Local Discriminant Regions Using Support Vector Machines for Object Recognition

David Guillamet and Jordi Vitrià

Centre de Visió per Computador-Dept. Informàtica, Universitat Autònoma de Barcelona, 08193 Bellaterra, Barcelona, Spain
Tel. +34 93-581 30 73 Fax. +34 93-581 16 70
{davidg, jordi}@cvc.uab.es,
WWW home page: http://www.cvc.uab.es/~davidg

Abstract. Visual object recognition is a difficult task when we consider non controlled environments. In order to manage problems like scale, viewing point or occlusions, local representations of objects have been proposed in the literature. In this paper, we develop a novel approach to automatically choose which samples are the most discriminant ones among all the possible local windows of a set of objects. The use of Support Vector Machines for this task have allowed the management of high dimensional data in a robust and founded way. Our approach is tested on a real problem: the recognition of informative panels.

Keywords: Support Vector Machines, Local Appearance, Computer Vision, Object Recognition.

1 Introduction

Visual recognition of objects is one of the most challenging problems in computer vision and artificial intelligence. Historically, there has been an evolution in recognition research from 3D geometry to 2D image analysis. Early approaches to object recognition were based on 3D geometry extraction [4,6,7] but the process of extracting geometrical models of the viewed objects leads to a difficult problem and fragile solutions. Furthermore, these 3D geometry based techniques can be made to work in a controlled environment but their application to real environments generate several problems.

An alternative to 3D reconstruction is to remain in the 2D image space working with measurements of the object appearance. Turk and Pentland [13] used subspace methods to describe face patterns with a lower-dimensional space than the image space. The appearance of a face is the combination of its shape, reflectance properties, pose in the scene and illumination conditions, and they use the Principal Component Analysis (PCA) technique to obtain a reduced space. Murase and Nayar [8] extended this idea using different instances of an object captured in a wide range of conditions (several viewpoints and illumination conditions) and used them to represent the object as a trajectory in the PCA space. Recognition is achieved by finding the trajectory that is closest to the projection

F.J. Ferri et al. (Eds.): SSPR&SPR 2000, LNCS 1876, pp. 550–559, 2000.

of an input image in the PCA space formed by all objects. Black and Jepson [1] have addressed the problem of partial occlusion by using robust estimation techniques in conjunction with PCA based projections. However, PCA based techniques suffer from several difficulties. Mainly, an image projection to a PCA based space depends on the precise position of the relevant objects, on the intensity and shape of background zones, and on intensity and color of illumination. Given that PCA technique treats its inputs (in our particular case, images) in a global manner, the relevant objects must be detected, segmented and normalized to manage them in the same way. This problem leads to a difficult process that can be unsolvable in certain cases.

PCA analysis can be done on different image representation data. Hancock [5] found that the results of applying a PCA projection over a set of natural images was nearly the same as a set of Gaussian derivative filters. Rao and Ballard [10] ascertained the results of Hancock with an extensive collection of images containing equal proportions of natural and man-made stimuli. Thus, the Gaussian derivative filters are natural basis functions useful for general-purpose object recognition and objects can be expressed as a set of reduced response vectors obtained as the result of an application of these filters.

Current research on visual recognition of objects is focused on the identification of physical objects from arbitrary viewpoints under arbitrary lighting conditions and being situated in an undetermined scene with possible occlusions. The presence of occlusions and different backgrounds can be minimized using local measurements instead of global treatments [3,9]. Some recent approaches [9,12] focus on the fact that an object can be divided into small windows but only a subset of them are necessary to identify an object. The basic idea is to process an object obtaining a set of reliable points (those that can contain reliable information) and selecting some of them getting a discriminant subset. Ohba and Ikeuchi [9] use a measure of *trackability* to obtain an initial set of candidate points that are reduced with an eigenspace projection. Schmid and Mohr [12] use the well-known Harris detector to obtain their candidate points. However, some authors [3] consider the application of their descriptors on a predefined grid instead of on a set of selected interest points. This criteria is justified by the fact that objects captured in non controlled environments manifest some inestabilities in the procedure of extracting interesting points.

