Institut für Neuroinformatik

Ruhr-Universität Bochum

Internal Report 93–01

Neural Mechanisms of Elastic Pattern Matching

by

René Doursat, Wolfgang Konen, Martin Lades, Christoph von der Malsburg, Jan Vorbrüggen, Laurenz Wiskott, Rolf Würtz

Ruhr-Universität Bochum
Institut für Neuroinformatik
44780 Bochum

IR-INI 93–01
January 1993
ISSN 0943-2752

To be published in: Proceedings of a BMFT--Workshop
held at Maurach, Germany, 20--21 October, 1992

Neural Mechanisms of Elastic Pattern Matching[1]

R. Doursat, W. Konen, M. Lades, C. v.d. Malsburg, J. Vorbrüggen, L. Wiskott, R. Würtz
Institut für Neuroinformatik, Ruhr-Universität Bochum

1 Introduction

We here report on the state of development of a number of individual efforts within the scope of our project. They all are intended to contribute to the goal of developing a visual architecture as a basis for the representation of scenes and for the construction of such representations. In our view, key elements of a visual architecture are a data structure to represent aspects of objects and a general mechanism to match such aspects. With these elements in hand, more complex models of individual objects and of entire scenes can be constructed as dynamic linkages of individual aspects, and stored structures can be matched to and recognized in new images.

As our basic data structure we are using labeled graphs. Nodes correspond to localities in aspects (or, later, scenes), links connect nodes that represent the same or neighboring localities, and labels correspond to local features. In many of our applications involving gray-level images we are using wavelets as local features (see section 2). These correspond closely to the receptive field types found in visual cortex, see, e.g., [ValVal88].

As our basic elastic matching mechanism we are using the dynamic link matching (DLM), which is part of the Dynamic Link Architecture (DLA) [Mal81]. In this scheme, dynamic links, which are not part of classical neural architectures, are represented by temporal correlations between the signals of the linked nodes as well as by rapidly modified synaptic weights. Rapid weight modification is controlled by signal correlations in a quasi-Hebbian fashion: positive correlation leads to weight increase, negative correlation to weight reduction. On the other hand, signal correlations are generated by the arrangement of synaptic weights, closing a loop of interactions with positive feedback. This loop is the basis of a system of dynamic link self-organization which favors certain link structures or "connectivity patterns". Among the connectivity patterns are low-dimensional topological graphs, which are an ideal data structure for the representation of visual aspects and of scenes. Furthermore, given two identical or similar connectivity patterns and a system of connections between them, link dynamic converges on a sparse connectivity pattern between them that connects corresponding nodes with each other. This is the mechanism of DLM.

In its most primitive and most general version, DLM takes the form of a sequence of activity events. Each one of this consists in the firing of a relatively small number of neurons, neurons that are distinguished by being more strongly connected with each other than average. About half of the neurons are in each of the two networks to be matched, presumably in corresponding locations. As a result of the activity event, all links between the participating neurons are strengthened, at the expense of other links of the same neurons. A

[1]Verbundprojekt NAMOS (Neuronale Architekturprinzipien für selbstorganisierende mobile Systeme), Teilprojekt "Bildverarbeitung in dynamischen Neuronennetzen", Projektleiter Prof. C. v.d. Malsburg.

sequence of such activity events knits the two networks together by activating links between corresponding locations and deactivating links between non-corresponding locations.

This basic mechanism has the advantage of great generality, not being prejudiced as to magnification, orientation, distortion or greater or lesser completeness of the part-networks involved, but it has the disadvantage of being very costly in terms of physical connectivity (from which active links are to be selected) and in terms of matching time. We therefore believe that this mechanism plays a dominant role only in the early ontogenesis of the brain when it is used to set up specific connection structures which are the basis for more efficient, though less general, matching. In the context of our project, which is concerned with technical implementation, our game has been to model the general DLM mechanism by specific algorithms which capture its basic style while having the advantage of being better adapted to computer implementation and being more economic in processing time and required connectivity, but which have the disadvantage of requiring explicit algorithmic specification of the degrees of freedom of the desired matches (position, magnification, distortion, rotation, partial match, to name some). The individual projects described below correspondingly fall into two categories. In sections 2 to 5 real camera-derived images are analyzed, but a matching algorithm is used that is efficient and narrowly taylored to the specific goal at hand and that somewhat dissimulates its neural basis. Sections 6 to 8 describe efforts to reach towards a formulation of DLM that retains the basic neural style in terms of temporal signal correlation and synaptic modification and yet is sufficiently economic to be used in technical implementations.

