DISTORTION TOLERANT PATTERN RECOGNITION USING INVARIANT TRANSFORMATIONS AND HIERARCHICAL SOFM CLUSTERING

Jouko Lampinen

Lappeenranta University of Technology
Department of Information Technology
PO Box 20, SF-53851 Lappeenranta, Finland
E-mail: jouko@deneb.it.lut.fi

A block structured pattern recognition system is proposed whose distortion tolerance is build up gradually by successive blocks in a pipeline architecture. The system consists of only feedforward neural networks, allowing efficient parallel implementation.
The feature extraction part of the system is based on distortion tolerant Gabor transformation and minimum distortion clustering by hierarchical self-organizing feature maps (SOFM). Classification is carried out by a subspace classifier based on Principal Component networks using local histograms of the previous level features as the input data.
The proposed pattern recognition system requires very small amount of supervised training due to many unsupervised stages refining the data to be suitable for classification.
The function of hierarchical SOFM is discussed and it is shown to be more than just vector quantization: hierarchical SOFM produces natural clusters whose shape is matched to the input data distribution.

1 Introduction

Pattern recognition task has been traditionally divided into two distinct stages, feature extraction and classification. Many neural network classifiers have proved to be general enough to produce arbitraryly complex class boundaries according to the training data, disemphasizing the role of feature extraction.

However, images of real world objects are often subject to various sources of distortions, making object's class boundaries very complex in the gray-scale space. To train a reasonably distortion tolerant classifier the training data is required to span the whole space of objects with all possible distortions and their combinations and collecting this data is practically impossible.

In this paper a neural pattern recognition system is proposed with emphasis in distortion tolerant feature extraction. The proposed system consists of blocks connected into a pipeline, so that the distortion tolerance of the final classification emerges gradually from contributions of the successive blocks.

There are three main blocks in the system:

1. **Fixed layer** doing Gabor transformation to the image on different resolutions using self-similar kernels.
2. **Unsupervised layer** clustering the Gabor coefficients to natural clusters while introducing minimal additional distortion. The clustering is accomplished by a hierarchical self-organizing feature map.
3. **Supervised layer** carrying out the classification. The classifier is based on linear PCA-networks and locally collected histograms of the previous level outputs.

[1]Work on this paper was supported by Tekes Grant 4196/1988 under the Finsoft project.

Role of the Gabor transformation in the system is explained earlier in [7] and is shortly reviewed here. The 2D Gabor kernels are complex valued orientation specific band-pass filters that are optimally localized in spatial domain and frequency domain making them very suitable for extracting frequency contents from as small area as possible.

Amplitude spectrum of an object is rather robust against several types of distortions, like relative translations of parts of the object that mainly affect the phase spectrum, see e.g. [2],[3]. Thus, feature vectors coming out from the first layer contain Gabor-filter amplitudes in different orientations and different frequencies.

2 Clustering the Features with Hierarchical SOFM

One of the most important features of the self-organizing feature map, SOFM [6], is the topology preserving mapping from high dimensional space to low dimensional map surface. SOFM is today used mostly as training strategy for vector quantizer, and in the final use the form of the map (order of the reference vectors) is not important. SOFM training gives, however, invaluable advantages in producing vector quantizer: the large neighbourhood in the beginning of training helps map to escape local minima of the minimized function, and updating several vectors for each training sample speeds up the convergence of minimization [12].

Direct advantage from the ordering of the map can be achieved in hierarchical systems of SOFM's, e.g. in a hierarchical vector quantization [8]. In the following a reasoning is given why the hierarchical SOFM is suitable for clustering also: it will produce "natural" clusters whos form is dynamically matched to the probability density structure of the input samples. If there are low dimensional structures in the data they will form long elongated clusters, while uniform data is mapped to circular clusters.

Let the SOFM lattice be D-dimensional set of reference vectors $\mathbf{m_r}$ indexed by a D-vector $\mathbf{r}$ and let the neighbourhood function (form of the "activity bubble" on the map [6]) be shift invariant over the lattice, $h : \mathrm{I\!R}^{\mathrm{D}} \mapsto \mathrm{I\!R}$, $h(\mathbf{r}, \mathbf{q}) = h(\mathbf{r} - \mathbf{q})$.