Our approach is similar in spirit to the work of Ohba and Ikeuchi [9] who extract a subset of local windows of an object to identify it. We select a subset of local windows in a different way: using the Support Vector Machines (SVM) technique that provides an optimal separating hyperplane between two different classes with an intrinsic distance notion that can be exploited. Ikeuchi's method does not depend on the the classification task given that a threshold distance must be defined in order to refuse similar local windows. The user must tune this threshold according to the objects nature, i.e, if the database is composed of similar objects, the threshold must be different from the one considered with a database of several kinds of objects. Our approach does not need a tunning parameter that reflects the possible similarities of the training objects given that the

SVMs technique can extract and detect those training points that are conflictive (support vectors) without any external help. We have chosen a reduced set of objects took in a non controlled environment to test the basis of our approach. A set of local windows has been extracted from each object in order to minimize the future effects of occlusions and possible background problems and a sorted list of the local windows has been done to show the discriminant information that each local window contains. Objects have been normalized in a constant image size in order to consider their local windows in the same way.

2 Support Vector Machines

A two-class problem can be defined [14] as:

$$(x_1, y_1), \ldots, (x_n, y_n), x \in \Re^d, y \in \{+1, -1\} \tag{1}$$

where each example has an assigned value (+1 or −1) depending on the class that it belongs. In such particular case, SVM technique can be used to seek for an optimal separating hyperplane $D(x)$ defined as:

$$D(x) = (w \cdot x) + w_0 \tag{2}$$

Where w_0 is a threshold value and w is a weight vector. Figure (1) [2] shows a graphic representation of an optimal hyperplane.

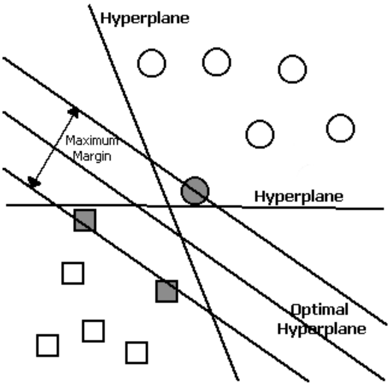


Fig. 1. Optimal separating hyperplane versus different possible hyperplanes. The optimal hyperplane is that hyperplane that defines a maximum margin between the support vectors. Support vectors are indicated in grayvalues.

Depending on the number of support vectors, Vapnik [14] states a generalization upper bound:

$$E_n[\text{Error}] \leq \frac{E_n[\text{Number of support vectors}]}{n-1} \tag{3}$$

This expression estates that the expectation of the number of Support Vectors obtained during training on a training set of size n, divided by $n-1$, is an upper bound on the expected probability of test error.

The difficulty of separating a certain class can be minimized if this class is mapped to a higher dimensional space where the SVM technique can improve its separability property. Such mapping can be done without affecting the complexity of SVM decision boundaries given that SVM technique is independent of the new space dimensionality, which can be very large (or even infinite). SVM optimization takes advantage of the fact that all the operations that have to be carried out in such new high dimensional space (feature space) can be done in the input space via the evaluation of a kernel function $k(x, y)$ defined by the inner product between support vectors and vectors in the input space:

$$k\left(x,y\right)=\left(\Phi\left(x\right)\cdot\Phi\left(y\right)\right) \tag{4}$$

where $\Phi\left(x\right)$ is a mapping function that maps an input vector x to a feature space vector. Different kernels can be used:

- **Linear kernel**: It is a simple inner product in the input space:

$$k\left(x,y\right)=x\cdot y \tag{5}$$

- **Polynomial kernel of degree d**: The optimal hyperplane will be defined as a polynomial expression:

$$k\left(x,y\right)=\left[\left(x\cdot y\right)+r\right]^{d} \tag{6}$$

- **RBFs kernel**: The optimal hyperplane will be defined as a radial basis function:

$$k\left(x,y\right)=exp\left\{-\frac{\left|x-y\right|^{2}}{\sigma^{2}}\right\} \tag{7}$$

Expression (2) can be expressed in terms of support vectors and a kernel function as:

$$D\left(x\right)=\sum_{i}^{n}\alpha_{i}^{*}y_{i}k\left(x_{i},x\right)+w_{0}^{*} \tag{8}$$

where α_i^* are the lagrangian coefficients of the quadratic optimization.

