

INVARIANT FACE RECOGNITION IN A NETWORK OF CORTICAL COLUMNS

Philipp Wolfrum

Frankfurt Institute for Advanced Studies, JWG University, Ruth-Moufang-Str. 1, Frankfurt am Main, Germany
wolfrum@fias.uni-frankfurt.de

Jörg Lücke

Gatsby Unit, UCL, London, United Kingdom
lücke@gatsby.ucl.ac.uk

Christoph von der Malsburg

Frankfurt Institute for Advanced Studies, JWG University, Ruth-Moufang-Str. 1, Frankfurt am Main, Germany
malsburg@fias.uni-frankfurt.de

Keywords: face recognition, neural model, recurrent network, generative model, dynamic links, cortical column, self-organization.

Abstract: We describe a neural network for invariant object recognition. The network is generative in the sense that it explicitly represents both the recognized object and the extrinsic properties to which it is invariant (especially object position). The model is biologically plausible, being formulated as a neuronal system composed of cortical columns. At the same time it has competitive face recognition performance.

1 INTRODUCTION

An impressive capability of our visual system is invariant object recognition. The same object seen at different position, distance, or under rotation leads to entirely different retinal images which have to be perceived as the same object. In short, the system has to be invariant in terms of these transformations.

The traditional approach to achieving invariance in a neural system is the use of feature hierarchies. This idea was first expressed by Frank Rosenblatt in his four-layer perceptron (Rosenblatt, 1961) and a multitude of similar models has followed since (e.g. (Fukushima et al., 1983; Riesenhuber and Poggio, 2003)). Feature hierarchies consist of a number of stages that combine simpler features into more and more complicated ones while at the same time pooling over position, scale, etc. in order to achieve invariance to these transformations. This leads to a potential weakness of the concept, the inability to distinguish patterns that contain the same features in different arrangement, an ambiguity that is especially likely to occur in scenes with complex background. While this problem has been partially solved by newer models, one property of feature hierarchies remains: By pooling over variances, they do not only become invariant to them, but they effectively discard this information! In consequence, a system of this kind may be able to detect and recognize objects, but it has no

way of telling where the object is, what size it has, whether the person just recognized has a happy or a sad expression on her face, etc.

Our visual system is definitely able to perceive these extrinsic properties in addition to identifying an object. There is even widespread belief (e.g., (Xu, 1993)) that it has the ability to reconstruct the attended parts of a scene from an internal representation. Only such ability to accurately reconstruct gives ultimate assurance that all relevant aspects of the image have been understood and represented correctly. This is related to a distinction in computer vision, where *generative* models represent the joint probability distribution of input data and recognized object explicitly, while *discriminative* models only use the posterior probability of the object given input data (Ulusoy and Bishop, 2005).

We here propose a novel model of object recognition that follows this spirit of explicitly representing extrinsic properties and reconstructing the perceived object. Information about where the object of interest is located in the visual field is represented by *dynamic links* (a concept first introduced in (von der Malsburg, 1981)) that control information flow between the input domain and an invariant “Assembly” window in the model (see Fig. 1). At the same time, the Assembly receives top-down input from the model “Gallery”, and uses this information to try to reconstruct the perceived object. In this way, the whole

system implicitly reconstructs the full image of the perceived object, representing its intrinsic properties in the assembly window and the extrinsic properties (position, deformation) in the dynamic links.

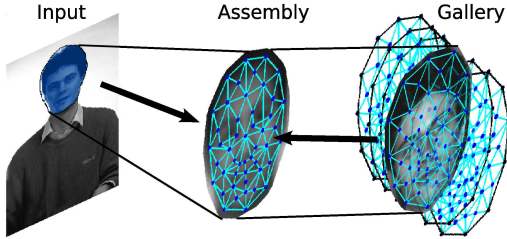


Figure 1: The principle of our reconstructive model for object recognition. See text for details.

We apply this mechanism here to face recognition as example of object recognition. Sect. 2 introduces the smallest computational elements that make up our model, which we then proceed to describe in detail in Sect. 3. We analyze performance of the system in Sect. 4 and close the paper with a discussion.

