^d and J^s are the detection and segmentation costs respectively, defined by the cost function: $\sum e^{-y \cdot H_t(x)}$, where y are the labels of the training samples and H_t the predictions results at round t.

Note that as is shown in Eq. 5, we weight the detection and segmentation costs with α and β. In fact, we give more weight to the detection cost, since this is an easier task and also because we use more training samples for the segmentation task. These two parameters are found empirically and fixed in all our experimental tests.

Moreover, we allow our approach to use the knowledge acquired during the training of the previous rounds. In particular, in a round t we apply the classifier trained during the $t-1$ rounds, obtaining a partial result for detection and segmentation for every training image. With these results we can apply the technique described in section 3.1 to cross the detection and segmentation results, so a good detection help to segment and viceversa.

Therefore, our boosting classifier has four inputs to select the best weak rule generating two different kinds of outputs: 1) the use of the image properties for detection, 2) the use of the previous segmentation results for detection, 3) the use of the image properties for segmentation, and 4) the use of the previous detection results for segmentation. Note that the training sample points should be the same when detecting using the image properties than when crossing from segmentation (the same applies for the segmentation process).

4. RESULTS

The aim of our experimental results is twofold: 1) to evaluate the performance of our detection and segmentation classifier in different kind of objects, and 2) to demonstrate that crossing information between detection and segmentation allows to improve the results. To test our segmentation approach we used different representative object classes: sky and road from the LabelMe database [10], cars (side view) from the TUD database [11], and horses (side view) from the Weizmann database [12].

We have trained a classifier using three different strategies: S1) separately classifiers for detection and segmentation, S2) simultaneously detecting and segmenting (at each round we select to detect or to segment) and S3) simultaneously detecting and segmenting incorporating the crossing information between detection and segmentation. On each experiment we trained the classifiers with 500 rounds. However, in the first option we have a 500 weak classifier for detection and another 500 for segmentation while in the other two experiments we have only 500 weak rules automatically distributed for detection and segmentation. For both experiments the same 15 and 50 images have been used to generate the dictionary and train the classier respectively. We have used 35, 262, 223 and 135 testing images for the car, horse, sky and road class respectively.

Table 1 illustrates the segmentation results applying the classifiers to the set of testing images. In order to evaluate the segmentation results the percentage of pixels well classified and the area overlap measures have been used. On the other hand, Figure 2 shows image results of each class. Note that both quantitative and qualitative results confirm the segmentation improvement when introducing the crossing approach. Regarding the detection results we obtained a high percentage of TP detections, 100%, 97%, 89% and 90% for cars, horses, sky and road respectively, while the FP per image for each data set were 0, 0.1, 0.14 and 0.09 respectively. Note that for sky and road lower detection results were achieved due to the inherent difficulty of establishing the object center.

With the aim of providing a general trend of the performance of our approach, we also compared the segmentation results with those reported by recent segmentation approaches that used the same databases. For instance, for the cars TUD class we obtain similar results than those reported by Winn and Jojic [13]. While for the horse class we achieve worse results than those reported in [13, 14] and better results than the ones presented in [15] although in this work authors tested the images in inverted direction and under significant occlusions. On the other hand, we obtain better results than Russell et al. [16] for the road class, although our results are

Table 1. Segmentation results per object class when using the normal boosting process, when simultaneously detecting and segmenting, and when adding the crossing approach.

	AREA OVERLAP			% OF PIXELS WELL CLASSIFIED		
	Seg. only	**Seg. and Det.**	**Seg. cross**	**Seg. only**	**Seg. and Det.**	**Seg. cross**
Car	0.6756	0.6839	0.7647	0.9148	0.9182	0.9439
Horse	0.5828	0.5795	0.5738	0.8622	0.8655	0.8705
Sky	0.5813	0.5779	0.6358	0.8658	0.8655	0.8840
Road	0.6851	0.7043	0.7165	0.9091	0.9136	0.9173

not as good for the sky class. It should be noted that their results decrease when they segmented objects like cars, although their approach is unsupervised. To sum up, our segmentation approach is able to detect and segment objects of different nature, providing competitive results in comparison with the current state-of-the-art.

5. CONCLUSIONS

A novel approach able to detect and segment objects of different nature and with different distinctive characteristics (i.e. cars or sky) has been presented in this paper. Our approach is based on a boosting training procedure which automatically decides whether is more convenient to give more weight to the detection or the segmentation process according to the object properties itself. The proposed procedure permits also to cross information from detection to segmentation and viceversa, allowing to obtain satisfactory results on both kinds of objects. The experimental results show that the crossing option substantially increase the final performance. Moreover, we have seen that the obtained results are competitive with respect to the ones reported on the state-of-the-art. Our future work will be focused on adding new segmentation options and image features into the boosting procedure with the idea to provide more accurate segmentations in the boundaries.

(a) (b) (c)

Fig. 2. Detection and segmentation results for the car, horse, sky and road class. (a) using segmentation patches, (b) combining detection with segmentation, and (c) introducing the crossing between detection and segmentation.

6. REFERENCES

[1] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, 2005.

[2] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *PAMI*, vol. 32, no. 9, pp. 1627–1645, 2010.

[3] D. Aldavert, A. Ramisa, R. Toledo, and R. López de Mántaras, "Fast and robust object segmentation with the integral linear classifier," in *CVPR*, 2010.

[4] J. Carreira and C.Sminchisescu, "Constrained Parametric Min-Cuts for Automatic Object Segmentation," in *CVPR*, 2010.

[5] L. Wang, J. Shi, G. Song, and I. Shen, "Object detection combining recognition and segmentation," in *ACCV*, 2007.

[6] D. Ramanan, "Using segmentation to verify object hypotheses," in *CVPR*, 2007.

[7] B. Wu and R. Nevatia, "Simultaneous object detection and segmentation by boosting local shape feature based classifier," in *CVPR*, 2007.

[8] A. Torralba, K. Murphy, and W. Freeman, "Contextual models for object detection using boosted random fields," in *NIPS*, pp. 1401–1408. 2005.

[9] J. Friedman, T. Hastie, and R. Tibshirani, "Additive Logistic Regression: a Statistical View of Boosting," *The Annals of Statistics*, vol. 38, no. 2, pp. 337–374, 2000.

[10] B. Russell, A. Torralba, K. Murphy, and W.T. Freeman, "Labelme: A database and web-based tool for image annotation," *IJCV*, vol. 77, no. 1-3, pp. 157–173, 2008.

[11] B. Leibe, A. Leonardis, and B. Schiele, "Combined object categorization and segmentation with an implicit shape model," in *WSLCV*, 2004.

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

[13] J. Winn and N. Jojic, "Locus: learning object classes with unsupervised segmentation," in *ICCV*, 2005, pp. 756–763.

[14] E. Borenstein and J. Malik, "Shape guided object segmentation," in *CVPR*, 2006, pp. I: 969–976.

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

[16] 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.