2.1 Multiclass Classification Using Support Vector Machines

Dealing with a $k-class$ classification problem, a set of binary classifiers $f^1, \ldots, f^k$ has to be constructed, each trained to separate one class from the rest, and combine them by doing a multi-class classification according to the maximal output obtained by expression (8), i.e by taking:

$$\text{argmax}_{j=1,\ldots,k}D^{j}\left(x\right), \quad \text{where} \quad D^{j}\left(x\right)=\sum_{i=1}^{n}y_{i}\alpha_{i}^{j}\cdot k\left(x,x_{i}\right)+w_{0}^{j} \tag{9}$$

2.2 Hyperplane Distances

Our approach is based on the fact that each training point has a relative distance to the optimal calculated hyperplane. The optimal hyperplane is defined by a set of support vectors, which are the closest and the most conflictive training points . Thus, extracting the most distant training points, we can obtain those points with a low probability of being misclassified.

Figure (2) schematizes our approach. Given a distribution of training points that have to be separated using Support Vector Machines, an optimal hyperplane is calculated. Figure (2.a) shows a complex distribution that it is not totally separable with a conflictive region where reside points of different classes. In such particular case, applying a linear kernel to obtain an optimal separating hyperplane implies that 10 training vectors are considered support vectors (as shown in figure (2.b)). Given that support vectors are conflictive points, we do not consider them as relevant training points and we sort the rest of training points depending on their distance to the optimal hyperplane.

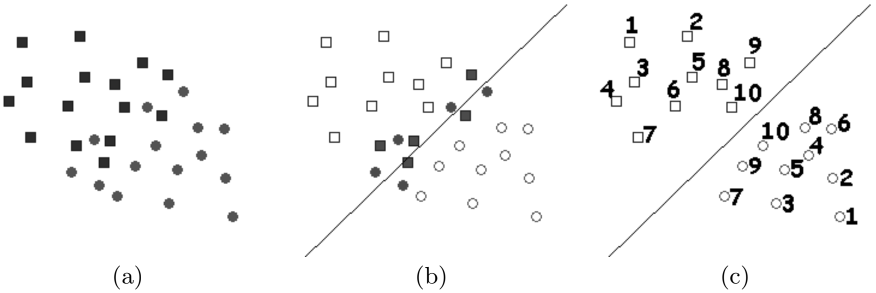
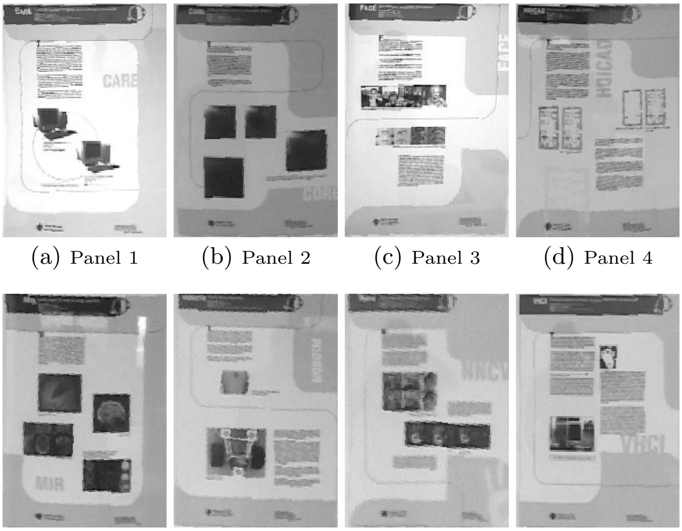


Fig. 2. (a) Original Distribution. (b) Optimal Linear Hyperplane and Support Vectors. (c) Training points that are not support vectors sorted depending on their distance to the optimal hyperplane. The most distant point is the most important point.

3 Experimental Results

The main aim of our approach is the extraction of the most discriminant local windows belonging to a set of objects. A local window division of an object is justified by the fact that background and occlusion influences will be minimized. However, this local window division leads to generate a very large database of local windows and requires a prohibitive amount of memory to store all of them. The basic idea is that not all the local windows of an object are necessary to recover the identity of an object given that most of them can be redundant or can not contain discriminant information. In our case, we divide each object in a set of local windows (different divisions are considered) and we have sorted all of them by considering their discriminant information. Depending on the final application and the memory space, the final user must select how many discriminant local windows has to use.