2 THE BASIC COMPUTATIONAL UNITS: CORTICAL COLUMNS

Our model is neurally implemented as a network of cortical columns. Today it is widely accepted that columns are basic computational units of the brain (Mountcastle, 1997). Columns in turn consist of *minicolumns*, which are bundles of ≈ 100 strongly interconnected cortical neurons.

All neurons of a minicolumnar network represent one single feature by their average firing rate x , which we call the unit's *activity*. It has been shown that the average spike rate of large excitatorily coupled neuron populations can be realistically described by continuous variables (van Vreeswijk and Sompolinsky, 1998) even on a fast timescale, and a specific minicolumn model has been proposed in (Lücke and von der Malsburg, 2004).

Several cortical minicolumns constitute a (macro)column (also called “hypercolumn” (Hubel and Wiesel, 1977) or “segregate” (Favorov and Diamond, 1990)), a group of minicolumns that together represent all relevant features at a certain point. The different features represented by a macrocolumn can compete through mutual inhibition of its constituting minicolumns (see below).

We describe column activity by a set of continuous differential equations called *modified evolution equa-*

tion. The activity of the i th minicolumn in a macrocolumn of K minicolumns is given by

$$\dot{x}_i = x_i^\nu I_i - x_i \sum_{j=1}^K I_j x_j. \quad (1)$$

The parameter ν introduces competition among the minicolumns forming a macrocolumn. For $\nu = 0$, there is no competition. All minicolumns represent their input proportionally, while the interaction term $\sum_{j=1}^K I_j x_j$ normalizes the steady state macrocolumn activity to a 2-norm of 1.

For $\nu = 1$, we have strong competition among the minicolumns, leading to winner-take-all (WTA) behavior. To see this, regroup Eq. 1 (with $\nu = 1$):

$$\dot{x}_i = x_i (I_i - \sum_{j=1}^K I_j x_j).$$

In this case the 1-norm of the column activity $\sum_j x_j$ is equal to 1 for the steady state, so the interaction term $\sum_{j=1}^K I_j x_j$ here is the average activity-weighted input to the macrocolumn. This means that only those minicolumn activities grow whose input is higher than this weighted mean input to the macrocolumn, otherwise they shrink. This lets the weighted input average grow, because the bias shifts towards strong inputs. Eventually, all minicolumn activities decrease to 0 except for the minicolumn with the strongest input, whose activity approaches 1.

In our model of object recognition we assume that there are two types of columns with different functions. Dynamically, they only differ in the use of the competition parameter ν :

- *Feature columns* represent their input in a linear fashion. Consequently, the minicolumns in a feature column have no need to compete among each other, i.e. for them the parameter $\nu = 0$.
- *Decision columns* show a WTA behavior leading towards a state where only the minicolumn getting the strongest input remains active. These columns receive a ν -signal that cyclically rises from 0 to 1. So they start out with linear dynamics like feature units. With rising ν , competition sets in, leading to an ever stronger WTA behavior that leaves only the minicolumn with the strongest input active.

In the networks that we will introduce in Sect. 3, minicolumns communicate with minicolumns of other macrocolumns. For this communication, a macrocolumn scales the output activities of its K minicolumns such that its output energy stays constant:

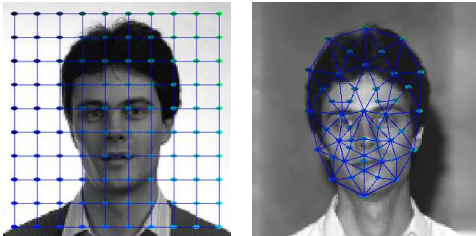
$$x_{i,\text{out}} := \frac{x_i}{\sqrt{\sum_{j=1}^K x_j^2}}. \quad (2)$$

3 THE MODEL

Our network is made up of *layers*, which loosely correspond to the different cortical areas that make up the ventral stream (the object recognition system of the brain). Layers are organized topologically, with a topology that may correspond to that of the retina or to a more abstract space. The layers of our network interact and activity collectively converges towards a final state that represents the “percept” of the network, in our case the possible recognition of a face.