Section 2 describes a system for object recognition that is invariant with respect to position and that is robust with respect to deformation, rotation in depth and illumination. In section 3, work performed by R. Würtz, we show the amount of detail retained in our model graphs, by reconstructing images from stored data. In section 4 we open the search space of the matching algorithm to in-plane rotations and to varying magnification, keeping matching time low by making up for additional search space by basing initial parameter estimates on small subsets of the data. In section 5 the matching algorithm is modified to be able to deal with partial matches and its power is demonstrated with cluttered scenes made up of arrays of partially occluded objects.

Section 6 is an advertisement for work that is intended as a contribution to the learning time problem, but that is relevant also to the mapping problem as generality is required and as it is based on the basic DLM. Section 7 describes ongoing work by W. Konen and T. Maurer in which they are developing a greatly speeded-up version of DLM, called FDLM (for fast DLM), without sacrificing much of generality, the key point being simply that activity events are not created by actual neural signal dynamic but in an algorithmic fashion (whereas link dynamic is retained in its original form). Section 8, finally, describes ongoing work by R. Doursat which aims at a biological theory for the capabilities of the human visual system when it comes to judge metric relationships within the frontal plain. The basic idea in this work is to use running waves as activity events, two parallel waves in the two planar networks to be matched. The speed is controlled by local in-plane connectivity and is regulated such that equivalent distance elements $\mathrm{d}x$ are covered in equal time (a large distance element in the foveal region of primary visual cortex being equivalent to a small one in a peripheral region).

2 Object Recognition Invariant to Position and Distortion

As a demonstration of the capabilities of the Dynamic Link Architecture, we developed a programme that can recognize objects — specifically, camera images of human faces — although different images of the objects vary strongly in aspect or, in our case, facial expression, hair style, and so on. On the other hand, the programme is not specialized to any type of object; it has been tested with, e.g., office implements as well. To achieve acceptable performance, the programme has been implemented on a network of up to 40 transputers.

2.1 Data Acquisition and Preprocessing

The first step captures a 512×512 pixel snapshot with a CCD camera at 8 bits (256 gray levels) of resolution. The image is then sampled down to 128×128 pixels. As motivated by receptive field properties, this image $I(\vec{x})$ is convolved with a bank of DC free, complex-valued Gabor-based wavelets $\psi_{\vec{k}}(\vec{x})$ (bandpass filters) parameterized by their spatial frequency $\vec{k}$. In our case 6 frequency levels and 8 orientations are sufficient according to power spectrum analysis for faces and biological data [JoPa87]. The Fourier transform of the filter functions are given by the following equation:

$$\left(\mathcal{F}\psi_{\vec{k}}\right)(\vec{\omega}) = \exp\left(-\frac{\sigma^2\left(\vec{\omega}-\vec{k}\right)^2}{2k^2}\right) - \exp\left(-\frac{\sigma^2\left(\vec{\omega}^2+\vec{k}^2\right)}{2k^2}\right)$$

where the second term makes $\psi_{\vec{k}}$ DC free. $\vec{k}$ is restricted to

$$\vec{k} = \begin{pmatrix} k\cos\phi \\ k\sin\phi \end{pmatrix};\ k = \pi(\sqrt{2})^{-\kappa}, \kappa = 2\ldots 7;\ \phi = \nu\frac{\pi}{8}, \nu = 0\ldots 7.$$

A vertex $\vec{x_0}$ is labeled with the vector of the absolute values of the responses $\mathcal{J}I(\vec{x_0})$ (called a jet):

$$(\mathcal{J}I)(\vec{k}, \vec{x_0}) := \left|\int \psi_{\vec{k}}(\vec{x_0}-\vec{x})\, I(\vec{x})\, d^2x\right|$$

This choice of vertex labels guarantees some robustness against local distortions.

To add a new face to the database, we extract and store the $\mathcal{J}I$ at the points of a 8×10 grid, reducing the precision to fit a byte for each filter response. This compresses the data by a factor of approximately 4 from the original 128×128 images.

Graph edges are labeled with the distance vector $\vec{\Delta}$ between their endpoints in order to ensure that similar vertex labels in noncorresponding regions of the image are not identified during matching.

2.2 Matching

The recognition step is a two stage matching process. In both stages two graphs are compared by the weighted sum of comparison functions of the two types of labels (E and V are the edge and vertex sets, respectively):

$$C_{total} := \lambda \sum_{(i,j)\in E} \mathcal{S}_e\left(\vec{\Delta}^I_{(i,j)}, \vec{\Delta}^O_{(i,j)}\right) - \sum_{i\in V} \mathcal{S}_v\left(J^I_i, J^O_i\right)$$

For $\mathcal{S}_v$ we chose the cosine of the angle between the jets to achieve some degree of intensity independence. For $\mathcal{S}_e$ we used the squared difference of the edge labels. This yields a cost function for a pair of image and object graphs which is to be minimized.