In our particular case, we have chosen a reduced set of 8 panels situated on the walls of our building captured in different viewpoints and lighting conditions. We have done a panel mapping operation in order to obtain a set of panels with a known size and we have divided each panel using a predefined grid [3]. This grid defines a set of interesting points where we have applied a set of descriptors. In our case, we have decided to use a set of Gaussian derivatives filters as local image descriptors given that this image representation is speacially suited to visual discrimination [11]. Figure (3) shows all the 8 different panels used in our experiments.



(a) Panel 1 (b) Panel 2 (c) Panel 3 (d) Panel 4

(e) Panel 5 (f) Panel 6 (g) Panel 7 (h) Panel 8

Fig. 3. 8 different panels mapped to a standard window size of 175 × 250 pixels.

All panels have been randomly divided to make up the training and testing set. The training and testing set are composed by 29 different instances of each panel. We have considered 7 different scales for our Gaussian filters and used up to third order derivatives. So, each interesting point has a response vector of 70 dimensions. The Gaussian window size applied to all the following experiments is constant (37×37) and different sizes of panels are considered in order to study how affect its neighborhood. Before obtaining a response vector, we have applied an illumination intensity normalization that consists of substracting from each local window its gray value mean and considering its variance as:

$$\tilde{I}(x) = \frac{I(x) - \mu}{\sigma} \tag{10}$$

being μ the region intensity mean and σ its variance.

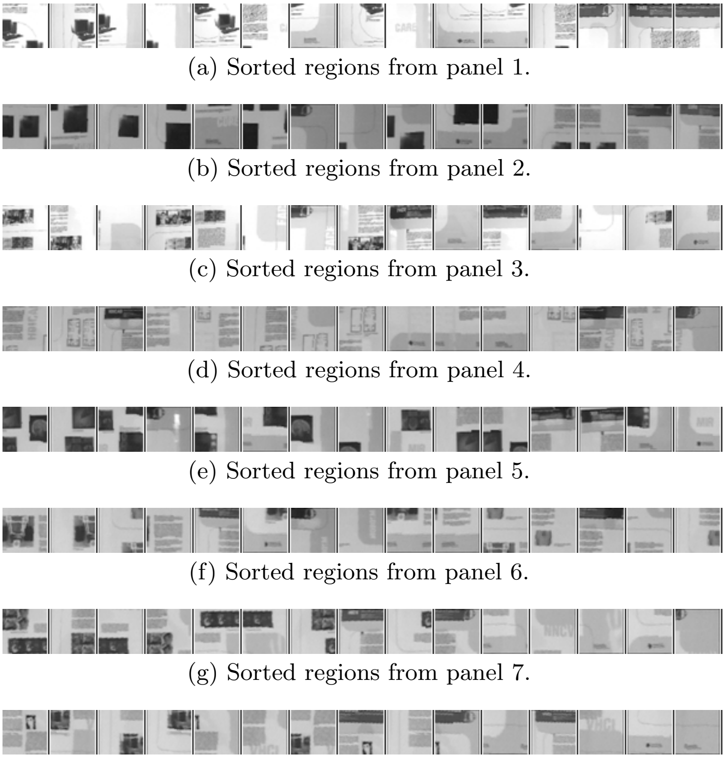
3.1 Semiglobal Experiment

Each panel is divided in 15 regions (3 horizontally and 5 vertically) obtaining a training and testing set of 3480 vectors. Panels are resampled to a window size of 85 × 125 pixels. Different kernels have been trained to separate each panel from the rest obtaining the results shown in table (1).

Table 1. Results obtained dividing each panel in 15 regions (8 panels with 29 instances divided in 15 regions = 3480 different response vectors). Different kernels have been tested obtaining a good performance using a RBF kernel $\sigma = 0.5$ with a test error rate of 4.87 %. Each table box has 3 different numbers: The number of Support Vectors obtained, the number of misclassified training vectors and the number of misclassified testing vectors. Last column shows the final test error obtained considering the maximal output obtained from the eight classifiers (see expression (9)).