Layers may contain both feature and decision columns. If we assume every feature column to represent all relevant features in one position of a retinal image, then layers of feature columns can represent whole images. The network introduced below uses layers of two different spatial arrangements:

- **Rectangular grid:** Straightforward representation suitable for any image. Every column represents one specific geometric location (see Fig. 2(a)).
- **Face graph structure:** An arrangement specifically suited for faces, where each macrocolumn represents an important landmark position on a face (see Fig. 2(b)). Note that in this case, a macrocolumn does not necessarily represent a fixed spatial location in the image, but rather a fixed semantic location (nose, mouth, eye, chin, etc.). Geometric locations of landmarks can change according to the face they represent.



(a) Rectangular grid (b) Face graph

Figure 2: Different representations of images.

The network consists of columns organized in the following layers (see Fig. 3):

- **Input Layer:** Represents the input image in a rectangular grid of $P = 20 \times 20$ points.
- **Assembly Layer:** Contains intermediate information from both the input image (represented in the *Input Assembly* macrocolumns) and the gallery (represented by the *Gallery Assembly* units).

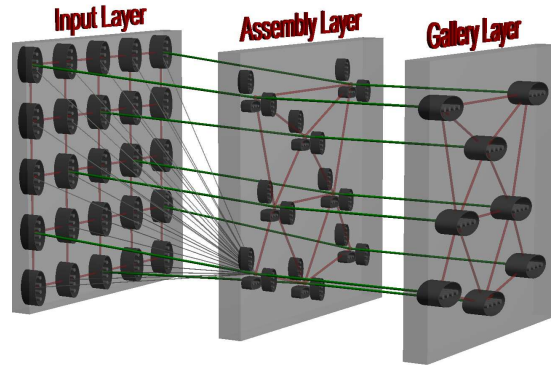


Figure 3: Architecture of our network. The gray oval structures represent macrocolumns, with minicolumns as lighter cylinders inside. The numbers of mini- and macrocolumns shown here are chosen exemplarily for visualization purposes only and are not identical to the real numbers of units used in this work. Input Layer is organized in a rectangular grid (represented by the red lines connecting macrocolumns), while both Assembly Layer and Gallery Layer have a face graph topology. Input and Assembly are connected all-to-all (shown exemplarily for the left-lowermost point in Assembly Layer), while Assembly landmarks are connected only to the same landmarks in Gallery, but to all identity minicolumns there. The green lines connecting the three layers represent a possible final state of the network.

- **Gallery Layer:** Represents all gallery face images in a face graph of macrocolumns.

The following three subsections describe these layers in detail.

3.1 Input Layer

The Input Layer represents the input image in a rectangular grid of $P = 20 \times 20$ points. At each grid point a macrocolumn represents $K = 40$ Gabor features (Daugman, 1980) extracted from the image at that position, namely wavelets of 8 orientations and 5 scales. The minicolumns $x_i^{I_p}$ (I_p being an index for the input layer columns) in this layer are feature units, i.e. they linearly represent the input they are getting from the image:

$$\dot{x}_i^{I_p} = \text{Gabor}_i^{I_p} - x_i^{I_p} \sum_{j=1}^K \text{Gabor}_j^{I_p} x_j^{I_p}, \quad (3)$$

for every Gabor feature i and every position p on the input grid.

3.2 Assembly Layer

The Assembly layer contains intermediate information from both the input image (represented in the *Input Assembly* macrocolumns and the gallery (represented by the *Gallery Assembly* units). This information is organized in an average face graph arrangement with $Q = 48$ landmarks (see Fig. 2(b)). The units of Input Assembly and Gallery Assembly are feature units. Mathematically, the input to an Input Assembly unit at position q of the face graph is given by

$$I_i^{I\mathcal{A}q} = \frac{1}{\sqrt{P}} \sum_{p=1}^P x_{\text{out}}^{C_{p,q}} x_{i,\text{out}}^{I_p}, \quad (4)$$

with $x_{\text{out}}^{C_{p,q}}$ the output strength of the dynamic link (see below) controlling the flow between Input column I_p and Input Assembly column $I\mathcal{A}q$.

