MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

Higher-Order Dependencies in Local Appearance Models

David Guillamet, Baback Moghaddam, Jordi Vitria

TR2003-93 December 2003

Abstract

We propose a novel local appearance modeling method for object detection and recognition in cluttered scenes. The approach is based on the joint distribution of local feature vectors at multiple salient points and factorization with Independent Component Analysis (ICA). The resulting densities are simple multiplicative distributions modeled through adaptative Gaussian mixture models. This leads to computationally tractable joint probability densities which can model high-order dependencies. Our techinque has been initially tested under different natural and cluttered scenes with different degrees of occlusions with promising results. With this present work, we provide a large statistical test with the MNIST digit database in order to demonstrate the improved performance obtained by explicit modeling of high-order dependencies.

International Conference on Image Processing (ICIṔ03)

Published in: *International Conference on Image Processing* (ICIP’03), September 2003.

HIGHER-ORDER DEPENDENCIES IN LOCAL APPEARANCE MODELS*

David Guillamet[1], *Baback Moghaddam*[2], *Jordi Vitrià*[1]

[1]Computer Vision Center, Dept. Informàtica
Universitat Autònoma de Barcelona
08193 Bellaterra, Barcelona, Spain
{davidg,jordi}@cvc.uab.es

[2]Mitsubishi Electric Research Laboratories
Cambridge, MA 02139, USA
baback@merl.com

ABSTRACT

We propose a novel local appearance modeling method for object detection and recognition in cluttered scenes. The approach is based on the joint distribution of local feature vectors at multiple salient points and their factorization with Independent Component Analysis (ICA). The resulting densities are simple multiplicative distributions modeled through adaptative Gaussian mixture models. This leads to computationally tractable joint probability densities which can model high-order dependencies. Our techinque has been initially tested under different natural and cluttered scenes with different degrees of occlusions yielding promising results. In this work, we provide a large statistical test with the MNIST digit database in order to demonstrate the improved performance obtained by explicit modeling of higher-order dependencies.

1. INTRODUCTION

For appearance based object modeling in images, the choice of method is usually a trade-off determined by the nature of the application and the availability of computational resources. Existing object representation schemes provide models either for global features [1], or for local features and their spatial relationships [2, 3, 4, 5]. With increased complexity, the latter provides higher modeling power and accuracy. Among various local appearance and structure models, there are those that assume rigidity of appearance and viewing angle, thus adopting more explicit models [4, 2, 6]; while others employ stochastic models and use probabilistic distance and matching metrics [5, 7, 3].

Recognition and detection of objects is achieved by the extraction of low level feature information in order to obtain accurate representations of objects. Extracted low level features must be carefully selected and it is often necessary to use as many salient features as possible. But one of the most common problems encountered is the computational cost of dealing with high dimensional data as well as the intractability of joint distributions of multiple features.

We propose a novel local appearance and color modeling method for object detection and recognition in cluttered scenes. The approach is based on the joint distribution of local feature vectors at multiple salient points and their factorization with Independent Component Analysis (ICA). Using the new statistically independent space to create $k = 3$ tuples ($k = 3$ salient points) of the most salient points of an object, we are able to obtain a set of joint probability densities which can model high-order dependencies. In order to model the tuple space, we use an adaptative Gaussian mixture model based on the Minimum Description Length (MDL)[8] criterion to properly estimate our probability densities.

We have tested our method in a closed environment where we detect real objects with different configurations, poses and levels of occlusions. Our technique is however able to manage with real, complex and cluttered environments and we present some results of object detection in these scenarios with promising results. Furthermore, a very large and statistically significant experiment (using the MNIST database) illustrates the generality of feature representations in our scheme as well as explicitly demonstrating the advantage of modeling higher-order statistics of our tractable joint distributions.

2. METHODOLOGY

We propose to use an adaptative Gaussian mixture model as a parametric approximation of the joint distribution of image features of local color and appearance information at multiple salient points.