In a first step, the approximate location of the face in the image is determined by comparing it with one fixed object. For that purpose C_{total} is minimized varying only the center of the graph and leaving its geometrical shape unchanged. This estimate initializes the second step, where for every object graph in the database C_{total} is minimized varying all vertex positions in the image graph independently. After convergence we have a final cost value for each object; the smallest of them belongs to the recognized person if it is significantly different (according to some statistical criteria) from the other matches. Both optimizations are done by simulated annealing at zero temperature, for which the cost landscapes are sufficiently well behaved [LVB92].

It is also possible to start with less than the full number of frequency levels and vertices and then to increase both during the matching process.

2.3 Recognition Performance

In order to assess the performance of the programme, we collected three galleries of face images from a set of 88 persons. One gallery is used to generate the database; the subjects where asked to look straight into the camera for this. A second gallery was taken with subjects looking approximately 20 degrees to their right, while for the third gallery they were asked to modify their facial expression.

The comparison process described above yields a number for every pair of image and stored graph. We thus need a mechanism to decide whether the graph with the best match value is indeed the correct one, or whether the graph of this person is not included in the database. For this, we developed two statictical criteria, described in detail in [LVB92]. If the values for these criteria exceed a threshold, the graph with the best match value is deemed recognized; otherwise, the system effectively says "I'm not sure.".

The programme's performance is given in table 1. The thresholds of the criteria were adjusted such that for one gallery, no false positives resulted (columns 2 and 5). The system then identifies 88% and 84% of the images in a significant way (column 1), while at the same time it avoids wrong and significant recognitions (column 6) and false positives in the other gallery. In two (gallery 1) and three (gallery 2) cases, shown in column 4, the best match is not the correct graph; thus, in $\approx$97% of the cases, the system picks the correct graph from the database (sum of columns 1 and 3). (In the few exceptions, the persons rotated their head in the image plane; invariance to this is discussed in section 4.)

3 Reconstruction from Stored Models

The coding of known faces as elastic graphs that are vertex labeled by the local components of the wavelet transform of the grey level image has proven very successful for the recognition task (see section 2). For a better theoretical understanding of the algorithm as well as for further improvements it is necessary to know how much of the original image information is preserved in this data format.

A (continuous) wavelet transform

$$(\mathcal{W}I)\left(\vec{k},\vec{x}\right) := \left(\psi_{\vec{k}} * I\right)(\vec{x})$$

gallery	criterion	case 1	2	3	4	5	6
gal. 1	κ_1	86	100	11	2	0	0
	κ_2	83	100	15	2	0	0
	κ	88	100	10	2	0	0
gal. 2	κ_1	79	100	17	3	0	0
	κ_2	80	100	16	3	0	0
	κ	84	100	13	3	0	0

Table 1: Results of comparing two galleries (gal. 1, head rotation by 15°; gal. 2, grimaces) against the standard image database of the same persons. All entries are expressed as percentages. For details, see the text.

allows reconstruction of the image I by means of the inversion formula

$$I(\vec{x}) = g \cdot \int (\mathcal{W}I)\left(\vec{k}, \vec{x}\right) * \psi_{\vec{k}}(-\vec{x}) d^2k.$$

The factor g in front of the integral is finite if the DC component of the wavelets vanishes (otherwise the inversion formula is invalid). This is the case for our wavelets, and in the continuous case the reconstruction is trivial. We are particularly interested here whether our sparsely sampled image data still give a satisfactory image reconstruction.

The frequency space sampling for the face recognition system has been restricted to an area where the *difference* between faces is expressed, i.e., the low and high frequencies have been cut off. Therefore, only the reconstruction of a bandpass filtered version of the original image can be expected.

Figure 1: **a)** An image of a face with the corresponding graph superimposed. **b)** Reconstruction from the complete transform (128 × 128 jets). **c)** Reconstruction from the graph (8 × 10 jets).

Figure 1 shows the results of the reconstruction procedure. The following observations (which have been tested on several images) can be made:

- Although the frequency space is not completely covered an image can be reconstructed which is easily recognizable by a human.

- The sampling in image space (8×10) is dense enough for recognition purposes.
- The locality of the wavelet components leads to a suppression of undesired background structure outside the area covered by the graph.

The recognition system as described in section 2 does not make use of the phase of the (complex-valued) wavelet components. Therefore, several attempts have been made to reconstruct the image from the moduli only. All of them have failed — the phases do contain important image information. As a consequence, they should be incorporated in further versions. Research on a useful way of doing this is currently in progress.