The input to a Gallery Assembly unit at position q is the weighted superposition of all Gallery activities at the same landmark, filtered/multiplied by the feature vector represented by that respective landmark:

$$I_i^{G\mathcal{A}q} = \frac{1}{\sqrt{M}} \sum_{m=1}^M x_{m,\text{out}}^{G_q} w_{i,m,\text{eff}}^{G_q}, \quad (5)$$

with the ‘‘efferent weight’’ $w_{i,m,\text{eff}}^{G_q}$ representing the strength of Gabor feature i in landmark q of Gallery image m (of M in total).

The Assembly Layer contains the *control units* mentioned above, which mediate the signal coming in from Input Layer. These control units provide potential connections (*dynamic links*) between every Input Layer point to every point in the Input Assembly. The activity of the control units is driven by the feature similarity of the corresponding points in Input Layer and Gallery Assembly and in the end represents a position-invariant match between Input and Assembly Layer. Additionally, there is mutual support of control units that together would represent a geometrically consistent match between Input Layer and Input Assembly (see Sect. 3.2.1 for details.). The dynamic links are decision units, all links at one position of the Assembly Layer pointing to different Input Layer positions forming one decision column. The input to a dynamic link $x^{C_{p,q}}$ connecting input position p and assembly position q is given by the scalar product between both macrocolumn outputs and the topological influence from its neighbors:

$$I^{C_{p,q}} = \sum_{j=1}^K x_{j,\text{out}}^{I_p} x_{j,\text{out}}^{I\mathcal{A}q} + c_{\text{top},C} \sum_{\tilde{p},\tilde{q}} f_{\text{top}}(p,q,\tilde{p},\tilde{q}), \quad (6)$$

where $c_{\text{top},C}$ defines the strength of topological interaction (see Sect. 3.2.1).

3.2.1 Topological cooperation among control units

As mentioned before, there is topological cooperation among the control units of the Assembly Layer. The purpose of this cooperation is to establish a reasonable match between the different geometries of Input and Input Assembly. A given dynamic link connects a specific macrocolumn A of the Input Layer with another macrocolumn B of the Input Assembly. Due to the geometry of both layers, both macrocolumns represent distinct positions \mathbf{p}_A and \mathbf{p}_B in retinal coordinates and internal image representation space, respectively. Consequently, the dynamic link between them represents a certain geometric distance $\mathbf{d}_i = \mathbf{p}_B - \mathbf{p}_A$.

The idea is now to have topological connections in order to support parallel or near-parallel dynamic links. Therefore we define the strength of a topological connection between any two dynamic links i and j whose macrocolumns are neighbors in the face graph through a monotonically decreasing function of their non-parallelity/disparity:

$$c_{ij} = f(\|\mathbf{d}_j - \mathbf{d}_i\|_2) \quad (7)$$

Here we use a linearly decreasing thresholded function of the form

$$f(x) = \max(0, 1 - \beta x) \quad (8)$$

Thus topological interaction is always positive and acts only between more or less parallel (depending on β) neighboring links.

3.3 Gallery Layer

The Gallery Layer represents all M gallery face images in a face graph of macrocolumns. Each macrocolumn corresponds to one landmark, with the minicolumns representing specific feature vectors for the individual faces at the respective landmarks. The units in Input Assembly activate the Gallery minicolumns through receptive fields representing the stored facial landmark features, activating more strongly units of faces that are similar to the normalized input image in Input Assembly:

$$I_m^{G_q} = \sum_{i=1}^K w_{i,m,\text{aff}}^{G_q} x_{i,\text{out}}^{I\mathcal{A}q} + \frac{c_{\text{top},G}}{\|\mathfrak{N}(q)\|} \sum_{\tilde{q} \in \mathfrak{N}(q)} x_{m,\text{out}}^{G_{\tilde{q}}}, \quad (9)$$

with $c_{\text{top},G}$ defining how strongly Gallery units representing the same face in a neighborhood $\mathfrak{N}(q)$ around landmark q cooperate. Since Gallery columns are decision columns, this process leaves only the correctly recognized identity active in the end.