Let i be the index for elementary feature components in an image, which can be pixels, corner/interest points [9, 10], blocks, or regions in an image. Let x_i denote the feature vector of dimension n at location i. x_i can be as simple as {R,G,B} components at each pixel location, some invariant feature vectors extracted at corner or interest points [11, 2, 12], transform domain coefficients at an image block, and/or any other local/ regional feature vectors.

For model-based object recognition, we use the *a posteriori* probability

$$\mathrm{m}ax_l P(M_l|T) \tag{1}$$

where M_l is the object model and $T = \{x_i\}$ represents the features found in the test image. Equivalently, by assuming equal priors, classification/detection will be based on maximum likelihood testing:

$$\mathrm{m}ax_l P(T|M_l) \tag{2}$$

For the class-conditional density in equation (2), it is intractable to model dependencies among all x_i's (even if correspondence is solved), yet to completely ignore these dependencies is to severely limit the modeling power of the probability densities. Objects frequently distinguish themselves not by individual regions (or parts), but by the relative location and comparative appearance of these regions. A tractable compromise between these two modeling extremes (which does not require correspondence) is to model the joint density of all k-tuples of x_i's in T. Figure (1) shows a general scheme of our methodology.

*This work is supported by Comissionat per a Universitats i Recerca del Departament de la Presidencia de la Generalitat de Catalunya and Ministerio de Ciencia y Tecnologia grant TIC2000-0399-C02-01.

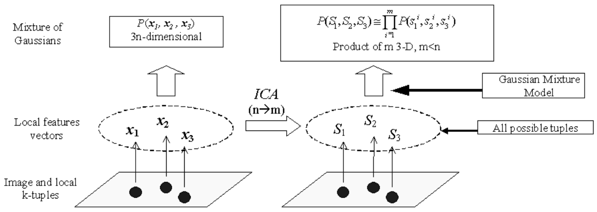

Fig. 1. System diagram for k-tuple density factorization using ICA and Gaussian mixture models.

2.1. Joint Distribution of k-tuples

Instead of modeling the total joint likelihood of all $x_1, x_2, \ldots x_I$, which is an $(I \times n)$-dimensional distribution, we model the alternative distribution of all k-tuples as an approximation:

$$P(\{(x_{i_1}, x_{i_2}, \ldots, x_{i_k})\}|M_l) \tag{3}$$

This becomes a $(k \times n)$-dimensional distribution, which is still intractable (Note: $k < n$ and $k << I$). We can use multi-dimensional histograms as an approximation of the joint distribution of image features with, i.e 20 histogram bins along each dimension, and such a framework would require $20^{(k\times n)}$ bins. Therefore, a factorization of this distribution into a product of low-dimensional distributions is required. We achieve this factorization by transforming x into a new feature vector S whose components are (mostly) independent. This is where Independent Component Analysis (ICA) comes in.

2.2. Density Factorization with ICA

ICA originated in the context of blind source separation [13, 14] to separate "independent causes" of a complex signal or mixture. It is usually implemented by pushing the vector components away from Gaussianity by minimizing high-order statistics such as the 4^{th} order cross-cumulants. ICA is in general not perfect therefore the IC's obtained are not guaranteed to be completely independent.

By applying ICA to $\{x_i\}$, we obtain the linear mapping

$$x \approx AS \tag{4}$$

and

$$\begin{aligned} &P(\{(S_{i_1}, S_{i_2}, \ldots, S_{i_k})\}|M_l) \\ \approx &\prod_{j=1}^{m} P(\{(s^j_{i_1}, s^j_{i_2}, \ldots, s^j_{i_k})\}|M_l) \end{aligned} \tag{5}$$

where A is a n-by-m matrix and S_i is the "source signal" at location i with nearly independent components (Note: $m < n$). The original high-dimensional distribution is now factorized into a product of m k-dimensional distributions, with only small distortions expected. We note that this differs from so-called "naive Bayes" where the distribution of feature vectors is assumed to be factorizable into 1D distributions for each component. Without ICA the model suffers since in general these components are almost certainly statistically dependent.