4 Invariance with Respect to Size and Orientation

4.1 Algorithmic Description

Our face recognizer described in section 2 and [LVB92] already has the ability of distortion invariant recognition. Its ability to cope with local distortions by elastic matching also accounts for variations in global transformation parameters limited to roughly 5-10% without serious performance degradation. Examples for global transformation parameters are scale and rotation in a plane vertical to the viewing direction. This lack of accounting for global transformations except translation was one of the most obvious reasons for the failure of the algorithm on some pictures of our standard face image galleries. After checking the behaviour of the potential derived from the similarity functions, the most reasonable approach seemed to try an optimization procedure transforming the model graph globally in scale and rotation space. This optimization procedure for scale and rotation angle can be applied to the model graph in alteration with a translation optimization (global move) correcting the center of gravity position and local node optimization adapting to local distortions.

4.2 Scale Transformation

Scale transformation of the graph model consists of two parts, the transformation of the vertex and the edge labels. The edge labels, the distance vectors in the model graph are just scaled by the appropriate scale factor. Since the feature vectors are sampled on a logarithmic scale, the application of a scale factor just means a shift of vector coefficients over the frequency coordinate. Before comparing a model feature vector to one in the image it has to be shifted along the frequency axis by the global scale factor α. Two further modifications to the comparison are necessary. First the sparse sampling of the frequency coordinate necessitates the use of interpolation between vector coefficients if no direct comparison is possible. Second the feature comparison has to be windowed to the overlapping region of the feature vectors after the shift was applied in this example of an input smaller than the model:

$$\begin{aligned} & \mathcal{J}_i^M(\vec{k})\alpha \\ \rightarrow \mathcal{J}_i^M(\tfrac{\vec{k}}{\alpha}, \alpha\vec{x_0}) & \overset{rect(\alpha)}{\rightarrow} \mathcal{J}_i^M(\tfrac{\vec{k}}{\alpha}, \alpha\vec{x_0})_{windowed}. \end{aligned} \tag{1}$$

An application of the described scale estimation is shown in figure 4.2 with scaling by 75% and 50% (compare also [BLM90]).

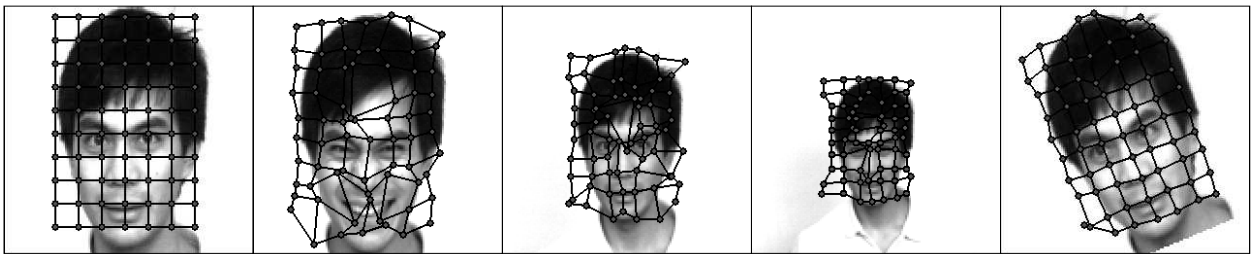

Figure 2: Recognition invariant with respect to distotion (2^{nd} panel), size (25% and 50%, 3^{rd} and 4^{th} panel) and orientation (last panel). The model is shown in the 1^{st} panel.

The procedure showed reasonable performance if the scale difference between model and incoming image allowed a sufficient feature overlap (equation 1).

4.3 Rotation

Adaptation to a global rotation of an object was done by rotating the graph edges around the center of gravity and transforming the vertex labels in a fashion similar to the feature shift for scaling along the frequency axis, only now along in the angular coordinate. The feature shift is done modulo 2π, interpolation between neighboring coefficients just as above. For a result see figure 4.2.

Estimation of global parameters can be sped up by finding a way to estimate the parameter only once for all models in the gallery and using the reduced information of a thinned graph. This global parameter estimation is one way to reduce complexity of the DLA related dynamics.

5 Analysis of Cluttered Scenes

We have extended the face recognition system in order to additionally handle occluded objects [WisMa92]. With slight modifications we can demonstrate highly successful recognition and localization of objects in gray level images of complex scenes in spite of extensive mutual occlusion. Objects are made known to the system in a semi-automatic fashion. The performance level demonstrated may be comparable to that of classical computer vision systems. However, our system is distinguished by utmost simplicity and flexibility, which are the hall-mark of neural systems.

To specify the state of the scene interpretation system completely it is necessary in addition to represent which regions of the image have been recognized by which model graphs and what the occlusion relations between the objects are. The system works under the tight restriction that models are to be mapped without any distortion to the image and we correspondingly can describe the relation of the model domain to the image domain with the help of a few binary variables that decide on the recognition status of a model and the visibility or occlusion of its individual nodes, plus a single position vector for the placement of the model graph in the image.