The Gallery projects a weighted superposition of its stored faces to Gallery Assembly. Point-to-point

comparison with the Input Assembly and competition among stored models leaves only the correctly recognized identity active in the end.

3.4 System Behavior

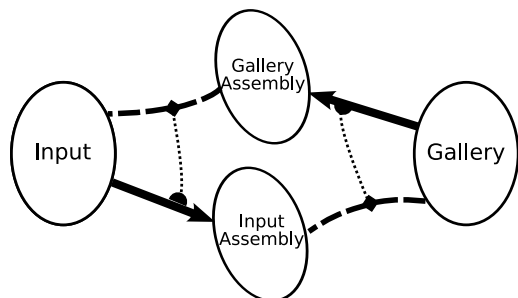


Figure 4: The principle of information processing in our system. Full arrows denote flow of information from one area to another. At the diamond symbols, information coming in via dashed lines is compared (scalar product). From there, a similarity signal flows out via the thin dotted lines, modulating point by point the information flow between areas.

We simulate the network dynamics using the simple Euler method, however, the time step is adapted dynamically to the average change of activity in the network in order to keep the system stable. All minicolumns have a small, but non-zero initial activity $x(0) = 0.01$. We drive the two decision layers with slightly different v dynamics: v_{Topology} rises from 0.25 to 0.9 during the cycle, while v_{Gallery} starts out from 0, also rising to 0.9. Consequently, competition and decision set in earlier in the control units than in the Gallery. The final value $v_{\text{max}} = 0.9$ makes the decisions of the network less sharp than a final value of 1.

The way the network processes information is sketched in Fig. 4. The units in the Input Layer, which receive input directly from the incoming image (not shown in Fig. 4), quickly develop to a state where they represent the input image via the different Gabor feature values at all grid positions. This information flows to the Input Assembly modulated by the activities of the control units connecting every point in Input Layer with every point in Input Assembly. The control units in turn are driven by the similarity of Input and Gallery Assembly at that pair of positions that they control.

The image information in Input Assembly in turn acts as input to the Gallery units, where it gets filtered through the individual receptive fields of the minicolumns, exciting those units more that represent faces more similar to the input image. A superposi-

tion of all stored faces, weighted by the current activity of the Gallery units, then flows to Gallery Assembly. This is equivalent to having a non-weighted flow of all stored faces from Gallery to Gallery Assembly, but modulated by the similarity of the representation in Input Assembly and each respective Gallery face (see 4).

The time course of the recognition process is visualized in Figs. 5 and 6. Since initially all control units have equal activity, this leads to a superposition of image information from all Input points at each Input Assembly location, resulting in a feature-less, more or less homogeneous mix of visual information in Input Assembly (first image in Fig. 5). In Gallery Layer, all faces are equally active initially. Gallery Assembly, which receives equal input from all Gallery units, will therefore initially receive a superposition of all Gallery faces, resembling an “average face” (like the first image in Fig. 6). The control units are driven by the similarity of the information stored in their dedicated feature units. Therefore control units that connect points of the average face with similar Input points will become stronger, while control units representing irrelevant matches will be weakened. Since the information flow from Input Layer to Input Assembly is modulated by the control units, the image in Input Assembly starts to develop from a gray non-descript superposition to a more and more clear version of the input image. It may be shifted and possibly distorted such that it conforms to the topology of the face graph of Gallery Assembly. This development is shown in Fig. 5. Due to competition between the minicolumns of each Gallery macrocolumn and cooperation among minicolumns of different landmarks representing the same face the Gallery will start to favor some of the stored faces over others. This in turn changes the image in Gallery Assembly from an average face to a superposition that is biased towards one or several of the better fitting gallery faces (second and third image of Fig. 6). This sharpened target face now helps at positioning the normalized input image even more precisely. In the final state, the Input Assembly will contain a shifted and maybe distorted version of the input image, while in Gallery the minicolumns of only one face are still active, and Gallery Assembly contains a copy of that face of the Gallery that the system judges to be most similar to the input image.

