Scale-independent object detection with an Implicit Shape Model

A. Furlan, D. Marzorati, D. G. Sorrenti

Dept. DISCo, Univ. di Milano - Bicocca, Italy; email: furlan@ira.disco.unimib.it, sorrenti@disco.unimib.it

Keywords: object detection, object recognition, tracking, implicit shape model

Abstract

In this paper we propose an improvement to the Implicit Shape Model (ISM) based robust object detection system proposed by Leibe et al. Object detection with ISM allows to approach the classification and tracking in a probabilistic way with multiple hypotheses. Unlike the original approach, our method is independent from object scale in the training sets, and this allows to work with a much smaller training sets and also to avoid to supply information about scale to the trainer. This is done while maintaining the robustness of the original approach. Leibe et al. mentioned a potential solution to overcome the scale problem in the training set, i.e., the usage of the scale produced by the local descriptor. Our proposal is different: since we believe that the scale measure generated by local descriptors is subject to noise, we try to walk around this noise by estimating the scale measure from the only evidence collected in the image.

1 Introduction

In the last years object detection and tracking could afford dealing with realistic, i.e., challenging, conditions thanks to the advancements in computer vision research. The object-detection task consists of the process of determining the class of specific objects in an image. Humans can perform this task easily, even if the objects appear rotated, scaled or partially occluded. To reach a human level ability is still an unsolved issue for state-of-the-art computer vision systems.

Usually, object detection algorithms cannot grant that the resulting interpretation of the image is the correct one. Therefore it is advisable to accumulate evidence over time, which implies coupled classification and tracking. We believe to be preferable to generate different interpretation hypotheses and then to choose those that better explain the actual observations of the world. In our thinking the different interpretation hypotheses should be characterized and compared in a probabilistic framework. The idea of probabilistic multiple hypotheses classification and tracking dates back to the early 90' and has been further developed over the years. The approach that best matches our objectives is the one in [5] [9].

In this paper we focus on object detection subsystem, improving the approach presented by Leibe et al. in [7].

In the next section we shortly review the state-of-the-art in object detection. In Sec. 3 we present the original formalization of the Implicit Shape Model. In Sec. 4 we focus on the codebook creation, and in Sec. 5 we review the original object detection system. In Sec. 6 we present our proposal for scale-independence, and in Sec. 7 we present some results of our system.

2 Related work

An ideal object detection system should be able to detect its targets in the observed scene independently on their orientation, position, scale, and also with some degree of occlusion.

An important challenge is to gain the capability of generalization, i.e., detecting the objects of a certain class and not only one instance, despite the differences in the object shape from one instance to the other in the class. An important contribution towards this objective has been the one by D. Lowe about scale-invariant descriptors [10]. The so-called SIFT descriptors, state-of-the-art today, provide descriptors of image patches that are invariant not only with respect to orientation and position, but also to scale and partial occlusion. Basing on such descriptors robust detectors can be developed.

For each image object one can detect many interesting features, for gathering a discriminating description of the object. By means of the descriptions accumulated during the processing of the images in an appropriate training set, one can then process new images in order to detect new instances of the object in new images.

In shape-based object recognition, see e.g., [12], the usage of statistics on the spatial frequencies of the descriptors is proposed. This is a quite effective approach whenever the objects to be detected have more or less constant shapes, and with regular textures; these approaches deal well with position, orientation, scale and occlusion. Such approaches do not perform well, on the other hand, on objects, e.g., people, that can appear in a wide set of different shapes and postures.

In 2003 Leibe and Schiele in [8] proposed a method that does not require an exact a priori knowledge about the objects to be detected, but only some instances. Therefore the training can take place on different instances of the objects with respect to the ones that the system could observe when running. This feature makes the system more complex, as the system cannot base on the specific objects, but has to consider large intra-class variations. In [6] the same authors, with Leonardis, formalized their approach in the so-called IMS (Implicit Shape Model), and later [4], with Mikolajczyk, they extended it to include different descriptors on common probabilistic grounds.