We have developed two scene interpretation algorithms. In the first, simple version, each model graph is matched separately to the image to decide if and where it fits and to what extent it is occluded. This algorithm has the advantage that there is no need for all objects

in the scene to be known to the system. The algorithm examines all graphs in the model domain. First, the graph is matched to the image. Then, all nodes under a certain threshold for $\mathcal{S}$ are marked as occluded. Since we assume that occlusion occurs for coherent regions, the algorithm then regularizes the occlusion decisions.

For the second algorithm to work, there must be models for all objects in the scene. Posing such a constraint has the advantage that the relative occlusion relations can be determined and used for more reliable interpretation of the scene. For two graphs A and B, this relation will be characterized by the "occlusion index" $\mathcal{Q}_{AB}$. When it is computed, the system may already have decided that a third object (or objects) are occluding parts of A or B so that only part of their graphs are visible in the image.

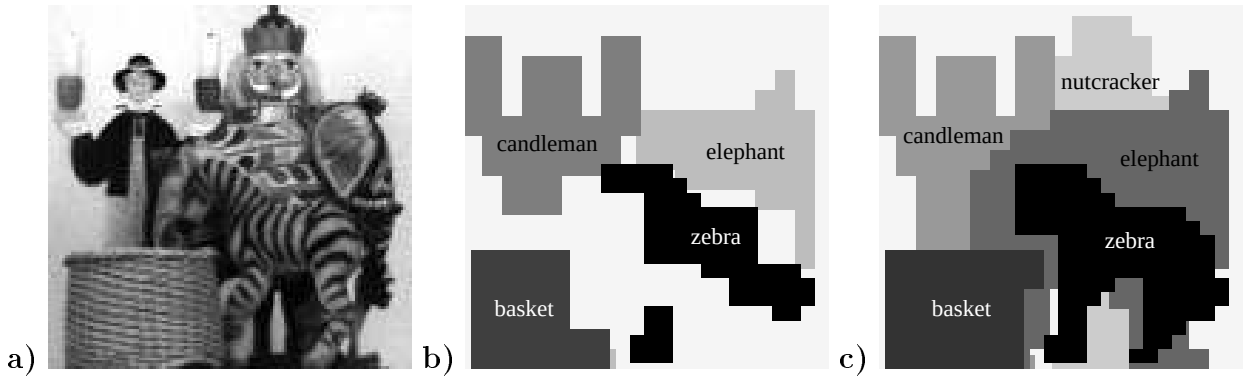

Figure 3: a) One of the 30 scene images. It contains the objects *zebra, basket, elephant, candleman*, and *nutcracker*. There are altogether 13 models in the gallery, and there are 121 objects in the scenes, with three to six objects in each scene. The resolution of the images is 128^2 pixels with 256 grey levels. b) Interpretation of this scene by algorithm 1. Visible regions of the matched model graphs are shown, from front (black) to back (light grey). The algorithm missed *nutcracker* in this scene, not finding the lower part under the zebra and discarding the identified head region as too small in area. For the zebra, large parts are interpreted as occluded, because of the perturbation of inner jets by overlap with the background. Altogether, 80% of the 121 objects were recognized correctly while 2 models were erroneously accepted. c) Interpretation of the scene by algorithm 2. All objects and their occlusion relations have been recognized correctly. Altogether, 96.7% of the 121 objects were recognized correctly, 3 models were erroneously accepted.

6 Learning from Single Examples to Recognize Symmetry

A large attraction of neural systems lies in their promise of replacing programming by learning. A problem with many current neural models is that with realistically large input patterns learning time explodes. For example, learning problems of higher order, like detecting an unknown symmetry within a pattern (Fig. 4), are difficult tasks for most neural networks. Even for small problem sizes a very large number of training examples is needed. This is a problem inherent in a notion of learning that is based entirely on statistical estimation: The patterns belonging to one symmetry class do not lie in clusters in the input space and

thus cannot be estimated from a small training set. This in turn leads to the explosion of learning time or of the number of prototypes required.

We propose a system which can classify symmetries after having encountered *one* example pattern of this symmetry class. The network is based on the Dynamic Link Architecture [Mal81], where rapidly modifiable links are used to express adaptive binding of features. It is based furthermore on the *a priori* restriction that significant symmetries are those wich preserve locally the topological structure of the input pattern. Where "learning" is required in other neural systems, we have here a stochastic self-organization process occuring within a single perception (i. e. pattern presentation). The aim of the unsupervised self-organization is to bring the actual perception into maximal consistency with the *a priori* knowledge.

