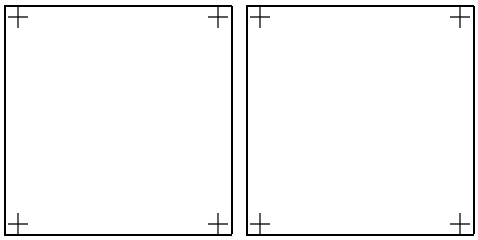
insensitive to small differences in object size. Figure J2 is a half octave smaller than J1 which is the largest possible difference, since a smaller image would appear as less than half octave larger in the next finer resolution, making the system capable of recognizing objects of any size.

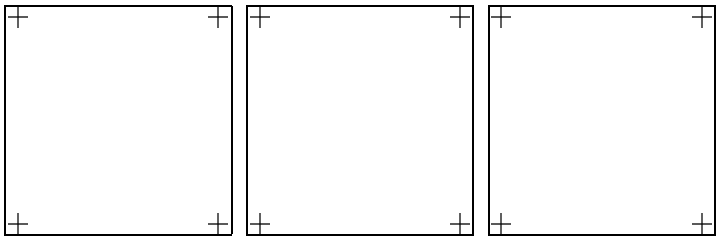
5. References

1. J. Buhman, J. Lange and C. von der Malsburg, "Distortion Invariant Object Recognition by Matching Hierarchically Labeled Graphs", Proc. IEEE IJCNN-89 Vol. 1. Washington DC, June 18-22, 1989, pp. 155-159.
2. J. Canny, "A Computational Approach to Edge Detection", IEEE Trans. on PAMI, Vol. 8, No. 6, Nov. 1986, pp. 679-698.
3. J. Daugman, "Two-dimensional Spectral Analysis of Cortical Receptive Field Properties", Vis. Res., Vol. 20, 1980, pp. 847-856.
4. J. Daugman, "Complete Discrete 2-D Gabor Transforms by Neural Networks for Image Analysis and Compression", IEEE Trans. on ASSP, Vol. 36, No. 7, Jul 1988, pp. 1169-1179.
5. V.S. Dotsenko, "Neural Networks: Translation-, Rotation- and Scale-invariant Pattern Recognition", J. Phys, A 21, 1988, pp. L813-L818.
6. 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.
7. D. Gabor, "Theory of Communication", J. Inst. Electr. Eng. 93, 1946, pp. 429-457.
8. J. Jones and L. Palmer, "An Evaluation of the Two-dimensional Gabor Filter Model of Simple Receptive Fields in Cat Striate Cortex", J. Neurophysiol., Vol. 58, 1897, pp. 1233-1258.
9. T. Kohonen, "Self-Organization and Associative Memory", Springer-Verlag, Berlin, Heidelberg, 1984.
10. S.P. Luttrell, "Hierarchical Vector Quantisation", Proc. of the IEEE, Vol. 136, Pt. I, No. 6, Dec 1989. pp. 405-413.
11. A.K.C. Wong, "Recognition and Shape Synthesis of 3-D Objects Based on Attributed Hypergraphs", IEEE Trans. on PAMI, Vol. 11, No. 3, Mar 1989, pp. 279-290.



a) The test image b) Edges

Fig.1. Edge detecting

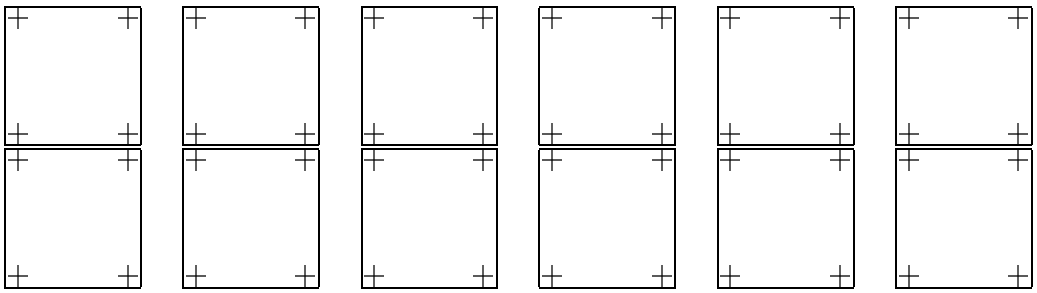


a) The test image b) Response $k = \pi$ c) Response $k = \frac{\pi}{2}$

Fig.2. Texture segmentation

mentation. The topological coding clusters the coefficients together so that texture detectors are automatically formed in the map. Since textures often have rather limited spectral distribution the feature clustering stage was carried out for each resolution separately ($M = 1$) with a 1-D map of 8 units and the codes are shown as grey values in the Fig.2.b-c.