In the view of integrating the output of an object detector in a classifier and tracking system, we need to express the detector output in the form of a probability distribution. Liebe and Schiele approach has the relevant peculiarity that the detection output is a set of hypotheses on the size, pose, and class of one or more objects. Their approach can also be easily modified to accommodate a pixel level segmentation. This means that the detector will be able to output, for each object, a pixel map where, for each pixel, the probability of that pixel being part of the object (or being part of the background) is available. The usage of such method in a probabilistic framework for classification and tracking allows to generate 3D hypotheses (by means of camera-to-world projection) and to select the best interpretation of the image, provided some constraints are satisfied like, e.g., a pixel not being associated to two objects, etc. This in turn allows the classifier and tracking system to handle realistic partial object occlusions.

3 The Implicit Shape Model formalization

The approach proposed in [8], developed in [6] and extended in [4] rely on an implicit description of the objects. Thanks to this characteristic we don't need an explicit description any more which, as seen in the previous section, requires large training sets and complicated description models. This approach, on the other hand, builds its knowledge base, called codebook, by collecting description and spatial evidence from the training images.

This is performed by computing some local descriptors (e.g. SIFT, Shape Context) on interest regions automatically extracted from the images. This information, in conjunction with the spatial coordinates of the interest regions, is stored in the codebook and used later at detection time. Thus we collect information about the object in areas which provide more details in object description.

Formally, as stated in [6], an Implicit Shape Model $ISM(C) = (I_C, P_{I,C})$ for a given object category C consist of a knowledge base I_C (*codebook*), built-up with local descriptors discriminative for the object class, and of a spatial distribution $P_{I,C}$ that indicates where each codebook entry may be found on the training object. There are two requirements for the spatial distribution $P_{I,C}$. First, such distribution should be defined independently for each codebook entry, thus making the approach flexible and capable to merge, at detection time, parts of objects observed on different training images. Second, this distribution should be computed in a non-parametric way, thus emulating the real distribution as in detail as the training objects permit, or, in other words, exploiting as much training informations as possible. Furthermore this requirements frees us from making Gaussian assumption on the spatial distribution.

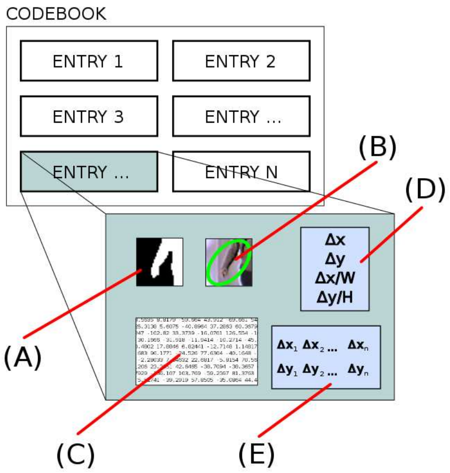
CODEBOOK
ENTRY 1
ENTRY 2
ENTRY 3
ENTRY ...
ENTRY ...
ENTRY N
(B)
(D)
Δx
Δy
Δx/W
Δy/H
(A)
Δx_1 Δx_2 ... Δx_n
Δy_1 Δy_2 ... Δy_n
(C)
(E)

Figure 1. Structure of the codebook and its entries. (A) represents the segmentation mask, (B) contains the parameters of the ellipsis which includes the image patch, (C) the values of the associated local descriptor calculated on the image patch, (D) the informations about the relative patch position with respect to object center, (E) lists all possible image centers for which this patch may cast votes.

4 Codebook creation

In the approach presented in [4] the codebook is filled with N entries, each one representing some evidence extracted from the training images. For each descriptor (named *cue* in the paper) we build a separate codebook. Each entry of such codebook should contain informations about its own relative position with respect to object center, the values of the associated local descriptor calculated on the image patch which generated that entry, a segmentation mask *figure/background*, the parameters of the ellipsis which includes the image patch, the object scale and the list of possible object centers for which this entry may cast votes. Since our proposed approach works independently of the object scale, this information can be omitted. In Fig. 1 the structure of the codebook and its entries are shown.

The procedure of codebook creation is divided in two parts. First we analyze the training images and populate the codebook with entries containing information about interest regions in the image. Second we compare, calculating the euclidean distance in the space of descriptor values, all codebook entries with each other. Each couple that results sufficiently similar enrich the spatial information of each entry with the other's one. This make it possible for a codebook entry to vote, at detection time, for more than one object center.

