Technical Report SFB360-TR-98/6

Perceptual Grouping in a Neural Model: Reproducing Human Texture Perception

Jörg Ontrup* and Helge Ritter

Neuroinformatics Group
University of Bielefeld

Abstract

Texture discrimination plays an important role in image segmentation tasks. We propose a model of visual texture perception which involves a biological motivated feature extraction mechanism and the application of a neuro-dynamical model, the Competitive Layer Model (CLM). After giving an introduction to the "Gestalt"-approach of perceptual grouping, we will describe the architecture of the fully connected recurrent CLM and propose an algorithm to efficiently simulate the dynamics of the system. Based upon Gabor filters we present a detailed description of how to gain a reliable representation of texture features from images. Further stages of processing include an early nonlinearity and the spatial pooling of the texture features. A single interaction function between these features which involves the two principles of similarity and proximity then allows the usage of the CLM for image segmentation. We show that our model of texture perception can be applied to a remarkable range of images known from psychophysical studies and that the results are in good consistence with human introspection.

1 Introduction

One of the intriguing characteristics of the human brain is its ability to organize the tremendous amount of information supplied by our senses in such a way that the world is not perceived as a chaotic stream of impressions, but as a well structured set of entities.

Let us consider the following example to clarify this statement: The image of an old English castle surrounded by a bunch of trees consists of thousands of different colours. Hundreds of greenish tones characterize grass patches and tree leaves, a vast amount of greyish colours might represent the walls of the castle. These walls are not necessarily visible in whole pieces, but might be occluded by trees standing in front of them. The sensory information collected by the retina of our eyes is nothing more but an unstructured multitude of data depicting frequency and intensity of light. Yet we are not observing an unstructured set of separated impressions, but a coherently structured representation of the scene: All elements sharing a common "idea" belong together, we're just experiencing a few single entities: the castle, the grass and some trees.

*Email: jontrup@techfak.uni-bielefeld.de

1.1 The Gestaltists

In the late 20's of this century the Gestalt psychologists around Wertheimer, Koffka, and Köhler proposed a general framework for understanding human perception. Their intention was to provide a unified description of psychological phenomena applicable to a wide range of domains, including other modalities such as auditory or tactile input, as well as episodic memory, motivation, and problem solving. Their approach centered around the idea, as Koffka [Kof62] states, that *the mind perceptually organizes the world such that internal representations are of minimal energy and acts within the world so as to further reduce this energy as much as possible*. The concept of "energy" was justified by the reference to thermodynamical principles, where complex systems like a setup of magnetic dipoles settle into configurations with minimal energy values. The Gestaltists therefore postulated a structural isomorphism between brain and behaviour: For each aspect of physiological neural dynamics there had to exist a corresponding aspect at the mental level. A new light is shed on this rather bold assumption of an "energy field" in reference to neurodynamical systems.

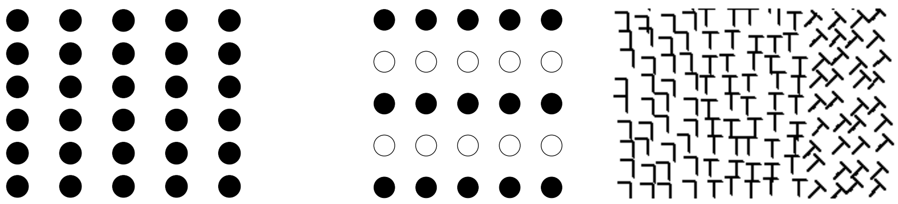
At the beginning of the Gestaltist's approach to perception stands the concept of an *environmental field*, which enfolds the total influence on the brain by sensory input. This can either be *atemporal* or *temporal*, which denotes an instantaneous "time slice" of perceptual input or a continuous sensation, respectively. The way the world is perceived is then described by the *law of Prägnanz*: the environmental field is mentally organized to maximize simplicity. The direction of greater simplicity is described by greater regularity and fewer experienced units, or *Gestalten*. Hence, the process of perception involves unit formation and simplification in perceptual organization.

