Local Representations for Multi-Object Recognition

Thomas Deselaers[1], Daniel Keysers[1], Roberto Paredes[2⋆], Enrique Vidal[2⋆], and Hermann Ney[1]

[1]Lehrstuhl für Informatik VI, Computer Science Department
RWTH Aachen - University of Technology, D-52056 Aachen, Germany
{deselaers, keysers, ney}@informatik.rwth-aachen.de

[2]Instituto Tecnológico de Informática
Departemento de Sistemas Informáticos y Computación
Universidad Politécnica de Valencia, E-46022 Valencia, Spain
{rparedes,evidal}@iti.upv.es

Abstract. Methods for the recognition of multiple objects in images using local representations are introduced. Starting from a straight forward approach, we combine the use of local representations with region segmentation and template matching. The performance of the classifiers is evaluated on four image databases of different difficulties. All databases consist of images containing one, two or three objects and differ in the backgrounds which are used. Also, the presence or absence of occlusions of the objects in the scenes is considered. Classification results are promising regarding the difficulty of the task.

1 Introduction

The problem of recognition of single objects in images has been thoroughly studied and satisfactory solutions exist for many applications such as face recognition (e.g. [10]), character recognition (e.g. [7]), and some classification tasks in medical applications (e.g. [5]). The methods used for these tasks are not applicable to the classification of more general images or complex scenes like general images from the world wide web. These images usually contain more than one object, and the objects may be subject to image transformations. A solution to this more general problem is desirable but so far no satisfying approach is known [13]. Methods not explicitly considering objects have been presented, e.g. [2]. In this paper, we examine some new algorithms based on local representations for multi-object recognition which are inherently invariant against translations. Considering the difficulty of the task, the results are promising, but not satisfactory. This implies that further research in this area is needed.

⋆Work supported by the Spanish "Ministerio de Ciencia y Tecnología" under grant TIC2000-1703-CO3-01 and the Valencian OCYT under the grant CTIDIA/2002/80.

2 Local Representations for Classification

The local representation approach is based on the representation of the image by a set of small square subimages taken from different relevant positions of the original image (e.g. determined by the image variance). This method achieves translation invariance and also partially compensates for image occlusions. Using this local representation scheme, each image is represented by several smaller images that are also called local feature images. To classify each test image, a nearest neighbor classifier is applied using a suitable voting scheme. Given a test image, the k-nearest neighbors of all extracted local feature images are searched among the feature vectors computed for the training images. Each neighbor votes for its own class and a vector of votes (per class) is obtained by counting all votes. Following a direct voting scheme, the test image is classified into the most voted class. This sum rule of the votes of each local feature image is similar to the sum rule used in classifier combination theory [3].

The reference training set, consisting of all the local feature images of each training image, usually contains a large number of prototypes. To search the nearest neighbors efficiently, the well known KD-tree data structure is used. An approximate nearest neighbor search is performed instead of the exact search. The search is based on the $(1 + \epsilon)$-approximate nearest neighbor search [1].

Local representations of image objects have the advantage to be invariant against translations of the whole object and of parts of the objects with respect to each other. The local representations approach considered here has been successfully applied to different image object classification tasks [9, 10].

Different approaches for local representations have been proposed, mainly in the image database retrieval literature [11, 12]. In that field, the images are generally completely unconstrained, and representations invariant to translation, scale and rotation, among others, are needed.

3 Multi-Object Recognition using Local Representations

We propose to use local representations to analyze images which contain more than one object and objects which are placed on complex backgrounds. Note that these tasks are considerably harder than the tasks considered so far, where one object is placed on a uniformly colored background.

We consider the following scenario: In training, segmented images representing the objects are given. The test images contain occurrences of these objects in arbitrary position and combination. The difficulty of the test is influenced by occurrence of occlusion and inhomogeneous background.

3.1 Direct Transfer to Multiple Objects

The first idea to use local representations for multi-object recognition would be a direct transfer of the well understood and effective algorithm for single object

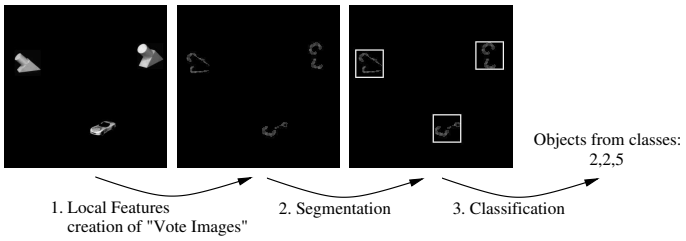

Fig. 1. Step 1: Obtaining the vote image from the test image. Positions where local representations have been extracted are marked white in the vote image. Step 2: The regions of votes are segmented into 3 regions. One for each object. Step 3: A majority vote for each of the 3 regions yields the classification result.

recognition. We slightly modify the classification algorithm and arrive at the following method.