The first phase, i.e., the analysis of the training image, is performed in the following steps:

- We extract the most interesting image areas using one or more automatic interest region detectors. These algorithms ground on functions such as Harris function, Hessian determinant, etc. in order to extract scale invariant regions of interest from an image. An interest region is an image portion with high descriptive content (high edgeness, cornerness, etc.) which can be used for discriminative purposes in conjunction with some local descriptors.

- We calculate the values of local descriptors on interest regions. Each descriptor is composed by a different number of values, thus the codebook structure need to be sufficiently flexible in order to contain different descriptors. Local descriptors extract some kind of "fingerprint" of an image region, which are typically scale and rotation invariant. Some example of local descriptors are SIFT, PCA-SIFT, Shape Context, etc [11]. In our implementation we tested both SIFT and Shape Context descriptors.

- Finally we save in the codebook, for each interest region, the parameters of the ellipse, the spatial information, the values of the local descriptor and the local segmentation mask (extracted from the global segmentation mask supplied in the training set.

Before approaching the second part of the codebook creation procedure we filter out some codebook entries. This choice brings to two improvements: first it speeds up the detection algorithm and, second, it increments the quality of the codebook by reducing noise generated by the region finder algorithms.

In the second part of the codebook creation we compare all codebook entries with each other. The aim of this procedure is to enforce the generation of object hypotheses and their segmentation during detection and to enable single image patches to vote for more than one object center. The similitude measurement is calculated as euclidean distance in the M-dimensional space of descriptor values:

$$d(P_i, P_j) = \sqrt{\sum_{k=1}^{M} \left(P_i^k - P_j^k\right)^2} \tag{1}$$

If the distance between two entries is under an acceptance threshold, we say that their domain of discrimination is similar, thus we enable both entries to cast votes, at detection time, for their own center and for the center of the other entry.

This step concludes the codebook creation procedure.

5 Object detection

5.1 Image analysis

The image analysis procedure is similar to the one described in the codebook creation section. Applying several region detection algorithms on the image we are able to extract some interesting areas (in the following they will be called patches). For each patch we compute different values obtained by executing different region descriptors. These values will be stored for the following operations.

Next, the set of extracted image patches will be compared with all the codebook entries. For each pair patch-entry an Euclidean distance in the space of descriptor values is computed. If the distance is less than an a-priori threshold (the same used in the codebook creation procedure) then the patch will be associated to the codebook entry.

5.2 The original approach for voting

The original approach proposed by Leibe et al. in [4] bases on the equation that describes the probability that an object is located in a particular position with a specific scale, given the evidence, the position of the patch, and its descriptor.

$$p(o_n, \lambda | \mathbf{e}, l, q) = \sum_i p(o_n, \lambda | C_i^q, l, q) p(C_i^q | \mathbf{e}) \tag{2}$$

The purpose of this section is to define a practical meaning for the components of this equation.

- The probability $p(o_n, \lambda | C_i^q, l, q)$ represents the strength with which the patch votes for the object center (λ). It is inversely proportional to the number of possible interpretations of the patch. This formulation comes from the intuitive idea that the more interpretations a patch has, the more its vote will be unreliable. For example, if a patch representing a car wheel matches in the codebook with both car wheel entries and hood entries, then it will be not discriminative to univocally identify the car center. For this reason the center will have a low probability $p(o_n, \lambda | C_i^q, l, q)$.

- $p(C_i^q | \mathbf{e})$ represents the probability for the patch being correctly explained by a codebook entry C_i^q. The intuitive meaning is the following. Given a pair patch-entry, the perfect explanation of the patch generates a vote for the exact center of the object. Since an entry generally votes for more than one center, the probability of obtaining the correct explanation of the patch decreases when this number increases. Therefore, we can represent the probability $p(C_i^q | \mathbf{e})$ as the inverse of the number of centers voted by the C_i^q entry.

Finally, the $p(o_n, \lambda | C_i^q, l, q) p(C_i^q | \mathbf{e})$ term represents the strength of the vote of each patch.