1.1.1 The Gestalt Laws

To make this rather vague formulation more concrete, the Gestaltists tried to specify the form in which this minimization takes place within a general framework of laws. These laws were mainly based upon introspection, relating external stimuli to internal subjective sensations. The examples of grouping phenomena were primarily from the visual domain, although others present examples on auditory and somatosensory grouping as well [Rob97]. The most important Gestalt laws mainly found in the literature are stated below:

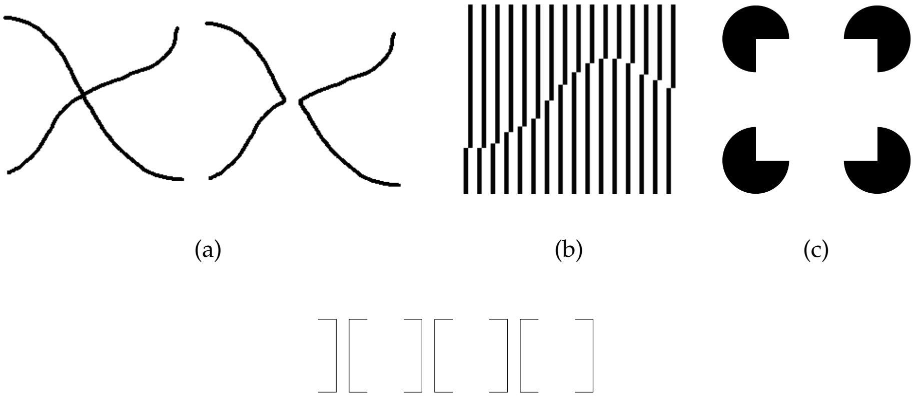
- **Law of Proximity:** Stimuli are grouped together into higher order units based on physical proximity.
- **Law of Similarity:** Similar features tend to form salient groups, where similarity can be applied to properties like brightness, colour, shape and orientation.
- **Law of Good Continuation:** Elements of a contour are grouped together when these elements establish an implied direction. Formation of a higher unit is strongly supported by line elements laying on a straight line. (See Figure 2(c))
- **Law of Closure:** Stimuli forming closed contours are grouped together.

The Gestaltists intendedly chose such simple examples as depicted in Figure 1 and 2, because this made it easier to separate the different mechanisms which might be involved in perceptual grouping. However, these laws are still not precisely formulated in computational terms. Given such general terms as "similarity" it is impossible to derive a quantitative theory. Therefore there also exists no general framework describing how these laws are interacting if more than one is applicable to a given input. See Figure 2(d)



(a) (b) (c)

Figure 1: Classical illustrations of visual grouping phenomena. Law of Proximity and Similarity: **(a) Proximity:** Perception of five vertical groups, **(b) Similarity:** Perception of five horizontal groups, **(c) Similarity:** The region with tilted T's segregates from the area with upright L's and T's (which seems to form a single unit at first sight)



(a) (b) (c)

(d)

Figure 2: (a) Good Continuation: Looking at the left figure one rather experiences two crossing, instead of two sharply bend lines, as suggested on the right hand side, **(b) Good Continuation:** Perception of an "illusory contour", the boundary elements are linked together, building a single line, **(c) Closure & Good Continuation:** The line elements represented by the "mouths" of the "pacmans" are laying on straight lines, they also form a closed figure, therefore a white square can be seen here, **(d) Closure:** Perception of three open squares

for an example: The figure can either be interpreted as a set of open squares or vertical I's, depending on whether the Law of Closure or Proximity is dominating.

1.2 Relevance of Perceptual Grouping to Computer Vision

Consider a robot vehicle, which should navigate autonomously in an unknown environment. The most natural way for humans to extract information from the surroundings is to "look and see what's there". However, these simple tasks to see *where* things are, and *what* these things are, represent the two major problems in computational vision. They are usually called *image segmentation* and *object recognition*, respectively.

The basic idea of conventional computational models is that visual perception involves a feed-forward progression of sensory information through various stages of feature processing. At each stage the information is getting more and more abstract and finally results in a symbolic representation of the scene. The beginning of this process is usually characterized by the extraction of basic features like edge positions, colours, lengths, orientations and contrasts. For a number of reasons, these features are not easy to interpretate (see

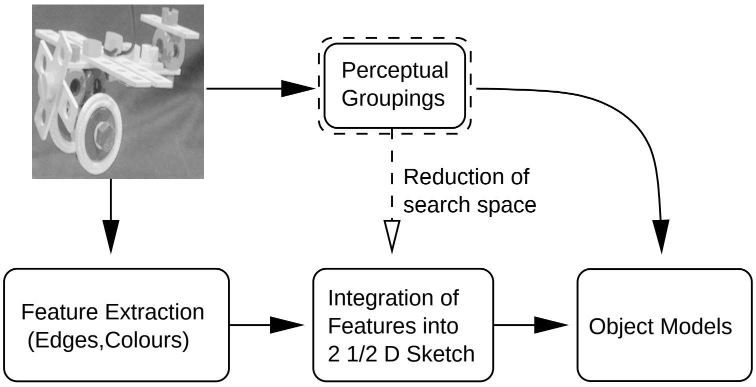


Figure 3: A model for visual recognition: The lower part shows the classical bottom-up approach. Perceptual grouping serves as a top-down control which reduces computational complexity.