SOFM training [6] is a gradient descent method to minimize a cost function [8], [12].

$$D(\mathbf{m}) = \sum_{\mathbf{q}} \sum_{\mathbf{r}} h(\mathbf{r} - \mathbf{q}) \sum_{\mathbf{x} \in A_{\mathbf{q}}} p(\mathbf{x}) ||\mathbf{x} - \mathbf{m_r}||^2 \tag{1}$$

where $A_\mathbf{r}$ is the set of points $\mathbf{x}$ that are mapped to unit $\mathbf{r}$. $A_\mathbf{r}$ is a convex volume limited by normal hyperplanes of nearest units in the space. In the following we denote set $\{\mathbf{x} | \mathbf{x} \in A_\mathbf{r}\}$ by $\mathbf{x_r}$. Let $p(\mathbf{x}) = 1/N$ where N is the number of training points.

Eq.1 means, that the map minimizes the mean distance from input vector to reference vectors that are in the $h()$-neighbourhood of $\mathbf{m_{r(x)}}$.

In a local minimum of the SOFM training the gradient for unit $\mathbf{q}$ is zero.

$$G_\mathbf{q} = \sum_{\mathbf{r}} h(\mathbf{q} - \mathbf{r}) \sum_{\mathbf{x_r}} (\mathbf{x_r} - \mathbf{m_q}) = \sum_{\mathbf{r}} h(\mathbf{q} - \mathbf{r})(\mathbf{C_r} - \mathbf{m_q}) = 0$$

where $\mathbf{C_r}$ is the mean of the points mapped to $\mathbf{r}$, $\mathbf{C_r} = \frac{1}{N_{\mathbf{x_r}}} \sum_{\mathbf{x_r}} \mathbf{x_r}$. Let $\sum_\mathbf{q} h(\mathbf{q}) = 1$. Then

$$\mathbf{m_q} = \sum_{\mathbf{r}} h(\mathbf{r} - \mathbf{q}) \mathbf{C_r}$$

and

$$\mathbf{m} = \mathbf{h} * \mathbf{C}$$

where $\mathbf{h}$ is a vector containing the neighbourhood feedback values in the discrete lattice points and $*$ is the D-dimensional discrete convolution. To guarantee convergence $\mathbf{h}$ must be low-pass filter giving smooth $\mathbf{m}$. In the final stage of the SOFM training $h()$ approaches delta function and $\mathbf{m} \approx \mathbf{C}$ (and in the limit $\mathbf{m} \rightarrow \mathbf{C}$ and optimal vector quantization results). Then

$$\sum_{\mathbf{r}} h(\mathbf{q} - \mathbf{r})(\mathbf{C_r} - \mathbf{m_q}) \approx \sum_{\mathbf{r}} h(\mathbf{q} - \mathbf{r})(\mathbf{C_r} - \mathbf{C_q})$$

So, for small neighbourhood the SOFM minimizes the weighted distance of points mapped to the neighbourhood. When the neighbourhood is smoothly decreased, map finds as compact set of points as possible to be mapped to each neighbourhood during the training. If the data contains compact structures (with small total volume of spheres required to contain the data points) then at some neighbourhood size the map can minimize the distance of points mapped to the neighbourhood by placing the units sequentially along the structure. This is how the map finds and preserves the local ordering relations of the training samples. On the other hand, if the data are uniformly distributed, then the units will end up close to each others on every neighbourhood size, turning the map to a space filling surface having the same structure on every scale [6].

2.1 Hierarchical SOFM

In a hierarchical SOFM a second layer map $\mathbf{m}^2$ quantizes the outputs $\mathbf{r}$ of the first layer map $\mathbf{m}^1$. Units of $\mathbf{m}^2$ divide the units in $\mathbf{m}^1$ into equiprobable convex areas corresponding approximately to some neighbourhood size used during training in $\mathbf{m}^1$.

Now the region mapped to one $\mathbf{m}^2$ unit is the set of all the regions of the corresponding units in $\mathbf{m}^1$, forming arbitraryly complex convex or concave volume limited by hyperplanes.

This clustering corresponds roughly to classical clustering methods (e.g. ISODATA) when the distance of a point from the cluster is defined as weighted distance from all the points in the cluster, but with HSOFM the clustering requires only two nearest neighbour searches instead of computing all the distances and the extent of weighting is adapted automatically to the data distribution ($h()$ is defined in the map coordinates).

In addition, the system minimizes the total quantization error introduced by both the layers [8]. Among clustering methods this is a rather unique balance between preserving maximal information and collapsing the data to natural clusters.