Kernel	Analyzed Feature	Panel 1	Panel 2	Panel 3	Panel 4	Panel 5	Panel 6	Panel 7	Panel 8	Error rate
Linear	# SVs	372	400	530	895	648	825	768	875	10.41 %
	Train Error	211	206	226	679	461	600	552	650	
	Test Error	241	187	194	524	320	502	429	520	
Polynomial degree $d = 2$	# SVs	128	66	171	304	123	306	301	246	6.25 %
	Train Error	116	0	149	313	47	425	378	213	
	Test Error	166	35	189	327	122	402	365	258	
Polynomial degree $d = 3$	# SVs	127	75	145	277	127	265	277	238	7.53 %
	Train Error	227	0	227	359	8	224	355	242	
	Test Error	316	28	275	392	98	270	370	298	
RBF $\sigma = 0.0005$	# SVs	481	438	638	856	772	888	887	912	9.88 %
	Train Error	189	94	229	309	287	412	427	391	
	Test Error	224	93	200	367	266	408	412	402	
RBF $\sigma = 0.005$	# SVs	411	158	521	841	498	828	716	868	6.03 %
	Train Error	116	19	166	246	117	377	270	332	
	Test Error	166	15	170	315	116	378	298	350	
RBF $\sigma = 0.05$	# SVs	230	78	313	525	227	554	505	450	5.39 %
	Train Error	28	2	37	122	15	93	122	90	
	Test Error	85	17	108	208	68	161	142	150	
RBF $\sigma = 0.5$	# SVs	133	97	155	240	179	205	249	227	4.87 %
	Train Error	0	0	0	12	0	1	8	9	
	Test Error	63	28	74	131	64	112	130	143	

Choosing the best kernel (the one with less support vectors and a low error rate), in such case the RBF Kernel with $\sigma = 0.5$, we have applied the idea mentioned in section 2.2 to extract the most discriminant regions of each panel (considering the distance of each region to the optimal hyperplane). Figure (4) shows the sorted list of the discriminant zones from each panel according to the distance of each region to the hyperplane obtained with the RBF kernel $\sigma = 0.5$.



(a) Sorted regions from panel 1.

(b) Sorted regions from panel 2.

(c) Sorted regions from panel 3.

(d) Sorted regions from panel 4.

(e) Sorted regions from panel 5.

(f) Sorted regions from panel 6.

(g) Sorted regions from panel 7.

(h) Sorted regions from panel 8.

Fig. 4. Sorted regions according to the best kernel calculated in table (1). It can be seen that the first discriminant regions of all the panels are those who belong to central zones and the last ones are those who are homogeneous zones (bright zones that are conflictive) or belong to the panel regions where the title is (given that all the panels titles have a similar tonality).

3.2 Semilocal Experiment

In such experiment, each panel is divided in 45 regions (5 horizontally and 9 vertically) obtaining a training and testing set of 10440 vectors. Panels are re-sampled to a window size of 125 × 175. In that case, each window contains less information about its neighborhood than the previous experiment given that the panel size is bigger than before. Different kernels have been trained to separate each panel from the rest and for lack of space, table (3.2) only shows the best one.

Table (3.2) shows that the final error rate and the number of support vectors increase given that in this particular case, each region contains less information than in the previous experiment. Having more local regions, white and homogeneous zones increase because each local window considers a more local

Table 2. Results obtained dividing each panel in 45 regions ($8 \times 45 \times 29 = 10440$ different response vectors). In this table, only the best kernel result is shown.

Kernel	Analyzed Feature	Panel 1	Panel 2	Panel 3	Panel 4	Panel 5	Panel 6	Panel 7	Panel 8	Error rate
	# SVs	981	368	1291	1392	1232	792	601	912	
RBF	Train Error	218	16	356	871	812	281	210	304	6.8 %
$\sigma = 0.5$	Test Error	387	140	509	903	874	393	415	544	

neighborhood. However, discriminant zones are concentrated in the same regions that in figure (4).

3.3 Local Experiment

In such experiment, each panel is divided in 153 regions (9 horizontally and 17 vertically) obtaining a training and testing set of 12240 vectors (only 10 instances of each panel are considered). The panel window size considered in such case is 175×250. The neighborhood considered in that case is lesser than in the previous experiment. Different kernels have been trained to separate each panel from the rest and for lack of space, table (3.3) only shows the best one.