[Zhe95] for a more detailed discussion):

Attributes extracted from natural images tend to be degenerated due to a variety of factors, such as uncontrolled lightning – which causes highlights and shadows – occlusions, and sampling effects. Therefore, one has to deal with unreliable and unstable features.

Furthermore, many features are just weak responses and of no significant meaning if interpretated in isolation. In other words, local features are perceptually ambiguous and contribute to the overall meaning of the scene only in combination with other – possibly weak – features.

Therefore, the main problem of the classical bottom-up approach is its computational complexity: Mohan [MN92] states, that for model matching, recognition is exponential if no prior knowledge about the feature relations is available.

This is where perceptual grouping plays a significant role: Humans can immediately detect relationships such as collinearity, connectivity and textual similarity. Unfortunately, the generality of the Gestalt laws makes it difficult to apply them to artificial computational systems, "where precision and definitness of procedural description are at a premium" [Rob97]. Nevertheless, the Gestalt principles give a good heuristic for perceptual analysis: Parts of the same object are likely to be close together (proximity) and have the same surface texture (similarity). Occlusion is likely to leave connectable segments on either side of the occlusion (good continuation) and distinct objects tend to have a close form (closure).

Robert [Rob97] notes, that the Gestalt principles are a possible secondary result of structure in the world: The relations of stimulus properties and the independently verified breakdowns of these stimuli into objects are statistically correlated and incorporated into the organization of the perceptual system. Therefore, the Gestalt laws provide a source of a priori knowledge, which can be used for the integration of a top-down control which serves as an input to a search-based recognition process. Figure 3 shows a possible integration of perceptual grouping into a classical model for visual recognition.

2 The Competitive Layer Model

As Robert [Rob97] notes, one of the main principles in perceptual grouping is that "equality [similarity] of stimuli produces forces of cohesion, inequality separation." This principle is mapped onto the architecture of the Competitive Layer Model which was first introduced by Ritter [Rit90] as a model for spatial feature linking. Similarity and disparity are expressed by an interaction function between characteristic features. This interaction function is then used to construct an energy function which measures the quality of a grouping state. In order to achieve a good perceptual organization the energy is then minimized by the dynamics of the recurrent CLM. See also [WSR97] for an analysis of the CLM's dynamics and its application to perceptual grouping for point clustering and symmetry and good continuation grouping of edge elements.

2.1 Architecture

2.1.1 Input and Features

In order to measure the similarity of perceived stimuli, it is necessary to extract a set of parameterized feature vectors from the input image:

$$\mathbf{m}_r \in \mathcal{M}, \qquad 1 \leq r \leq N, \tag{1}$$

where $\mathcal{M} \subseteq \mathbb{R}^n$ is the set of feature vectors, and N the cardinality of $\mathcal{M}$. Furthermore, each r is associated with a position $\mathbf{r} \in \mathbb{R}^2$ which states from where in the input image the feature was taken. This makes it possible to project the groups found during the grouping process back into the imageplane, i.e. the image itself can be divided into a set of groups. The gathered features can represent any modality, a few suggestions are:

- **Spatial information:** Each stimulus can be described by its position in the image. Therefore, $\mathbf{m}_r = (x, y)^T$, where x and y denote the x- and y-coordinate of the stimulus' position.
- **Colour:** Image regions can be described by their colour: $\mathbf{m}_r = (r, g, b)$, where r, g, b denote the values for the red, green and blue channel, respectively.
- **Textual information:** Surfaces are usually characterized by a specific textual appearance. So, $\mathbf{m}_r = \mathbf{t}_{\text{text}}$, where $\mathbf{t}_{\text{text}}$ is a suitable local texture description.
- **Edge information:** Objects usually form closed contours, which turn up through edge detection. So, $\mathbf{m}_r = (x, y, \theta)^T$, where x, y denote the position, and θ the orientation of an edge element.

The above list is only a small selection of practicable features. The combination of different modalities into a new attribute is also feasible, e.g. the combination of colour and position.