After factorization, each of the k dimensional factored distributions becomes manageable if k is small, e.g., $k = 2$ or 3. Moreover, matching can now be performed individually on these low-dimensional distributions and the scores are additively combined to form an overall score.

Figure (2) is a graphical model showing the dependencies between a pair of 3-dimensional feature vectors x_1, x_2. The joint distribution over all nodes is 6-dimensional and all nodes are (potentially) interdependent. The basic approach towards obtaining a tractable distribution is to remove intra-component dependencies (vertical and diagonal links) leaving only inter-component dependencies (horizontal links). Simultaneously, we seek to reduce the number of observed components from $n = 3$ to a smaller number $m = 2$ of "sources". Ideally, a perfect ICA transform results in the graphical model shown in the right diagram where the pair S_1, S_2 only have pair-wise inter-component dependencies. Therefore, the resulting factorization can be simply modeled by 2D histograms or Gaussian mixture models[1].

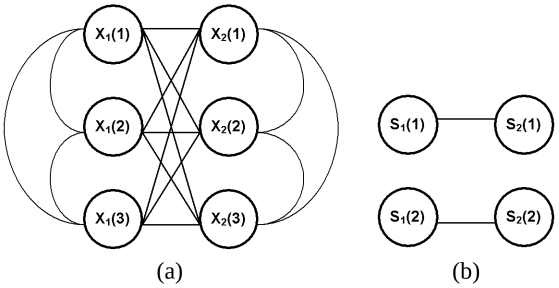

Fig. 2. Graphical models: (a) fully-connected graph denoting no independence assumptions (b) the ICA-factorized model with pairwise only dependencies.

2.3. Class-Conditional ICA

When object recognition consists of having r different classes and each class represented using a specific ICA model, it turns out that the combination of all ICA models must be normalized. In [15] a class-conditional ICA (CC-ICA) model is introduced that, through class-conditional representations, ensures class-conditional independence. The basic CC-ICA model is estimated from the training set for each class. If W_r and s_r are the projection matrix and the independent components for class C_r with dimensions $M_r \times N$ and M_r respectively, then $s^r = W^r(x - \overline{x}^r)$ where $x \in C_r$ and $\overline{x}^r$ is the class mean, estimated from the training set. Most ICA methods require, or at least advise, data whitening as preprocessing. Since some simple denoising is also recommended, dimensionality reduction and whitening through PCA is very common practice as a preprocessing stage for ICA. In this case, W^r can be decomposed as $W^r = B^r E^r$, where E^r is the $M_r \times N$ PCA whitening matrix and B_r the ICA unmixing matrix. Also $v^r \overset{def}{=} E^r(x - \overline{x}^r)$ is the whitened data. Assuming the class-conditional representation actually provides independent components, we have that the class-conditional probability in transformed space noted as $p^r(s) \overset{def}{=} p(s^r)$ can now be expressed in terms of unidimensional densities,

$$p(v|C_r) = \nu_r p^r(s) = \nu_r \prod_{m=1}^{M_r} p^r(s_m) \tag{6}$$

[1] We should note that in practice with an approximate ICA transform, the diagonal links of the original model are less likely to be removed than the vertical ones.

with $\nu_r = (\int p^r(s)ds)^{-1}$, a normalizing constant. Actually, from the change of variables rule, $\nu_r = |\det(B^r)|$. See [15] for more information.