The training process is unchanged and yields a KD-tree with local representations of the training images. Processing of the test images is performed as for single object recognition, arriving at a number of votes for each class. If the number of votes for one of the classes exceeds a certain threshold, the image is assumed to contain an object from this class. Obviously, one of the immediate drawbacks of this method is the inability to classify an image correctly which contains more than one object from one class.

Apart from this drawback, this method also leads to high error rates even on simple classification tasks, although the underlying principle is very effective for classification of single objects.

3.2 Combination with Region Segmentation

Local feature images belonging to an object that is present in the test image are localized within a specific region corresponding approximately to the size of the object in that image. Therefore, it should be required that only votes that are sufficiently close together lead to a joint vote for one object. To fulfill this constraint, we consider a region segmentation process for the votes.

The training process remains the same as for the single object recognition approach. The test image is processed in three steps, creation of vote images, region segmentation, and classification, as illustrated in Fig. 1.

Test images are processed as described for the single object case, yielding votes for one of the classes. Each vote can be uniquely associated with a position in the test image, representing the position the classified local representation was extracted from. The class number voted for can thus be associated with that position, yielding a new image with 'grey' values in the range $1, \ldots, K$, called vote image. These vote images are then segmented into d-connected regions. Regions with a size below a certain threshold are deleted, as they are probably resulting from noise. Those regions which are close enough together and from the same class are joined to one region. Each region is then classified by determining

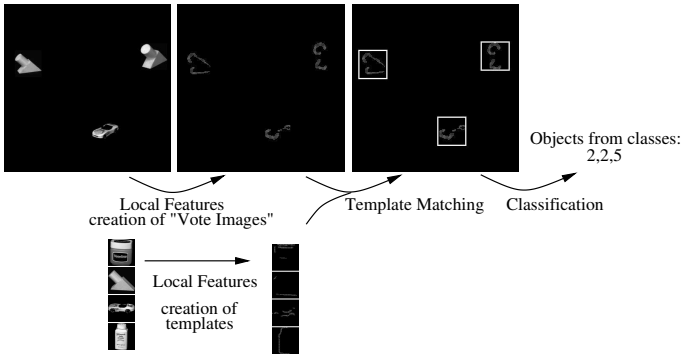

Fig. 2. The training data is processed to obtain the templates for the template matching process. The vote image is calculated from the test image and the template matching process is applied. The classification result is computed from this result.

the winner class using a majority vote procedure and this class is added to the list of classes assumed to be contained in the image.

3.3 Combination with Template Matching

To further refine the algorithm, we propose to use the information about the location of the local representations extracted in the training phase. As this information may enhance discrimination between classes, we apply the following template matching approach.

The extraction of local representations in the training phase is performed as described above. Additionally, for each training image the positions from which local representations are extracted are stored in a template that is equivalent to a binary image marking positions of high variance.

Processing of the test images is again a three step process. First, the vote images are produced in the same way as described above. Second, the template matching is performed by determining the correlation value between the templates produced in training and the vote images. These values are determined for each class and each position in the test image. A completely black template is included as a competing reference to model the background in the case of homogeneous black background. For each position, the highest correlation value and the correlation value for the black template are stored. In the third step, the information gained so far is used to obtain a classification result. If the correlation value for a class is higher than the correlation value for the background, no further detection of an object is allowed within the direct neighborhood and the corresponding class is added to the list of classes of objects assumed to be contained in the image. Template matching can be regarded as implicitly including segmentation, since the regions that are not classified as objects are considered as background. The whole process is illustrated in Fig. 2.

Fig. 3. One training image from each class of the COIL database

This approach and the approach laid out in section 3.2 implicitly detect the object. For classification, a summation of appropriate scores over all positions should be performed, but here the maximum approximation is used instead.

4 Databases

As standardized image databases for multiple object recognition are not publicly available, we generated appropriate databases with different levels of difficulty based on the well known COIL-20 database (Columbia Object Image Library, [8]). It consists of images taken from 20 different 3D-objects viewed from varying positions. Each image contains a single object subject to different illumination conditions. There are 1,440 reference images of size 128×128 pixels available. Examples are shown in Fig. 3.