In simulations the network achieved a classification reliability of 96% when trained on three different symmetry classes. An example of the self-organization process is shown in Fig. 4.

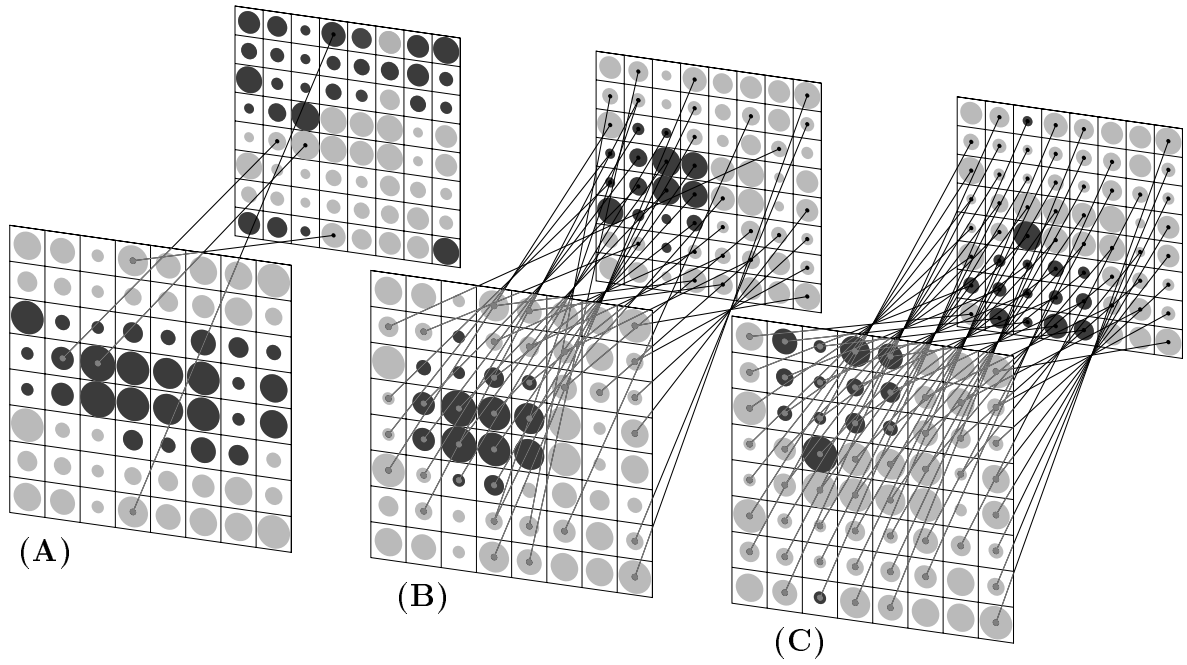

Figure 4: **Self-organized formation of dynamic links.** The figures **(A)**–**(C)** show different activation states generated from a single input pattern. The input pattern imposed to both layers has horizontal symmetry, its features represented here by the diameter of the different circles. The self-organization process consists of a sequence of activations, each of them activating large, overlapping regions in both layers (dark circles). The figure shows the network state after **(A)** 15, **(B)** 50, **(C)** 80 activations. Links grow between cells which are active simultaneously and have similar features.

7 Fast Dynamic Link Matching

7.1 Problem formulation

Consider the following task: Given are two patterns which consist each of $N \times N$ local features arranged in two 2D-layers X and Y. The problem is to decide whether the two patterns

can be matched by some invariance transformation and if so, to determine this invariance transformation. In the present case, invariances include translation, rotation, mirror symmetry along arbitrary axes and local distortions.

7.2 The neuronal system

Finding a match between patterns means finding a set of mutual corresponding cells $a \in X$ and $b \in Y$. [2] A cell a may be considered as neuron or neuronal group capable of two functions: *(i)* coding for a local feature f_a imposed by the actual pattern; *(ii)* representing an activity state x_a which can be transmitted to other cells. Correspondence can be expressed by binding the cells a and b through a dynamic link J_{ab}. Local features assigned to a and b may help as a guideline for candidate matches but will not solve the correspondence problem unambiguously, since a cell $b \in Y$ with feature f_b will have usually more than one candidate match $a \in X$ with the same feature. Let us define a similarity matrix T_{ba} which has high entries for all candidate matches, i.e., pairs of cells with similar features. In the present case we simply take

$$T_{ba} = \delta(f_b, f_a). \tag{2}$$

An initial – ambiguous – guess for the dynamic link network J_{ba} is $J_{ba} = T_{ba} / \sum_{a'} T_{ba'}$, i. e. each cell b is linked to all cells a with similar features such that the total link strength converging on each cell b is 1. A solution of the match problem must resolve these ambiguities such that each cell a has at most one correspondence b or – in other words – one dynamic link J_{ba} has a much higher value than all other links $J_{b'a}$ emerging from cell a. A neuronal activity mechanism capable of finding such a solution has been described earlier in [WilMa76, KoMa92b].