3. EXPERIMENTAL RESULTS

For our experiments, we used a Harris operator [10, 12] to detect interest points and extracted the first 9 differential invariant jets [11] at each point as the corresponding feature vector x. Using these jets as our features results in a local appearance model which is not only invariant to in-plane rotation (and translation) but is also robust with respect to partial occlusions as we shall see later. We must emphasize however that our methodology is not restricted to differential invariant jets and can in principal be used for any local set of features, for example, color, curvature, edge-intensity, texture moments or even shape descriptors. We then performed ICA to get $m < 9$ independent components for the feature vectors (jets). We then used $k = 3$, resulting in a set of 3D Gaussian mixture models which were used to model 3-tuple joint component densities. Once an ICA space is defined, we used the definition of class-conditional ICA of Equation (6) in order to obtain the probability of a tuple belonging to each training class.

We evaluated this approach with real laboratory scenes where deformable objects can appear under various configurations, poses and occlusions. Figure (3) shows the objects and images used for this experiment. Two different objects with similar colors but different shapes were learned in order to detect them in a complex environment. As noted in this figure (3), objects can be hard to recognize since they contain different levels of occlusions and can be seen under different poses. Despite these difficulties, objects are correctly detected thus indicating the degree of robustness in our system.

As seen, our method works for real imagery where various configurations, poses and occlusions of objects may appear. The previous experiment was carried out using a hand-selected $k = 3$ tuple model, however proper model order selection remains an issue. Increasing the tuple model would imply to incorporate high-order dependencies between the detected keypoints but, also, an increment of computational resources. Thus, a good trade-off between the model order and the computational resources required must be found and analyzed.

We have also applied our object recognition scheme in a totally different context in order to demonstrate how to integrate mutiple instances into a single model and that increasing of the tuple order does in fact lead to improved performance. Our scheme is designed to work with different keypoint dependencies (from $k = 1$ to $k = \infty$) but when we consider a high model order, more dependencies are generated and the complexity is also increased. We want to test different high-order dependencies (modifying the k parameter) with a huge database to obtain statistically reliable and significant evidence of the behaviour: Performance of our technique is increased when we incrementing the order of our joint distributions. We chose the MNIST [16] digit database because it contains a huge number of training and testing samples ($60,000$ training samples and $10,000$ testing samples), so we can statistically verify that incrementing the order of our models will lead to better recognition rates. We must note that our scheme is not especially adapted to work with the MNIST database rather it is a general technique for use in complex and cluttered scenes with the presence of occlusions. Our main goal here is to explore how increasing tuple order affects to the recognition rates using a well-known and large database.

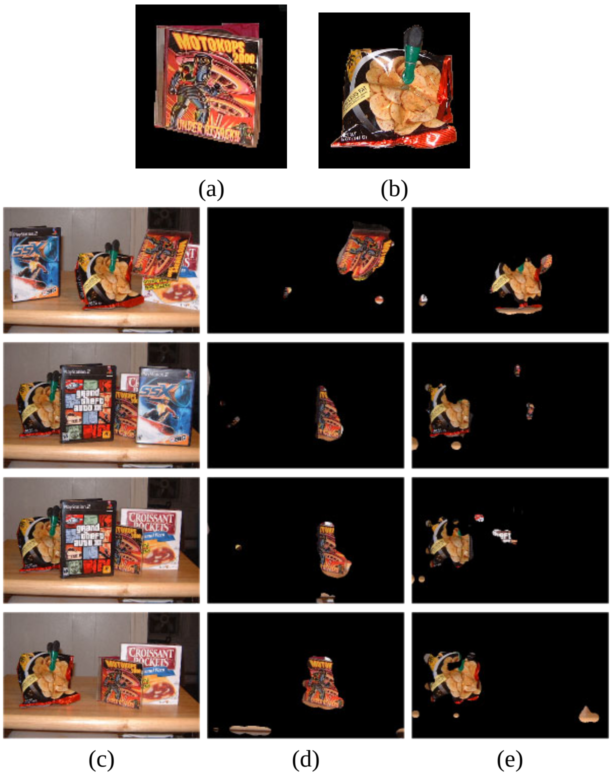
(a)
(b)
(c)
(d)
(e)

Fig. 3. Two objects (a) and (b) with similar colors but differing in shape used to train our models. First column (c) contains 4 testing images where the two learned objects are present under different occlusions and poses. Columns (d) and (e) show the detection maps for objects (a) and (b), respectively.