Four databases were created, two with homogeneous black background and two with complex backgrounds. Each database consists of 400 images with 400×400 pixels. (The databases are available upon request.)

To create the test and training databases, the 1440 images from the COIL training set were split into two parts of 720 images each. The 720 images with even 3D-rotation angles were used as training images and the remaining 720 images were used to create the four test databases. This was done to avoid the occurrence of exactly the same objects in the test and the training dataset.

The databases with homogeneous background are named `black-noocc` and `black-occ`. Every image contains 1, 2, or 3 objects. The background is completely black. The difference between `black-occ` and `black-noocc` is that the objects in `black-occ` may occlude each other and in `black-noocc` not.

The databases with complex background are `dark-occ` and `normal-occ`. The images contain 1, 2, or 3 objects each and occlusions are allowed. The backgrounds are taken from a set of 110 background images. In the `dark-occ` database the images are darkened by 50% to reduce the background variability. Sample images for all databases are shown in Fig. 4.

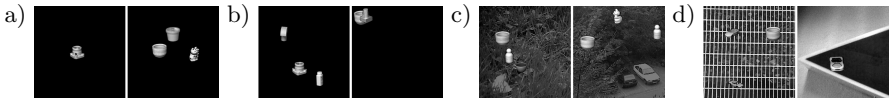

Fig. 4. Examples from a) `black-noocc` b) `black-occ` c) `dark-occ` d) `normal-occ`.

5 Results

For multi-object recognition the error rate known from single object recognition is not a sufficient error measure. This is because images that are classified partly correctly should be distinguished from those classified entirely incorrectly. Therefore, we compute two different error rates inferred from the measures used in speech recognition known as sentence error rate and word error rate. We use the *image error rate* (IER) where an image is only counted as correctly classified if all objects in the image have been recognized (and not more). This measure corresponds to the sentence error rate in speech recognition. The *object error rate* (OER) is similar to the word error rate in speech recognition and we distinguish between insertion (INS), deletion (DEL), and substitution (SUB) errors. The object error rate itself is then defined as the ratio of the minimum number of insertions, deletions, and substitutions to the number of objects in the image. The object error rate can be above 100% if there are more objects detected than actually contained in the image.

With the approach laid out in Section 3.1 the results given in Table 1 were obtained. The results are not satisfactory even on this simple classification task although the approach used performs well for single object classification. With this approach many errors occur because it is obiously not possible to classify an image correctly which contains two objects from the same class and because the position information of the local features is completely disregarded. These experiments were only performed with the database `black-occ` because results were not satisfactory even on this easy task. The threshold given in Table 1 is the threshold used to decide whether an object from one class is in the image given the number of votes per class. Here μ is the mean over the whole vector of votes and σ is the standard deviation. Several experiments were performed, but the best error rate of 28.07% is not sufficient for a task of this low complexity, which can be processed at a low error rate using background segmentation and a nearest neighbor classifier. If no occlusions are allowed, this approach even leads to 0% error rate. Nevertheless, segmentation is an unsolved problem in the presence of complex background and therefore this method is only applicable to images with homogeneous background.

The approach of local representations and region segmentation laid out in Section 3.2 as well as the approach of local representations and template matching described in Section 3.3 were applied to all of the four databases presented in

Table 1. Results for naive approach on `black-occ`. The threshold of minimum number of votes is given in terms of the mean μ and the standard deviation σ for each image.

threshold	INS	DEL	SUB	OER [%]
μ	177	49	96	41.65
$\mu + \sigma$	10	195	34	30.91
$\mu + \frac{1}{4}\sigma$	71	88	65	29.00
2μ	39	106	72	28.07

Table 2. Summary of the results on the different databases using local representations with region segmentation and local representations with template matching.

	region segmentation					template matching				
database	INS	DEL	SUB	OER [%]	IER [%]	INS	DEL	SUB	OER [%]	IER [%]
`black-noocc`	7	60	14	9.69	19.25	3	64	4	7.89	16.25
`black-occ`	29	70	18	14.66	26.75	1	132	2	13.66	25.75
`dark-occ`	458	90	118	88.33	75.00	170	320	11	64.23	60.00
`normal-occ`	1943	28	315	290.84	97.01	57	534	40	80.79	88.22

Section 4. Table 2 contains the results where the free parameters were manually optimized. The approach of local representations and region segmentation suffers mainly from insertion errors while the template matching approach suffers mainly from deletions. The figures show that the computationally more expensive template matching solution yields better results in all cases.