For object recognition one can use Gabor Jets and labeled graphs directly from the coded pyramid with somewhat decreased accuracy and greatly increased speed due to shorter vectors being compared. To demonstrate the robustness effects of Gabor filters and topological coding we set up a following classification test: the test image representing several faces, each 64×64 pixels, was transformed and coded and two resolutions were combined with a 1-D 200-unit map. From resolution $k = \{\pi/2, \pi/4\}$ we took a 32×32 template for each face and matched it over the whole image with similarity measure as sum of squared difference of the codes. Fig. 3. shows the templates in the lower row and corresponding subimages in the upper row. Table 1. shows the the minimum differences for each face and for each template in units of 10^6.



T1 T2 J1 J2 K1 K2

Figure 3. Coded templates and corresponding images

	Face					
Template	T1	T2	J1	J2	K1	K2
T1	0	1.4	2.2	2.5	2.5	2.1
T2	1.4	0	3.1	2.6	2.2	2.3
J1	2.2	3.1	0	1.3	1.9	1.8
J2	2.5	2.6	1.2	0	2.0	2.1
K1	2.5	2.2	1.9	2.0	0	1.5
K2	2.1	2.8	1.8	2.1	1.5	0

Table 1. Minimum difference of templates of Fig.3. and test images.

There were two images of each person. Table 3. shows that either of the templates can be used for correct classification, with largest ratio of differences in the same class and the nearest other class being 1.5/1.8 (for template K2, K1/J1). Note that due to the detail expanding the coding is

some selected locations [6], to the Gabor mini-block representation and to use the mini-blocks for recognition. This is probably suitable for recognizing isolated objects, but for scene analysis, with different size objects in various backgrounds, it would be reasonable to store more coefficients at finer resolutions.

The approach we propose is to code the coefficient vectors in the Gabor pyramid so that the similarity relations are preserved, making code matching correspond to coefficient matching. In coding some information will inevitably be lost, but this is in fact desirable, if the coding will map clusters of input vectors to a given code. Small distortions introduce small changes in the Gabor coefficients, which is the very reason for their use in recognition, and such distortions form a cluster that will map to the same feature in the coding.

A coding scheme possessing all the outlined properties is the Kohonen topological feature map [9]. In [10] its is shown that the ordering of units in the map minimizes the average difference of input vectors that are mapped to a neighborhood used in training the map. This will optimally preserve the topology, or the structure, of the input probability density.

The 2-dimensional map is trained with 8-vectors obtained from Gabor-transformed arbitrary real world images. The a priori distribution of Gabor coefficients is the same at every resolution level, because the image appearing at each resolution depends on object sizes and distances that can be arbitrary. Thus the same map is used to code the whole Gabor pyramid, with the training samples drawn in random from all the resolutions.

2.3. Feature Clustering

The microfeature clustering stage is a topological map whose input is a vector consisting of codewords of, say, 3 consecutive resolutions and output is a new code for each pixel. If M resolutions are combined , the image representation is then a $F - M + 1$-multiresolution pyramid with each element representing a multiresolution feature.

The main advantages gained by the described feature extraction are

1. The coding means clustering and quantization of the Gabor-responses, mapping similar responses to the same code. Effects of small distortions disappear while the coding still maintains metrics properties of the responses.
2. Gaussian window makes the response of the Gabor filter to change smoothly over the image. If there is a clear detail with a specific unit in the coding map, the response of the unit will spread to some area around the detail in the coded image. That detail is "seen" not in one pixel but in some area in the image. This "detail expanding" helps matching prominent features in recognition and classification phase.

3. Applications

Three possible applications for the described feature extractor are edge detection, texture analysis and object recognition. In all the experiments the same 8×8 map was used in the coding stage.

Many edge detection algorithms use filters that resemble the Gabor functions. In [2] filters are derivatives of Gaussians in different orientations and the maximally responding filter determines the strength of edge detected. Then outputs of two different resolution filterings are combined to increase tolerance to edge sharpness. Although not mathematically equivalent, the Gabor filters give qualitatively similar responses, and the topological map forms edge detectors automatically. Fig.1.a shows the test image that was Gabor transformed and coded. Then the edge detecting units were selected manually and the image was classified to edge and nonedge pixels at resolution $k = \pi/2$. Fig.1.b shows the result. There were 8 edge detectors out of 64 units.

In [4] it is shown, that distributions of the 2-D Gabor coefficients can be used for texture seg-

real time. Any neural (or classical) classification method may be used in the microfeature space, instead of having to use optimization methods for graph matching. This is noteworthy because the elementary function of the majority of neural nets is signal transformation that can directly be used for classification. Optimization, on the other hand, is computationally expensive and difficult to implement in parallel hardware, and turning object recognition problem from classification to optimization will waste a large amount of knowledge achieved in the domain of pattern recognition.

The robustness of the microfeature space is demonstrated by showing that even a very simple template matching becomes rather tolerant to distortions.

2. Feature Extractor Overview

The system consists of 3 stages, or neural network layers, with one fixed and two adaptive layers.