In particular, features were extracted from hand-written MNIST digits using the same technique as in [17] where they obtain a set of shape histograms for each digit. In our case, each digit is represented by a set of 75 points sampled from the shape contour (75 pixel locations sampled from the output of the Canny detector). Having 75 pixel locations, we have represented each location using a shape histogram (exactly the same as in [17]), so that each digit is represented by 75 shape histograms of 60 dimensions. In order to find the “right” ICA dimension to reduce our feature vectors, we did a k-NN (with k=5) based classification using the original shape histograms taking a reduced set of training and testing samples (200 training samples per each digit and the first $5,000$ testing samples) using the χ^2 test statistic (as in [17]) as a distance metric. Also, a k-NN (with k=5) based classification was done using the ICA projected feature vectors between $d = 5$ to $d = 50$ ICA dimensions with the same training and testing set as before using the L_1 norm as a distance metric in order to evaluate which is the ICA dimension that preserves the same recognition rates of the original space. The dimension found by the experiments to be the most suitable one for our ICA scheme was 25, which was used thereafter.

We have tested two different approaches: (1) learn an adaptative mixture model per each training instance and (2) learn an adaptative mixture model per each digit class. Our factored $k = 2$ and $k = 3$ high-order models generate a huge number of tuples. In this particular case, when using $k = 2$ tuples, we generate an order of $5,000$ tuples per each digit and when using $k = 3$ tuples, $100,000$ possible tuples are generated. We have not tested higher dependencies because the number of possible tuples is really huge. Thus, having a huge number of tuples, we have to choose a reduced number of them in order to train our factored models. We have tested three different approaches: (1) a random selection of tuples, (2) tuples with close keypoints and (3) tuples with distant keypoints. Since we are working with digits, tuples created from having 3 close keypoints would not be as significant as having tuples created from 3 distant keypoints because digits are homogeneous representations (they do not have changes in texture or colors and the neighborhood of two close keypoints does not change significantly) and relevant changes are manifested when considering two different shape contexts (distant keypoints).

Our random tuple selection consists of randomly selecting $1,000$ $k = 2$ tuples and $5,000$ $k = 3$ tuples to learn our adaptative Gaussian mixture models. When considering tuples with near keypoints, we take the $1,000$ $k = 2$ tuples and $5,000$ $k = 3$ tuples with the closest keypoints. Finally, we select tuples with distant keypoints and we take the first most distant $1,000$ $k = 2$ tuples and $5,000$ $k = 3$ tuples. For our experimental tests, we used 500 training samples per each digit ($5,000$ in total) and all the testing MNIST set ($10,000$ digits). Experimental results are shown in Table (1) where we can clearly see that incrementing the order of our models leads to an improvement in the recognition rates. Interestingly enough, we note also that there seems to be little difference between the two different approaches of handling multiple training instances: using one model/instance *vs.* one model/class. Also, it can be seen that a good selection of tuples leads to obtain improved recognition rates.

k tuples		1 Model / Instance	1 Model / Class
$k = 1$ tuples		74.23%	71.85%
$k = 2$ tuples	Random	83.14%	82.03%
	Near	78.36%	75.47%
	Distant	85.09%	83.71%
$k = 3$ tuples	Random	91.57%	91.13%
	Near	88.67%	87.92%
	Distant	93.03%	91.85%

Table 1. Experiments done using 500 digits per each class as training and $10,000$ testing digits. First column of results indicates each training instance is represented by one model ($5,000$ training models) and second column indicates each class is represented by one model (10 training models in total). Recognition rates are represented according to the tuple order used and tuple selection technique.