7.3 FDLM: Fast Dynamic Link Matching

The neuronal system based on nonlinear differential equations requires a number of parameters and extensive numerical calculations. We tried to find a simpler and faster equivalent algorithm with less parameters, the fast dynamic link matching (FDLM).
Before starting the algorithm one has to choose an unimodal blob function $B_0(b)$ with center at $b = 0$. This may be either an equilibrium solution of a neural field differential equation simply a window function. Let $\sigma(\cdot)$ be a sigmoidal function (which in the simplest case may be the unit step function). Here are the steps of the algorithm:

(i) Initialize the dynamic links with $J_{ba} = T_{ba} / \sum_{a'} T_{ba'}$.

(ii) Choose a random center $a_c \in X$ and place the blob there: $x_a = B_0(a - a_c)$. Compute the resulting input to layer Y

$$I_b^{(y)} = \sum_z J_{ba} T_{ba} \sigma(x_a). \tag{3}$$

(iii) Use $I_b^{(y)}$ to compute the position $b_c \in Y$ for which $\sum_{b'} B_0(b' - b_c) I_{b'}^{(y)}$ is maximal, and place the blob there: $y_b = B_0(b - b_c)$.

[2] We use lower-case symbols a, b, a', b' for index vectors $a = (a_1, a_2)$ specifying row a_1 and column a_2 of a cell within a 2D-layer.

(iv) Update the links between active cells such that the total link strength converging on each cell b is kept constant:

$$J_{ba} \rightarrow \frac{J_{ba} + \epsilon J_{ba} T_{ba} \sigma(y_b)\sigma(x_a)}{\sum_{a'}(J_{ba'} + \epsilon J_{ba'} T_{ba'} \sigma(y_b)\sigma(x_{a'}))}. \quad (4)$$

(v) Proceed with step (ii).

In its simplest form, where B_0 is only a window function and $\sigma(B_0) = \theta(B_0) = B_0$, the algorithm has only two free parameters: the size l of the blob and the update parameter ϵ.

A small number of blob activations ($\leq$ 20 for sheets of 8*8 neurons) suffices to establish an unambiguous mapping. With this formulation, establishment of a mapping can be accelerated in comparison to the original formulation in terms of differential equations by a factor larger than 10.

8 Metric Coding and Running Waves

The past few years have seen a renewed interest in investigating the fine temporal structure of neuronal activity. A modern trend in neurosciences is indeed to see temporal coding as a general format of representation used by the brain. In this frame, the new functional entities are *dynamic cell assemblies*, defined as groups of cells whose activity patterns engage in long-range synchronization during short periods. Originally put forward by von der Malsburg [Mal81] in the theoretical domain to tackle the "feature-binding problem", the idea of correlation coding gained evidence through recent experiments on cat visual cortex [GKE89]. These findings show synchronization phenomena between remote orientation columns during perception depending on global stimulus properties, and suggest *phase-locking of oscillatory activity* as a simple binding mechanism. They have also given rise to numerous models [KöSch91, MaBu92], ranging from detailed accounts of the experiments to more theoretical explorations. All these works share the hypothesis that perceptually distinct objects are labelled by the collective synchronization with *zero delay* of the units encoding their components. This amounts to figure-ground separation using tags of a temporal kind, viz. the phase of oscillatory responses assumed uniform over a coherent domain.

However, what proves appropriate to gross segmentation systems might not be sufficient for refined recognition tasks. The only relational information found in zero-phase locking is the mere fact that two units i and j "belong or not to the same set". All things considered, synchronization groups are akin to Hebbian assemblies on a faster time course. In these models the potential richness of temporal coding is far from being fully exploited, if we consider the possibility of more general correlation events such as: $< x_1(t)\ x_2(t-\tau_{12}) \ldots x_n(t-\tau_{1n}) >$, involving n neurons and taking also into account the natural *transmission delays* between their firing processes. On the other hand, topological relations between parts of an object or a scene are obviously an information of outmost importance, ignored by global synchronization. Four corners of a square do not just "belong" to the square but are also located in a specific manner with respect to each other.