Interestingly, nearly all insertions in `dark-occ` result from just three test images with very high background variance. When using only the 397 test images that do not contain this background, only one insertion remains, with the same number of deletions and only 5 substitutions. This results in an OER of 42.45% and an IER of 59.64%. An impression of the amount of background noise in the vote images is given in Fig. 5. This high amount of noise stresses the need for a better background model.

6 Discussion and Conclusion

We presented an approach to classifying images containing multiple objects using local representations and different enhancements. The results may serve as a starting point for further work in the field of multi-object recognition and need further improvement.

Different improvements to the methods may be considered: In the segmentation step it would be possible to use class and direction dependent distances for joining regions. We also observed that there are some objects which are very similar in some regions (e.g. the two cups). This often results in some parts of the objects being classified as part of an object of another class. This information might also be learned from the training data and used for joining regions.

As suggested in [6] it may lead to better results to consider the whole image and not only some parts in taking the classification decision (holistic approach),

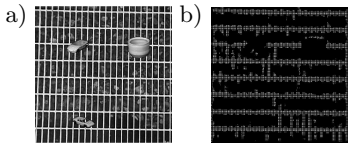

Fig. 5. a) an image from database `normal-occ` b) the vote image to the image from a)

which also includes a better background model. The training phase is not yet fully automated. Here, well segmented data is used for training, which is not always available. It is desirable to learn the representations of the objects from a number of given scenes. First steps into this direction are described in [4, 6].

References

1. S. Arya, D.M. Mount, N.S. Netanyahu, R. Silverman, and A.Y. Wu. An optimal algorithm for approximate nearest neighbor searching. *Journal of the ACM*, 45(6):891–923, November 1998.
2. H. Burkhardt and S. Siggelkow. Invariant features in pattern recognition – fundamentals and applications. In C. Kotropoulos and I. Pitas, editors, *Nonlinear Model-Based Image/Video Processing and Analysis*, pages 269–307, Wiley, 2001.
3. R.P. Duin, J. Kittler, M. Hatef, and J. Matas. On combinig classifiers. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 20(3):226–239, March 1998.
4. B.J. Frey and N. Jojic. Transformation-invariant clustering using the EM algorithm. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 25(1):1–17, January 2003.
5. D. Keysers, J. Dahmen,H. Ney, B. Wein, and T. Lehmann. Statistical framework for model-based image retrieval in medical applications. In *Journal of Electronic Imaging*,12(1):59–69, January 2003
6. D. Keysers, M. Motter, T. Deselaers, and H. Ney. Training and recognition of complex scenes using a holistic statistical model. In *DAGM 2003, Pattern Recognition, 25th DAGM Symposium, Magdeburg, Germany*, September 2003. This volume.
7. D. Keysers, R. Paredes, H. Ney, and E. Vidal. Combination of tangent vectors and local representations for handwritten digit recognition. In *SPR 2002, Int. Workshop on Statistical Pattern Recognition*, *Lecture Notes in Computer Science*, Vol. 2396, pp. 538–547, Windsor, Ontario, Canada, August 2002.
8. S. Nene, S. Nayar, and H. Murase. Columbia object image library: COIL-100. Technical Report CUCS-006-96, Department of Computer Science, Columbia University, New York, February 1996.
9. R. Paredes, D.Keysers, T. Lehmann, B. Wein, H. Ney, and E. Vidal. Classification of medical images using local representations. In *Bildverarbeitung für die Medizin*, pp. 171–174, Leipzig, Germany, March 2002.
10. R. Paredes, J. Perez-Cortes, A. Juan, and E. Vidal. Local representations and a direct voting scheme for face recognition. In *Workshop on Pattern Recognition in Information Systems*, pp. 71–79, Setúbal, Portugal, July 2001.
11. C. Schmid and R. Mohr. Local grayvalue invariants for image retrieval. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(5):530–535, May 1997.
12. C. Shyu, C.E. Brodley, A.C. Kak, A. Kosaka, A. Aisen, and L. Broderick. Local versus global features for content-based image retrieval. In *Proc. IEEE Workshop on Content-Based Access of Image and Video Libraries*, pp. 30–34, June 1998.
13. A.W.M. Smeulders, M. Worring, S. Santint, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, December 2000.