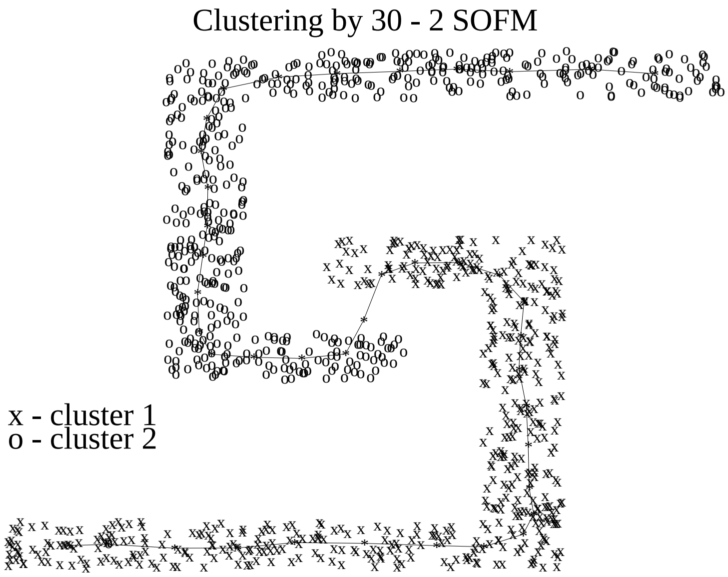

Figure 1. shows a very simple example about the capabilities of hierarchical SOFM in clustering. The input probability distribution contains continuous structures that are partially interleaved. Clustering by the HSOFM is based on continuous compact areas, and it thus easily finds the natural cluster boundary. The continuous line shows the first layer 30-unit map oriented to the data distribution. The second layer had only 2 units splitting the $\mathbf{m}^1$ to two halves as shown in Fig.1.

In the proposed system the output of the feature extraction stage is a ”feature image” $c(\mathbf{p})$ where each image location $\mathbf{p}$ is mapped by the HSOFM to unit c corresponding to a cluster in the space of Gabor amplitudes in different orientations and resolutions.

3 Distortion Tolerant Classification with Histograms and PCA-networks

The feature extraction based on Gabor-filters is not shift invariant, but instead relates the features to as small area as possible increasing probability that the features come from only one object. The distortion invariance of a any method is always limited to the spatial support of its receptive field, and any distortions larger than that will affect the output.

In the feature extraction used here the problem is not very prominent since the hierarchical clustering absorbs most small variations and expands the spatial support of clear details, see [7]. Consequently, good recognition results have been achieved by direct template matching in the feature space [7]. But since the features are local the problem of distortions in feature positions is there and should be taken into account in classification.

There are three main approaches to tackle these types of distortions

1. *Requirement that the important features must appear in correct order* based usually on editing distances [11].
2. *Planar graph matching algorithms.* Using with Gabor functions see e.g. [2].
3. *Requirement that the important features must appear inside certain area* by a translation invariant transformation applied to the area. The transformation may be e.g. a histogram of the underlying features. Distortion tolerance of the Neocognitron [4] is based on this category, since a C-cell turns on if any of the pattern detecting S-cells in its receptive field is on. According to recent results, by weighting the histogram with a smooth window the invariance property is enhanced [1].

The last approach was chosen in the current system because of most promising test result from the three categories and also because of possibility for fast parallel implementation with only feedforward networks. Both the other categories require time consuming iterations to find the optimal match.

3.1 Local Feature Histogram

The input vector for the classifier is a histogram of HSOFM codes weighted by a Gaussian window:

$$\mathbf{w}_i(\mathbf{p}) = \sum_{c(\mathbf{p}')=i} \Omega(\mathbf{p} - \mathbf{p}')$$

where $\mathbf{p}$ is a position vector in the feature image $c(\mathbf{p})$ and $\Omega()$ is the weight function $\Omega(\mathbf{p}) = \exp\left(-\frac{\mathbf{p}^T\mathbf{p}}{2R^2}\right)$ and R determines the size of the window. A similar approach has been developed for handling 1D time dependent data by SOFM [5]: for each time step the input is quantized by a SOFM, and the history of the data is stored as exponentially weighted histogram over time of the outputs of the SOFM. Then these histories are learned by another SOFM.

For small number of objects there will be specific units in the HSOFM map for details of each object, and the recognition can be based on histograms from the whole object. For larger or more complicated objects more position information must be maintained. Then the input vector is collected from several locations with smaller windows and the vectors are stacked together.