Table 3. Results obtained dividing each panel in 153 regions ($8 \times 153 \times 10 = 12240$ different response vectors). In this table, only the best kernel result is shown.

Kernel	Analyzed Feature	Panel 1	Panel 2	Panel 3	Panel 4	Panel 5	Panel 6	Panel 7	Panel 8	Error rate
	# SVs	2201	1926	2039	1882	2109	2321	2552	2118	
RBF	Train Error	854	781	832	698	864	917	1021	869	13.08 %
$\sigma = 0.5$	Test Error	971	896	991	817	1005	1011	1221	1067	

Table (3.3) shows that the final error rate and the number of support vectors increase much more than before. The reason is that there are a lot of regions that are homogeneous or similar regions to other panels regions that have appeared as a consequence of a more local neighborhood treatment. However, discriminant zones are concentrated in nearly the same regions that in figure (4).

4 Conclusions

An automatic discriminant method has been developed in order to extract discriminant regions from a determined set of different objects. Objects have been divided in various levels of regions considering different neighborhood hierarchies and Support Vector Machines have been used to extract the most discriminant ones. Despite of the several experiments performed using different neighborhood hierarchies, all of them show that the most discriminant information is always localized in the central regions of a panel leading to consider the method as a robust one. The results are satisfactory enough to consider that Support Vector

Machines is a reliable technique to be applied to such discrimination problems. In our experiments, each object is divided in different regions considering several neighborhood hierarchies. Our method sorts these regions according to their distance to an optimal hyperplane calculated by the SVMs considering different kinds of kernels. The final window size has to be selected according to the degree of possible occlusions (a major degree of occlusions will imply that the regions with extensive neighborhoods can not be used given that they will surely be partially occluded).

5 Acknowledgments

This work is supported by CICYT and EU grants TAP98-0631 and 2FD97-0220, and Comissionat per a Universitats i Recerca del Departament de la Presidència de la Generalitat de Catalunya

References

1. M. Black and A. Jepson. Eigentracking : Robust and tracking of articulated objects using a view-based representation. In *Proc. of 4th European Conference on Computer Vision*, volume 1, pages 329–342, Cambridge, April 1996.
2. V. Cherkassky and F. Mulier. *Learning From Data.* Wiley - Interscience, New York, 1998.
3. V. C. de Verdière and J. L. Crowley. Visual recognition using local appearance. In *Proc. ECCV'98*, 1998.
4. W. Grimson. *Object Recognition by Computer.* MIT Press, 1990.
5. P. J. B. Hancock, R. J. Baddeley, and L. S. Smith. The principal components of natural images. *Neural Networks*, 3:61–70, 1992.
6. D. Huttenlocher and S. Ullman. Recognizing solid objects by alignement. In *Proc. IEEE ICCV'87*, pages 102–111, 1987.
7. D. G. Lowe. Three-dimensional object recognition from single two dimensional images. *Artificial Intelligence*, 31:355–395, 1987.
8. H. Murase and S. Nayar. Learning and recognition of 3d objects from appearance. In *Proc. IEEE Qualitative Vision Workshop*, pages 39–49, 1993.
9. K. Ohba and K. Ikeuchi. Detectability, uniqueness and reliability of eigen windows for stable verification of partially occluded objects. *IEEE Transaction on PAMI*, 19(9):1043–1048, September 1997.
10. R. P. Rao. *Dinamic Appearance-Based Vision.* PhD thesis, University of Rochester, 1997.
11. B. Schiele and A. Pentland. Probabilistic object recognition and localization. Technical Report 499, MIT Media Laboratory, Perceptual Computing, 1999.
12. C. Schmid and R. Mohr. Combining greyvalue invariants with local constraints for object recognition. In *Proc. IEEE CVPR'96*, pages 872–877, 1996.
13. M. Turk and A. Pentland. Face recognition using eigenfaces. In *Proc. of IEEE Conf. on CVPR'91*, pages 586–591, June 1991.
14. V. Vapnik. *The Nature of Statistical Learning Theory.* Springer Verlag, New York, 1995.