4 TEST RESULTS

We tested our system on the FERET (Phillips et al., 1998) and the AR (Martinez and Benavente,

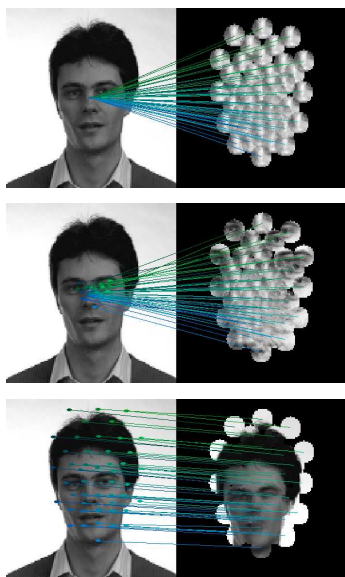


Figure 5: The process of finding the correct mapping between Input Layer and Input Assembly. The three images show the nonchanging input image on the left side, and an image reconstructed from the activities of the 48 landmarks of the Input Assembly on the right side. This initially (top image) contains a superposition of all input information, resulting in a gray more or less uniform image. This image then differentiates towards a normalized (i.e. shifted and deformed if necessary) version of the input image. This process is driven by the control units, which are represented here by lines connecting the input image with the Input Assembly. Each line represents the “center of mass” of a control macrocolumn, i.e. the location in the input image where its minicolumns are pointing to as a group, weighted by their activity.



Figure 6: Time course of the image represented in Gallery Assembly (from left to right). The Gallery Assembly gets input from all Gallery units and thus contains an activity weighted average of all faces in the gallery. Initially, when all Gallery units are equally active, this weighted average is a real average of all gallery faces, i.e. a mean face like in the leftmost image. With ongoing dynamics and rising competition, it gets biased towards the better fitting gallery faces, and finally contains only a representation of the image the system has recognized.

1998) databases, two very popular databases for the evaluation of face recognition systems. We followed the testing protocols of (Phillips et al., 2000) (the official FERET evaluation) and of (Tan et al., 2005). The FERET database contains images of 1196 individuals, while the subsets of the AR database used in (Tan et al., 2005) and by us contain 100 faces. In order to test the performance of a face recognition system, it is confronted with a gallery of images of all faces in the database, and is then asked to identify a different set of images containing pictures of (possibly a subset of) the faces in the gallery photographed under different conditions. Often not only the best match chosen by the system is recorded, but also the follow-up matches. This allows to construct *cumulative match scores*, the match score of rank n representing the fraction of test images whose correct match appears among the n best matches found by the system.

From the FERET database we used the following testing subsets: The set `fafb` contains photographs of 1195 individuals taken on the same day as the gallery images, but with the subjects showing a different facial expression. The set `Duplicate I` contains 722 of images that were taken at least one day but less than 18 months after the gallery images. Finally, the set `Duplicate II` contains 234 images taken more than 18 months after the gallery images. The cumulative match scores of our system for this database are shown in Fig. 7.

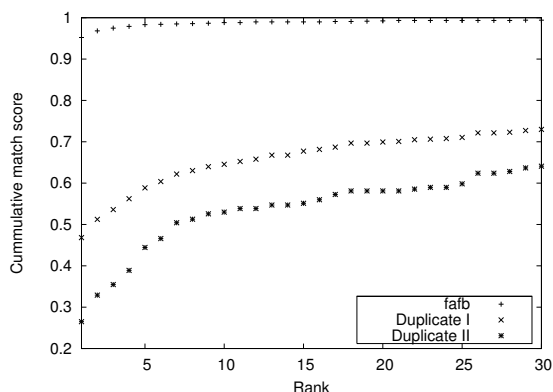


Figure 7: Performance of our system on the FERET database.