After the features were extracted during a preprocessing stage, to each of them is assigned a scalar intensity value h_r. Therefore, the input to the CLM is characterized by

$$I = \{(\mathbf{m}_r, h_r)\}, \quad r \in \{1, \ldots, N\} \tag{2}$$

So, the question is now, how to actually find a good perceptual organization of this feature set.

2.1.2 Interactions and Energy Function

Reconsidering the ideas of the Gestaltist's again, we can say that perceptual grouping involves the integration of sensual information into higher units, or "Gestalten". This unit formation is characterized by some sort of energy minimization, which mirrors the tendency to form simple and highly regular units.

To obtain a more quantitative mathematical model, the grouping problem can be formulated in the following way [Wer96]: A good perceptual organization of the feature set $\mathcal{M}$ is achieved if a partition of $\mathcal{M}$ into L disjoint subsets $\mathcal{M}_\alpha, \alpha \in \{1, \ldots, L\}$ is found, such that the sum of the grouping energies $\sum_\alpha E(\mathcal{M}_\alpha)$ is minimal. Each $\mathcal{M}_\alpha$ then corresponds to a salient "Gestalt". This process of energy minimization is devolved upon the recurrent architecture of the CLM.

The CLM consists of L *layers* with index α, each containing N formal neurons with non-negative activity $x_{r\alpha} \geq 0$. The second structure in addition to the layers are the *columns*: Neurons sharing the same position r describe a column with index r. Therefore, we have N columns with L neurons each. Each input component $(\mathbf{m}_r, h_r)$ is then associated with its corresponding column r – see Figure 4 for a visualization of the CLM.

Two types of interaction are involved in the construction of the energy-function: Firstly, a *vertical* interaction among the neurons of a column sustains the *superposition condition*

$$\sum_{\alpha=1}^{L} x_{r\alpha} = h_r, \tag{3}$$

i.e. the sum over all activities in a column r has to be equal to the input intensity h_r. By meeting this constraint the pattern of the input intensities is divided upon the L layers.

Secondly, the grouping energy of a layer's activity pattern is determined by a pairwise *lateral* feature interaction function f: Activities belonging to similar features – which therefore might belong to the same perceptual group – are "bound together" by a positive interaction. Activities belonging to diverse features are separated by a negative interaction. Hence, for each input image a symmetric $N \times N$ interaction matrix with elements $f(\mathbf{m}_r, \mathbf{m}_{r'}) = f_{rr'} = f_{r'r}$ is computed.

So, taking the lateral and the vertical interaction together we arrive at the following energy function for the CLM:

$$E = \frac{J_1}{2} \sum_r \left(\sum_\beta x_{r\beta} - h_r \right)^2 - \frac{1}{2} \sum_\alpha \sum_{rr'} f_{rr'} x_{r\alpha} x_{r'\alpha} \tag{4}$$

The first term of (4) is a constraint term corresponding to the superposition condition (3). The second term measures the sum of all grouping energies of each layer. The parameter J_1 controls the coupling between the superposition constraint and the overall grouping energy. Unfortunately, it is not clear, how the interaction function f has to be choosen in order to get a desired grouping behaviour. Once a suitable function is found, the problem to actually find the minimum of (4) remains. This is a nontrivial problem, because the complexity of the energy function makes it impossible to find a general analytical solution. Therefore, we have to rely on numerical algorithms, which takes us to the next section.