Using the Gaussian weighted histogram as the final feature complies with using the Gabor filter amplitudes as the primary features since they are basically histograms of certain frequencies inside a Gaussian window.

3.2 Classification with PCA-networks

Subspace methods [9] are efficient and computationally simple classification methods for histogram and spectrum type data. Each class is defined as a linear subspace and represented by a set of orthogonal base vectors. Output of the net is the projection of the input vector onto the class subspace, and an unknown input vector is classified according to the largest output.

The neural subspace classifier is based on the principal component network [10]. In [10] it is shown that with the learning rule

$$\frac{dM}{dt} = \alpha(\mathbf{I} - MM^T)\mathbf{x}\mathbf{x}^T M \tag{2}$$

the columns of matrix M become orthonormal and span the same subspace as the k largest eigenvalues of the autocorrelation matrix of the input data ($\mathrm{E}\{\mathbf{x}\mathbf{x}^T\}$) i.e. the space of the k largest principal components of the inputs.

Now each class i is represented by a subspace M_i. The most simple approach for finding the class subspaces is to compute the largest principal components for each class separately using Eq.2.

Better results are obtained by the Learning Subspace Method LSM [10] that finds subspaces minimizing classifications errors. LSM requires only a small change in update rule of Eq.2 (compare to LVQ vs. SOFM). If classification of an input is a and the correct class is b then update M_a with negative learning parameter and M_b with positive one.

In the current problem 1 and 2 dimensional class subspaces where found to be sufficient. In most of the experiments even the principal component subspaces were able to separate the classes but for better resolution the LSM was used.

4 Experimental Results

In the unsupervised clustering block the 8 Gabor filter responses were quantized by a 10×10 map separately on each resolution and then two successive resolutions were combined by a 100-unit 1D map. Now the histogram of these features is a 100 dimensional vector that was passed over to the subspace classifier. The width of the Gaussian window Ω was about 50 pixels ($R = 20$) for face recognition.

Fig. 2. shows some of the test images. For each person two sample images were selected and a PCA-network was trained with the samples. Table 1. shows the network outputs for the classes T,K and J, respectively. The training samples are denoted by bold face letters in the table.

The Gabor filters are not scale invariant, since scaling of the image shifts the whole spectrum and the amount of energy in the filter pass-bands changes depending on the frequency contents on other bands. The system is merely scale tolerant; some features span, due to the HSOFM, long clusters reaching the invariance over the 1 octave interval, and some features are much more sensitive to scaling. Thus, and according to most experiments, scaling causes projections to all the classes to decrease but their ratios still facilitate correct classification (see e.g. J2 in Fig.2).

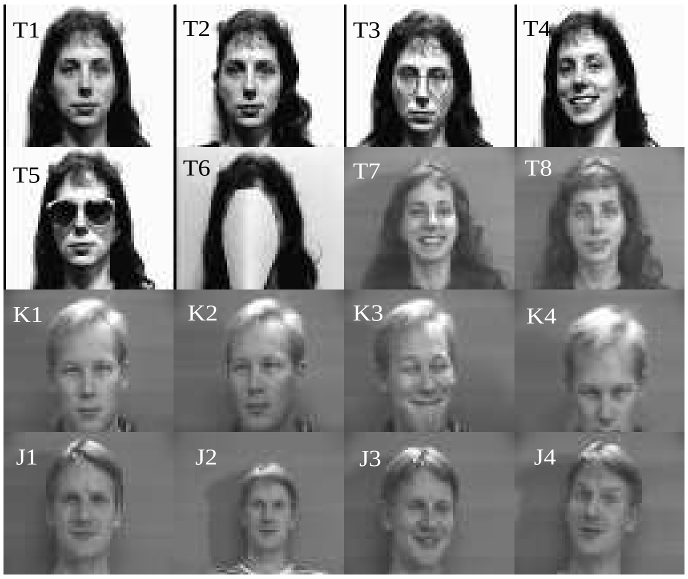

Figure. 2. Some test images (each one 128×128 pixels)

T1	T2	**T3**	T4
0.9413	0.8687	0.9524	0.6623
0.2429	0.1074	0.1039	0.1402
0.2546	0.1182	0.0996	0.1447
T5	T6	T7	T8
0.3804	0.4751	0.7026	0.6575
0.3283	0.2040	0.3728	0.3631
0.2200	0.1682	0.4532	0.3650
K1	**K2**	K3	**K4**
0.2182	0.1671	0.4749	0.1100
0.6283	0.7861	0.5223	0.8200
0.5061	0.5802	0.4719	0.5644
J1	J2	**J3**	**J4**
0.1497	0.2235	0.1158	0.1661
0.5113	0.2236	0.6297	0.5488
0.7355	0.4563	0.8066	0.8380

Table 1. Classifier outputs for the images of Fig.2.