6 Our proposal

In the approach described above each image patch can cast votes for some object centers **as point coordinates**. This implies that a single vote represents a possible object center at that location and **at that scale**, thus requiring the scale information generated by local descriptors. Since this information is often subject to noise, we try to walk around it by estimating the object scale only from the spatial distribution of the image patches detected. Furthermore, we want the voting procedure to be independent from the object scale. Such a result would allow to operate with a much smaller training sets, since the many images of the same objects at different scales would not be necessary any more. Smaller training sets and, thus, smaller codebooks, noticeably reduce computational costs when comparing

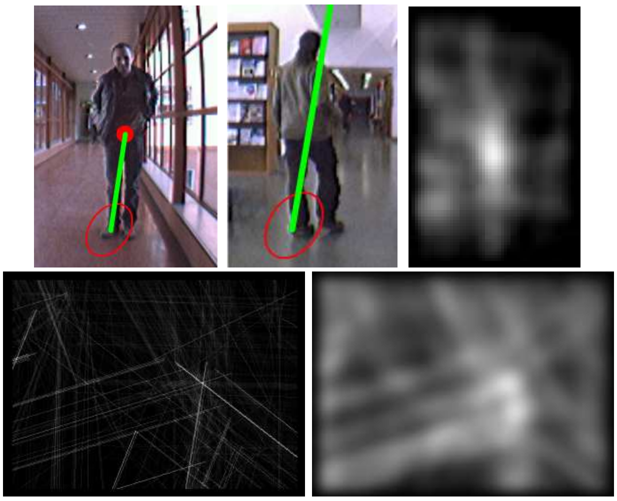

Figure 2. In this figure we show the difference between the original (upper-left) and our (upper-center) approach. Note that in the former each image patch casts a vote for the object center; in the latter each patch casts a votes for a line on which the object center may be found. The result of our method is a voting space like the one shown in the upper-right image. The lower two images show the voting space with all image patches votes (left) and the same space after some image processing (mainly Gaussian blurring) for the enhancement of the local maxima extraction (right).

image patches to codebook entries, since each patch needs to be compared with all entries.

In our approach each patch will no longer vote for a point as the center of the object. Instead, a half straight line will be drawn in the voting space, starting from patch center and going towards the hypothesized object center. Since the region descriptors are scale-invariant, the same object detail should generate similar descriptor values even if scaled or rotated. Thus we save, at training time, the information about the direction of the object center with respect to the patch position in the training image. This information is used at detection time to specify the relative position of the trained object center with respect to the matched patch. In Fig. 2 we show the difference between the original method and ours. The strength of each vote, as described above, is given by the product $p(o_n, \lambda | C_i^q, l, q) p(C_i^q | \mathbf{e})$. As suggested in [4], the procedure for extracting local maxima is performed by the mean shift mode estimator technique.

7 Results

In order to show that our system is capable to detect objects at different scales, regardless of object scale in the training sets, we trained a codebook with only one training image and run the detection system both on the original image, and a set of scaled copies of the same. In Fig. 3 we can see one example of this experiment, where a scaled copy of the original training object is detected without any loss of precision. Please note how different are the interest regions, extracted by the automatic interest region finders, in the two images.

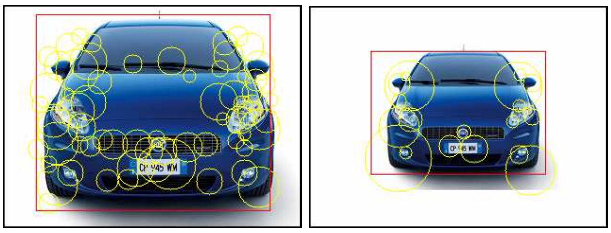

Figure 3. In this figure we show an example of the independency of our approach from the scale of the training objects. Please note that the interest regions, extracted by the interest region finders, are quite different in the two images.