Taking up these last remarks, we suggest here that *delayed correlations be the basis for metric coding*. To illustrate this conceptual viewpoint, we simulate a 2-D layer of oscillators locally coupled through connections including small delays: each element is thus inclined to

reproduce the behavior of its neighbors shifted with a short time-lag. The net result is the propagation of *waves* over the layer instead of uniform synchronizations. From the differential equations leading the dual excitatory/inhibitory dynamics of oscillators we also derive a single *phase-equation* [Kur84]. Such a formulation brings to the fore the crucial aspect of the network, viz. its temporal organization irrespective of the individual patterns of activity. Under this format, running waves are equivalent to *phase-gradients*, replacing the traditional phase-plateaus, and this is precisely what we assume to subserve the implementation of a *coordinate mapping*. In short, the effect of a plane wave is to mark out the layer with a coordinate axis along the direction of its propagation: neurons are labelled by the relative time of their activation. Therefore, two independent waves (or more) are needed to encode a 2-D metric system. Objects' topology is thus revealed by "multidirectional scanning".

Metric labelling is of particular interest when coming to shape recognition involving *graph-matching* operations. The organization of a topographic mapping between two layers is made much easier by the position information contained in the nodes (activity waves play a role analog to chemical gradients in developmental biology). Here, dynamical links undergo collective moves induced by global drifts of the phase-landscapes: through fast Hebbian plasticity, connections get their maximal strength where the nodes they connect have similar phases. Further formalization of the model leads to the "elastic-matching" algorithm developped in [BiDo89, BLM89]: phase interactions within a layer are equivalent to elastic forces and perturbations of the phase-landscape amount to deformations of the object.

In conclusion, compared to classical retinotopic models where neighborhood relationships are encoded through local, independent blobs of correlated activity, running waves install a *global* correlation order on the layer, which, on the other hand, is richer than uniform synchronization.

10.6

References

[BiDo89] E. Bienenstock and R. Doursat *Elastic Matching and Pattern Recognition in Neural Networks*. In: Neural Networks: From Models to Applications. L. Personnaz and G. Dreyfus eds., IDSET, Paris, 1989.

[BLM89] J. Buhmann, J. Lange and C.v.d. Malsburg *Distortion Invariant Object Recognition by Matching Hierarchically Labeled Graphs*. IJCNN International Conference on Neural Networks, Washington, Vol.I, 155-159, 1989.

[BLM90] J. Buhmann, M. Lades, C.v.d. Malsburg, *Size and Distortion Invariant Object Recognition by Hierachical Graph Matching*, Proceedings of the IJCNN International Joint Conference on Neural Networks, San Diego 1990, pp. II–411–416

[GKE89] C.M. Gray, P. König, A.K. Engel and W. Singer *Oscillatory responses in cat visual cortex exhibit inter-columnar synchronization which reflects global stimulus properties*. Nature (London) 338, 334-337, 1989.

[JoPa87] J.P. Jones, L.A. Palmer, *An Evaluation of the Two-Dimensional Gabor Filter Model of Simple Receptive Fields in Cat Striate Cortex*, Journal of Neurophysiology, 1987, pp. 1233–1258.

[KöSch91] P. König and T.B. Schillen *Stimulus-Dependent Assembly Formation of Oscillatory Responses: I. Synchronization.* Neural Computation 3, 155-166, 1991.

[KoMa92a] W. Konen, C.v.d. Malsburg, *Unsupervised symmetry detection: A network which learns from single examples*, accepted for the International Conference on Artificial Neural Networks 1992, Brighton, UK.

[KoMa92b] W. Konen, C.v.d. Malsburg, *Learning to generalize from single examples in the DLA*, submitted to Neural Computation.

[Kur84] Y. Kuramoto *Chemical Oscillations, Waves and Turbulence.* Springer, Berlin, 1984.

[LVB92] M. Lades, J.C. Vorbrüggen, J. Buhmann, J. Lange, C.v.d. Malsburg, R.P. Würtz, W. Konen, *Distortion Invariant Object Recognition in the Dynamic Link Architecture*, IEEE Trans. Comp., in print.

[Mal81] C.v.d. Malsburg, *The Correlation Theory of Brain Function*, Max-Planck-Institute for Biophysical Chemistry, P.O. Box 2841, D-3400 Göttingen, FRG. Internal Report 81-2, 1981.

[MaBu92] C.v.d. Malsburg, J Buhmann, *Sensory segmentation with coupled neural oscillators*, Biol. Cybern. 67, 233–242, 1992.

[ValVal88] R.L.de Valois, K.K.de Valois, *Spatial Vision*, Oxford University Press, 1988.

[WilMa76] D.J. Willshaw and C.v.d. Malsburg, *How patterned neural connections can be set up by self-organization*, Proc. R. Soc. London B, 194: 431-445, 1976.

[WisMa92] L. Wiskott, C.v.d. Malsburg, *A Neural System for the Recognition of Partially Occluded Objects in Cluttered Scenes*, submitted to Int. J. Pattern Recognition and Artificial Intelligence.