The AR Face Database contains several testing subsets with images of the same 100 subjects that make up the gallery. The subsets b, c, and d contain images of the subjects smiling, expression anger, or screaming, respectively. While those subsets were taken on the same day as the gallery images, the subsets h, i, and j show the subjects at a later session expression those same three emotions. Subsets e and

Table 1: Rank 1 match scores (in %) of our system, compared to those reported in the literature. The middle column shows the scores for the best performing system (Wiskott et al., 1997) of the official FERET evaluation, and in brackets the average score of all 13 systems evaluated. The next column shows the performance of the two systems (SOM-Face/LocPb) proposed in (Tan et al., 2005). The probe sets from the FERET database are the same as those of Fig. 7, while for the AR database, the three emotion sets (b,c,d and h,i,j, respectively) and the two types of occlusion (e,f and k,l) have been averaged.

		our system	(Phillips et al., 2000)	(Tan et al., 2005)
FERET	fafb	95	95 (85)	92/-
	duplicate I	47	59 (40)	
	duplicate II	26	52 (22)	
AR	Emotion	91		95/82
	Em. duplicate	61		81/82
	Occlusion	73		96/81
	Occ. duplicate	36		56/51

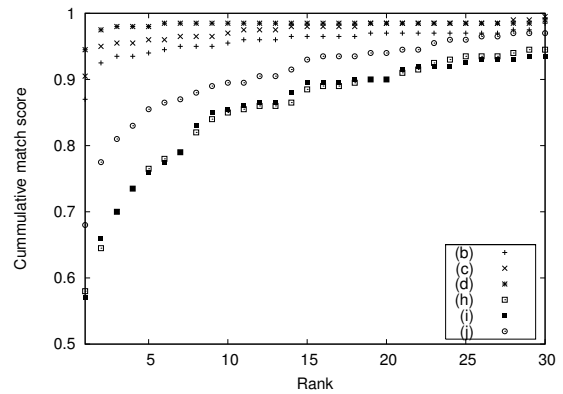
f show subjects wearing sunglasses and scarfs, and subsets k and l show the same situation at a later date. Cumulative match scores for this database are shown in Fig. 8.

Table 1 shows the performance of our system considering only the best match (i.e. rank 1), and compares it to the recognition rates of the systems evaluated in (Phillips et al., 2000), and to the performance of (Tan et al., 2005). We can see that our system outperforms the average of the systems tested in the FERET evaluation, but does not reach the performance of the winner of this evaluation. Similarly, performance on the AR database is poorer than that of the better approach proposed in (Tan et al., 2005).

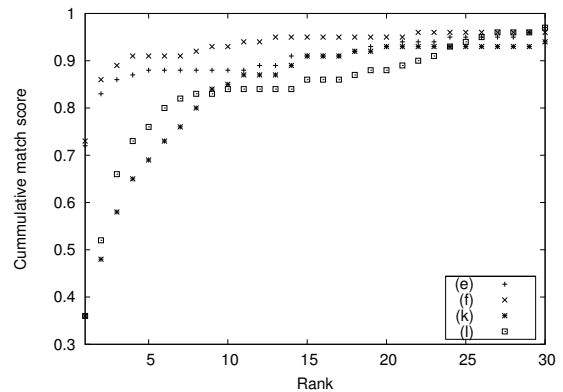
We can conclude that while our system is definitely competitive, its performance does not reach that of current top-notch face recognition systems. However, we did not apply any parameter tuning to the system as tested here. For example, it turns out that performance of the model grows monotonously with input grid size, with our resolution of 20x20 points still being far from saturation. In fact, the winner of the FERET evaluation (Wiskott et al., 1997) uses Gabor wavelets from every pixel of the input image! Other parameters that could be adjusted include the relative contribution of different landmarks, or the strength of topological interaction among the control units (Sect. 3.2.1).

5 DISCUSSION

Our main motivation behind this work was to create a fully neural and biologically plausible model, which is why we did not pay more attention to parameter tuning. The nature of our approach of explicitly using neural elements to route information enables a system that performs object detection *and* recognition in



(a) Results for the emotion datasets.



(b) Results for the occlusion datasets.

Figure 8: Performance of our system on the AR database.

one single network. This is in contrast to most other neurally inspired object recognition approaches, (including the approaches in (Wiskott et al., 1997) and (Tan et al., 2005)), which usually have to fall back on some algorithmic shortcuts to close the gap between the detection/segmentation and the recognition sub-systems.

The price we have to pay for such a system is its enormous computational cost. The strong dependence of this cost on input grid size is the reason why we constrained ourselves to a sub-optimal resolution of 20x20 points. However, we are working on solving this problem by spreading the routing from input to assembly over several layers (Wolfrum and von der Malsburg, 2007). Also, our current system is only invariant to translation and slight deformation of images. Work is under way to also address other transformations like rotation and scaling.