Using the nearest neighbor classifier (k-NN with k=3) in the original space of shape histograms with the χ^2 test statistic, we obtain a recognition rate of 75.87%. We are not using any kind of point matching between our features as in [17] and it should be obvious that our method is not best-suited for the MNIST database (that is not the point here) but we do notice the improvement of our factored distribution models from $k = 1$ to $k = 3$. We should emphasize that even though we do not achieve the best reported recognition rates for the MNIST, our factored models with $k = 3$ are not only significantly better than $k = 1$ but also better than using k-NN in the original space of shape histograms (a recognition rate of 75.87%).

4. CONCLUSIONS

A novel probabilistic modeling scheme was proposed based on factorization of high-dimensional distributions of local image features. Our framework was initially tested using real imagery where objects were correctly detected under different configurations, poses and occlusions. These experiments with complex and cluttered scenes demonstrate that this technique is well suited to object detection and localization tasks in natural environments. Finally, a large experiment with the MNIST digit database was performed in order to validate the underlying assumption that increasing the high-order dependencies of our factored distributions does in fact lead to improved performance. Also, it has been demonstrated that a good selection of learning tuples is an important factor to take into account.

5. REFERENCES

[1] M. Swain and D. Ballard, “Color indexing,” *International Journal Computer Vision*, vol. 7, no. 1, pp. 11–32, 1991.

[2] C. Schmid and R. Mohr, “Local grayvalue invariants for image retrieval,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 5, pp. 530–534, 1997.

[3] P. Chang and J. Krumm, “Object recognition with color cooccurrence histograms,” in *Proc. of International Conference in Computer Vision and Pattern Recognition*, 1999.

[4] H. Schneiderman and T. Kanade, “Probabilistic modeling of local appearance and spatial relationships for object recognition,” in *Proc of CVPR*, 1998, pp. 45–51.

[5] J. Huang, S.R. Kumar, M. Mitra, W.J. Zhu, and R. Zabih, “Image indexing using color correlograms,” in *Proc. of CVPR*, 1997.

[6] B. Moghaddam and A. Pentland, “Probabilistic visual learning for object representation,” *IEEE Transactions on PAMI*, vol. 19, no. 7, pp. 696–710, 1997.

[7] B. Moghaddam, H. Biermann, and D. Margaritis, “Regions-of-interest and spatial layout in content based image retrieval,” in *Proc. of European Workshop on CBMI*, 1999.

[8] H. Tenmoto, M. Kudo, and M. Shimbo, “Mdl-based selection of the number of components in mixture models for pattern recognition,” in *Proc. of SSPR/SPR*, 1998, pp. 831–836.

[9] R. Deriche and G. Giraudon, “A computational approach for corner and vertex detection,” *International Journal Computer Vision*, vol. 10, no. 2, pp. 101–124, 1993.

[10] C. Harris and M. Stephens, “A combined corner and edge detector,” in *Proc. 4th Alvey Vision Conference*, 1988, pp. 147–151.

[11] J.J. Koenderink and A.J. van Doorn, “Representation of local geometry in the visual system,” *Biological Cybernetics*, vol. 55, pp. 367–375, 1987.

[12] C. Schmid, R. Mohr, and C. Bauckhage, “Comparing and evaluating interest points,” in *Proc. of International Conference in Computer Vision*, 1998.

[13] P. Comon, “Independent component analysis - a new concept?,” *Signal Processing*, vol. 36, pp. 287–314, 1994.

[14] C. Jutten and J. Herault, “Blind separation of sources,” *Signal Processing*, vol. 24, pp. 1–10, 1991.

[15] M. Bressan, D. Guillamet, and J. Vitria, “Using an ICA representation of local color histograms for object recognition,” In *Pattern Recognition*, vol. 36, no. 3, pp. 691–701, 2003.

[16] Y. LeCun, *The MNIST DataBase of Handwritten digits*, http://yann.lecun.com/exdb/mnist/index.html.

[17] S. Belongie, J. Malik, and J. Puzicha, “Shape matching and object recognition using shape contexts,” *IEEE Transactions on PAMI*, vol. 24, no. 24, pp. 509–522, 2002.