1. Gabor transformation
2. Topological coding
3. Feature clustering

The input to the system is a grey level image. At the first stage the image is presented at various resolutions and Gabor-transformed to yield a Gabor pyramid of coefficient vectors. At each resolution one pixel is now replaced by a vector of Gabor coefficients. The coding stage then clusters together similar response vectors and compresses the amount of data approximately to the original level. To preserve similarity metrics of the coefficient space we use Kohonen's topological map [9] for coding. The final stage of the feature extractor is another topological map combining a number of frequency bands to a set of microfeatures covering several resolutions.

2.1 Producing the Gabor Pyramid

For both biological motivation and computational efficiency we use the self-similar family of Gabor functions [4]

$$\psi(x,y) = \exp\left(i(k_x x + k_y y)\right) \cdot \exp\left(-\frac{(k_x^2 + k_y^2)(x^2 + y^2)}{2\sigma^2}\right)$$

where

$$k_x = k\cos\theta, \quad k_y = k\sin\theta \quad \text{and} \quad \begin{cases} k = \frac{2\pi}{2^m}, & m = 1,2,3,\ldots \\ \theta = \frac{\pi n}{T}, & n \in \{0,\ldots,T-1\} \end{cases}$$

Parameter m enumerates frequencies in one octave intervals and n enumerates the T orientations. Parameter σ determines the width of the Gaussian window of the function, and the frequency band the filter covers. We used $\sigma = 2\pi$ which gives overlapping frequency responses but better localized features in spatial domain.

The discrete Gabor transformation of image I is defined by

$$G(x_0, y_0) = \sum_{x,y} I(x,y) \cdot \psi(x_0 - x, y_0 - y)$$

To Gabor transform the image at F resolutions, we use only one set of kernels at $T = 8$ different orientations and use self-similarity by resampling the image at lower resolutions. The resulting data structure, the Gabor pyramid is a quad-tree representation of the image with each pixel replaced by an 8-vector of magnitudes of the transformation coefficients.

2.2. Topological Coding

The whole Gabor pyramid can be stored for each image but it requires a large amount of storage ($\sim 1\frac{1}{3} \times 8 \times N^2$ real numbers where N^2 is the number of pixels in the image, vs. N^2 bytes in the grey level representation). Another possibility is to store all coefficients, forming a Gabor Jet, at

DISTORTION TOLERANT FEATURE EXTRACTION WITH GABOR FUNCTIONS AND TOPOLOGICAL CODING

Lampinen J. and Oja E.

Lappeenranta University of Technology, Department of Information Technology
PO Box 20, SF-53851 Lappeenranta, Finland

Abstract

A distortion tolerant feature extractor based on a 3-layer neural network is proposed. The first layer is a Gabor transformation giving distortion insensitive multiresolution representation of the image. The second layer is a topological coding that clusters the Gabor coefficients preserving their similarity properties for recognition. The third layer is another topological coding combining data from several resolutions to a set of microfeatures. The system is a pure feedforward network making parallel implementation easy. Three possible applications are discussed and preliminary experiments in object recognition, texture segmentation and edge detection are presented.

1. Introduction

The real world object recognition problem is extremely difficult, because the system must be tolerant to various distortions and variations in the objects. Variations are due to differences in scale, orientation and position, different lighting conditions, various backgrounds and partial occlusions by other objects. If feature extraction pays no attention to distortion tolerance, then the classifier must form the distortion invariant class boundaries, and that would require all the possible distortions and variations in training data and very a complex classifier. This approach has shown to be inefficient.

In classical pattern recognition applications these problems are usually avoided by controlling imaging conditions and object placement. Distortion invariant object recognition has been proposed by extensive search through pattern space [5] or by coding the objects as labeled graphs and matching them to maximize some defined similarity measure [11]. Both these approaches are computationally very expensive and applicable to rather restricted problem domains.

The biological visual system incorporates all the desired invariance properties which makes it tempting to try to use same kind of feature extraction and recognition in artificial systems. Trying to replicate the visual system with artificial neural networks seems a quite hopeless effort not knowing the higher mental processes involved in the recognition task, but some special functions of the biological system may turn useful.

To date, heavy expectations are set upon using Gabor functions in feature extraction. The Gabor functions were originally invented as functions optimally localized in both time and frequency domain [7]. Independently, the 2-D Gabor functions are reported to fit well to the receptive fields of simple cells in mammalian visual cortex [3][8].

Gabor functions have been used for object recognition in e.g. [1] and [6] showing promising tolerance to various distortions. In both works the data structure used for recognition was a labeled graph with nodes as the locations where Gabor filters were applied and labels as the filters outputs. Graph matching was carried out by Monte-Carlo optimization.

In this work we present a multilayer neural network for feature extraction using Gabor filters. The output of the network is a multiresolution representation of local microfeatures in the image. As purely feedforward network, the system is readily implementable in parallel pipeline hardware in

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