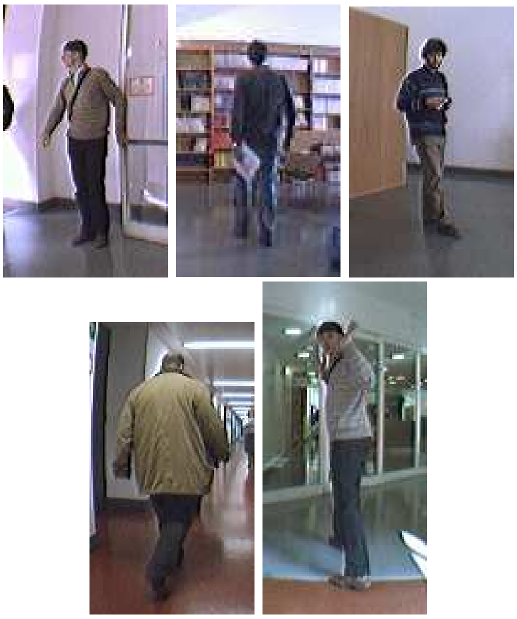

Figure 4. The images used as training set.

We need also to demonstrate that our system is capable to detect objects in cluttered scenes and under difficult lighting, comparably to the original proposal. Out of the many experiments performed, we present here the ones concerning detection of persons. The codebook for these experiments was generated from the 5 images presented in Fig. 4. The results are from images from other datasets. Fig. 5 presents some results for images from the RAWSEEDS datasets, which have been collected by a mobile robot in both indoor and outdoor conditions; they are freely available on the web, see [1]. The 5 training images were also taken from RAWSEEDS project ones, though different from the ones used for the evaluation. The results in Fig. 6 are from datasets from the VISOR repository, also freely available on the web, see [2]. The same codebook, trained on the images in Fig. 4, was used for the images from VISOR.

Note that the number of images in the training set is much smaller than the cardinality of the training sets used by Leibe et al., which were typically including more than 200 images, see e.g., [9], [3]. Furthermore, the training sets only consist of training images and their segmentation masks, without any

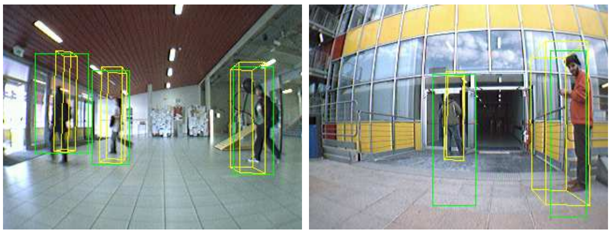

Figure 5. Some results achieved by the detection system on images from the RAWSEEDS project.

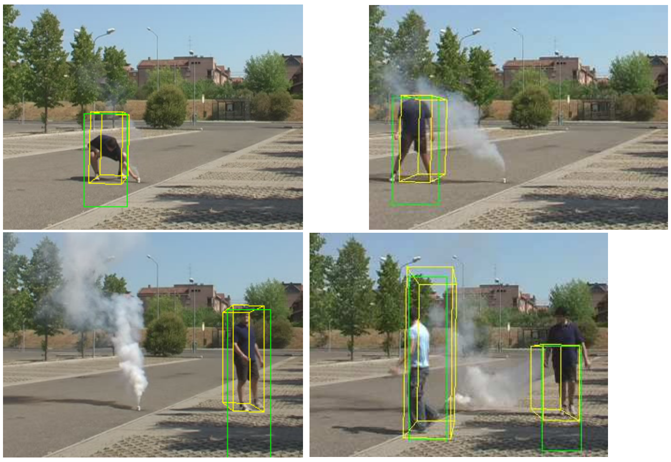

Figure 6. Some results achieved by the detection system on images from the VISOR datasets.

information about object scale.

We collected statistics about one VISOR dataset (visor_1196179837385_movie11_viper.mpg). This dataset includes 1901 images; we processed one image every 20, for a total of 95; some of these images did not include any person. In total we had 80 images including at least one person. The processed images were then checked by human inspection (ourselves) in order to collect the number of false positives and negatives. The results on a total number of 98 person observed in the 80 processed images are: 60 correct detections of the person; 14 ”pure” false positives, i.e., the detection of a non-existant person when another person, correctly detected, was in the image; 20 ”pure” false negatives, i.e., missed detection; 18 false positives with a false negative beside, i.e., an error in the localization of the person. Concerning this aspect, it has to be mentioned that the camera projection parameters were not available, we therefore used the parameters of the camera used in the RAWSEEDS dataset, which introduces gross mistakes. The outcome, in terms of accuracy of localization, can be seen, e.g., in Fig. 8. Therefore, we conclude that these 18 errors are more the consequence of the inaccuracy of the camera calibration than of the inaccuracy of the detector. Concluding, we can consider that we had (60+18) / 98 correct detections = 78/98 = 80%. In Fig. 7 we show some real mistake of the system.