On the other hand, a fully neural system with most parameters represented locally naturally allows for adaptation through learning. Therefore we refrained from hand-tuning such local parameters here, but will address this issue through learning in future work.

In that sense, the system presented here marks but a starting point from which we could develop a fully neural version of a generative vision model that does not throw away variance information, but retains and uses it for recognition through active information routing. We are convinced that also for technical systems this approach to vision can serve as an inspiration. The impending transition to massively parallel processor arrays will revive interest in data flow architectures, in which data arrive just in time over dedicated pathways on processing nodes. Studying how these mechanisms work in the brain may turn out fruitful for designing robust and autonomous parallel computing systems.

ACKNOWLEDGMENTS

We thank Urs Bergmann for help in programming, and Alexander Heinrichs for helping to preprocess the database images. This work was supported by the European Union through project FP6-2005-015803 (“Daisy”) and by the Hertie Foundation.

REFERENCES

Daugman, J. G. (1980). Two-dimensional spectral analysis of cortical receptive field profiles. *Vision Res*, 20:847–856.

Favorov, O. V. and Diamond, M. (1990). Demonstration of

discrete place-defined columns, segregates, in cat SI. *Journal of Comparative Neurology*, 298:97 – 112.

Fukushima, K., Miyake, S., and Ito, T. (1983). Neocognitron: A neural network model for a mechanism of visual pattern recognition. *IEEE Transactions on Systems, Man and Cybernetics*, 13(5):826–834.

Hubel, D. H. and Wiesel, T. N. (1977). Functional architecture of macaque visual cortex. *Proceedings of the Royal Society of London - B*, 198:1 – 59.

Lücke, J. and von der Malsburg, C. (2004). Rapid processing and unsupervised learning in a model of the cortical macrocolumn. *Neural Computation*, 16:501 – 533.

Martinez, A. and Benavente, R. (1998). The AR face database. Technical Report 24, CVC.

Mountcastle, V. B. (1997). The columnar organization of the neocortex. *Brain*, 120:701–722.

Phillips, P., Moon, H., Rizvi, S., and Rauss, P. (2000). The FERET evaluation methodology for face-recognition algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(10):1090–1104.

Phillips, P. J., Wechsler, H., Huang, J., and Rauss, P. J. (1998). The FERET database and evaluation procedure for face recognition algorithms. *Image and Vision Computing*, 16(5):295–306.

Riesenhuber, M. and Poggio, T. (2003). How visual cortex recognizes objects: The tale of the standard model.

Rosenblatt, F. (1961). *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*. Spartan Books, Washington, D.C.

Tan, X., Chen, S., Zhou, Z.-H., and Zhang, F. (2005). Recognizing partially occluded, expression variant faces from single training image per person with SOM and soft kNN ensemble. *IEEE Transactions on Neural Networks*, 16(4):875–886.

Ulusoy, I. and Bishop, C. M. (2005). Generative versus discriminative methods for object recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 258–265.

van Vreeswijk, C. and Sompolinsky, H. (1998). Chaotic balanced state in a model of cortical circuits. *Neural Computation*, 10:1321–1372.

von der Malsburg, C. (1981). The correlation theory of brain function. Internal report, 81-2, Max-Planck-Institut für Biophysikalische Chemie, Postfach 2841, 3400 Göttingen, FRG. Reprinted in E. Domany, J.L. van Hemmen, and K.Schulten, editors, *Models of Neural Networks II*, chapter 2, pages 95–119. Springer, Berlin, 1994.

Wiskott, L., Fellous, J.-M., Krüger, N., and von der Malsburg, C. (1997). Face recognition by elastic bunch graph matching. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(7):775–779.

Wolfrum, P. and von der Malsburg, C. (2007). What is the optimal architecture for visual information routing? *Neural Computation*, 19(12):3293–3309. in print.

Xu, L. (1993). Least MSE reconstruction: A principle for self-organizing nets. *Neural Networks*, 6:627–648.