The system is not shift invariant since moving the window Ω changes the histogram, but to the extent of Ω it is shift tolerant, small shifts causing small changes in projections. In Fig.3. projections to the three subspaces are plotted along a horizontal line in an image containing figures T8, K1 and J1 in a row, respectively. Shift tolerance is very good and classification is correct even near the midplanes of two objects.

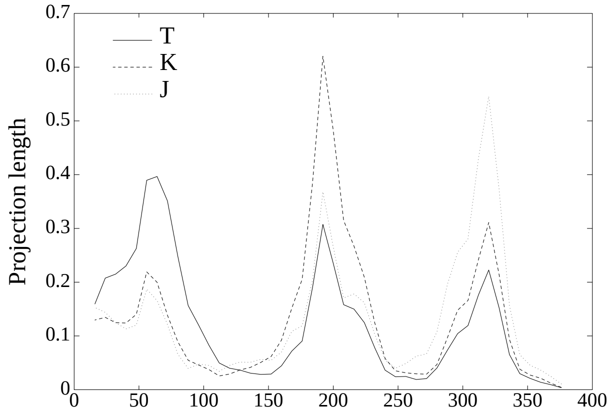

Figure 3. Class projections as functions of histogram window location.

5 Conclusions

Using hierarchical SOFM to map continuums in data space to clusters while causing minimal additional quantization error provides a feature space that is distortion tolerant in small scale and offers specific enough features to facilitate classification based on local feature histograms.

Whether this scheme can provide sufficient recognition capability to serve as a universal set of features in general scene analysis, and how the SOFM's should be structured, will require further analysis. For limited number of object classes the feature space seems to be adequate for reliable and very fast recognition with minimal amount of supervised learning.

The current system has no feedback in any stage. Adding feedback to control focusing of attention when proceeding from coarse levels to higher resolutions will be subject to future study.

Acknowledgements

The author wishes to thank prof. Erkki Oja for fruitful discussions throughout the work.

References

[1] E. Barnard, D. Casasent, Shift Invariance and the Neocognitron, Neural Networks, Vol. 3, 1990 , pp. 403-410.

[2] J. Buhman, M. Lades and C. von der Malsburg, Size and Distortion Invariant Object Recognition by Hierarchical Graph Matching, Proc. IEEE IJCNN-90 Vol. 2. San Diego, CA, Jan. 1989, pp. 411-416.

[3] K.A. Flator and S.T. Toborg, "An Approach To Image Recognition Using Sparse Filter Graphs", Proc. IEEE IJCNN-89 Vol. 1. Washington DC, June 18-22, 1989, pp. 313-320.

[4] K. Fukushima, Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by a shift in position, Biological Cybernetics, 36, 1980, pp.193-202.

[5] J. Kangas, Time dependent self-organizing maps for speech recognition, *To appear in* Proc. ICANN-91, Espoo, June 24-28, 1991.

[6] T. Kohonen, Self-Organization and Associative Memory, Springer-Verlag, Berlin, Heidelberg, 1989.

[7] J. Lampinen and E. Oja, Distortion Tolerant Feature Extraction with Gabor Functions and Topological Coding, Proc. INNC-90, Vol. 1., Paris, July 9-13, 1990, pp. 301-304.

[8] S.P. Luttrell, Self-Organisation: a derivation from first principles of a class of learning algorithms, Proc. IEEE IJCNN-89 Washington DC, Vol.2. June 18-22, 1989, pp.495-498.

[9] E. Oja, Subspace Methods of Pattern Recognition, Research Studies Press Ltd., 1983.

[10] E. Oja, Neural Networks, Principal Components and Subspaces, Int. Jou. of Neural Systems, Vol. 1., No. 1., 1989, pp 61-68.

[11] E. Reuhkala, Recognition of strings of discrete symbols with special application to isolated word recognition, Acta Polyt. Scand. Ma 38. Dr. Tech . dissertation, Helsinki Univ. of Tech. 1983.

[12] H. Ritter and K.Schulten, Kohonen's Self-Organizing Maps: Exploring their Computational Capabilities, Proc. IEEE IJCNN-88 SanDiego, Vol.1. July 24-27, 1988, pp.109-116.