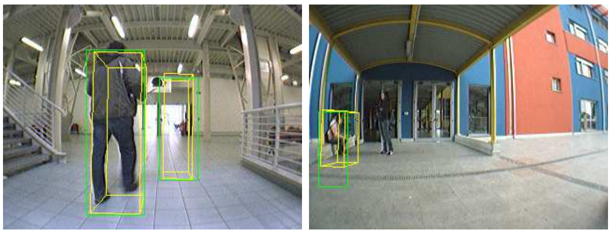

Figure 7. Some mistakes of the detection system: (left) a correct detection with a false positive on its right; (right) a correct detection with a false negative on its right, notice that the correct detection is a seating person, i.e., a situation missing in the training set.

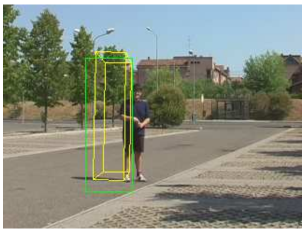

Figure 8. An example of the localization error for the VISOR dataset, turning into both a false negative and a false positive in the same image.

8 Conclusions

In this paper we proposed a substantial improvement to the ISM-based detection system proposed by Leibe et al., e.g., in [4]. Unlike the original approach, our method is independent from the object scale in the training sets. This allows to work with much smaller training sets and to avoid to supply information about scale and size to the trainer. This has been done maintaining the robustness of the original approach. We also showed preliminary results on challenging datasets, challenging with respect to the size of the training set. We are currently working on creating a large statistic from the output of our system that will be used for comparison with other detection systems in a more structured way (ROCs, efficiency, etc.). The qualitative evaluation performed so far, which in our opinion has been very satisfactory, showed that the percentage of correct detections for the tested category of objects (pedestrians) is very high. False positives are quite rare. False negatives are primarily present in low contrast scenes, where interest region finders achieve their worst performance. The size of the training sets (and, thus, of codebooks) are notably smaller than the ones required by the original approach.

Acknowledgements

This work has been partially supported by the European Commission, Sixth Framework Programme, Information Society Technologies: Contract Number FP6-045144 (RAWSEEDS).

References

[1] Website of the RAWSEEDS project, checked Aug. 2009, available at http://www.rawseeds.org.

[2] Website of the VISOR repository, checked Aug. 2009, available at http://www.openvisor.org.

[3] Homepage of Bastian Leibe, checked Aug. 2009, available at http://www.vision.ee.ethz.ch/˜bleibe.

[4] B. Leibe, K. Mikolajczyk, and B. Schiele. Segmentation based multi-cue integration for object detection. In *British Machine Vision Conference (BMVC'06)*, September 2006.

[5] Bastian Leibe, Nico Cornelis, Kurt Cornelis, and Luc Van Gool. Dynamic 3d scene analysis from a moving vehicle. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 0:1–8, 2007.

[6] Bastian Leibe, Aleš Leonardis, and Bernt Schiele. Combined object categorization and segmentation with an implicit shape model. In *In ECCV workshop on statistical learning in computer vision*, pages 17–32, 2004.

[7] Bastian Leibe, Aleš Leonardis, and Bernt Schiele. Robust object detection with interleaved categorization and segmentation. *International Journal of Computer Vision*, 77(1):259–289, May 2008.

[8] Bastian Leibe and Bernt Schiele. Interleaved object categorization and segmentation. In *In BMVC*, pages 759–768, 2003.

[9] Bastian Leibe, Konrad Schindler, Nico Cornelis, and Luc Van Gool. Coupled object detection and tracking from static cameras and moving vehicles. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(10):1683–1698, 2008.

[10] D. G. Lowe. Object recognition from local scale-invariant features. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2, pages 1150–1157 vol.2, 1999.

[11] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(10):1615–1630, October 2005.

[12] Cordelia Schmid. Constructing models for content-based image retrieval. In *In Proc. CVPR*, pages 39–45, 2001.