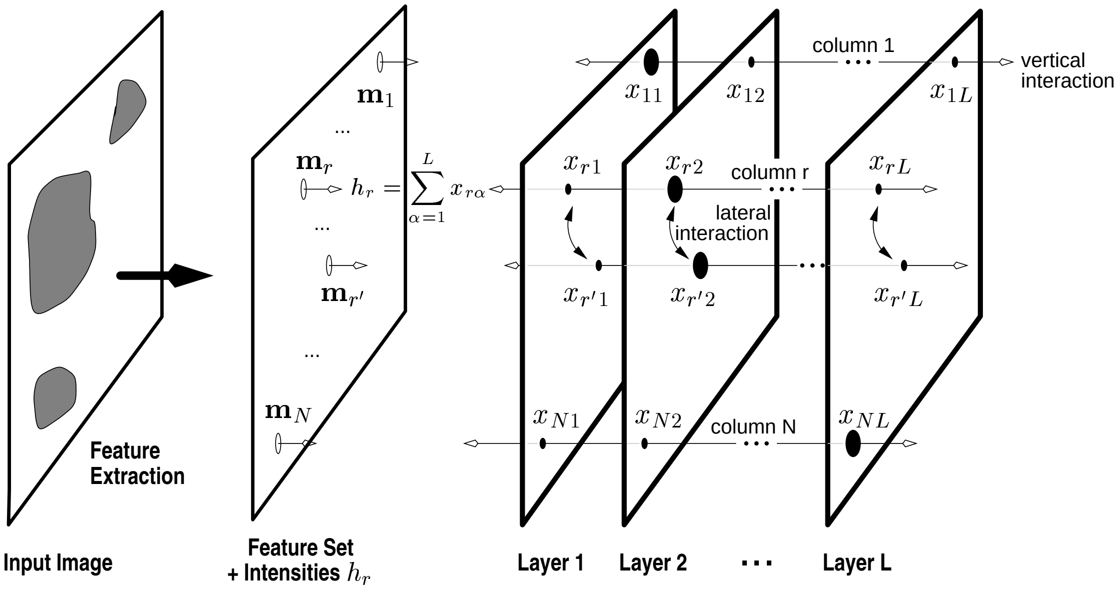


Figure 4: The architecture of the Competitive Layer Model is characterized by two types of interaction: Firstly, all neurons of a column *r* are *vertically* competing amongst all layers and secondly in every layer α each neuron $x_{r\alpha}$ is *laterally* interacting with all the other neurons in that layer.

2.2 Dynamics

2.2.1 Heat Bath Update

A method rooted in theoretical physics for the simulation of large scale dynamical systems is called *heat bath method*. The key point of this method is to randomly select a state variable and update its value in such a way that a local energy is minimized [KM97].

This method can be transfered to the CLM in the following way: The dynamics of the CLM reaches a stable state if $\dot{\mathbf{x}} = 0$. Since

$$\dot{x}_{r\alpha} = -\sigma_{r\alpha}\frac{\partial E}{\partial x_{r\alpha}} = \sigma_{r\alpha} \cdot \left(J_1(h_r - \sum_{\beta} x_{r\beta}) + \sum_{r'} f_{rr'}x_{r'\alpha} \right), \tag{5}$$

we find that $\dot{x}_{r\alpha}$ is a linear function of $x_{r\alpha}$, which gives us the possibility to solve $\dot{\mathbf{x}} = 0$ locally, i.e. we look for a value of $x_{r\alpha}$ such that its derivative with respect to time becomes zero:

$$\begin{aligned} 0 \overset{!}{=} \dot{x}_{r\alpha} &= J_1(h_r - \sum_{\beta} x_{r\beta}) + \sum_{r'} f_{rr'}x_{r'\alpha} \\ &= J_1(h_r - \sum_{\beta\neq\alpha} x_{r\beta}) - J_1 x_{r\alpha} + \sum_{r'\neq r} f_{rr'}x_{r'\alpha} + f_{rr}x_{r\alpha} \\ &= x_{r\alpha}(f_{rr} - J_1) + J_1(h_r - \sum_{\beta\neq\alpha} x_{r\beta}) + \sum_{r'\neq r} f_{rr'}x_{r'\alpha} \\ \Leftrightarrow \quad x_{r\alpha} &= \frac{J_1(h_r - \sum_{\beta\neq\alpha} x_{r\beta}) + \sum_{r'\neq r} f_{rr'}x_{r'\alpha}}{J_1 - f_{rr}} \end{aligned} \tag{6}$$

The idea is now to use this as an update rule for an asynchronous dynamics: With probability $p_{r\alpha} = \frac{1}{NL}$ a neuron $x_{r\alpha}$ is selected and its state $x_{r\alpha}(t)$ is updated to a new state according to:

$$x_{r\alpha}(t+1) = \sigma\left(\frac{J_1(h_r - \sum_{\beta\neq\alpha} x_{r\beta}(t)) + \sum_{r'\neq r} f_{rr'}x_{r'\alpha}(t)}{J_1 - f_{rr}} \right), \tag{7}$$

where σ keeps up the constraint $x_{r\alpha} \geq 0$:

$$\sigma(x) = \begin{cases} 0 & \text{if } x < 0 \\ x & \text{otherwise} \end{cases} \tag{8}$$

The question is now: Is the CLM's dynamic convergent if defined by the stochastical update rule given by (7)? Feng [Fen97] constructs a Lyapunov function for a neural network consisting of N neurons X_i and an asynchronous dynamics defined by

$$X_i(t+1) = f\left(\sum_{j=1}^{N} a_{ij} r_j X_j(t) + b_j\right), \tag{9}$$

where $X_i(t)$ is selected with probability $p_i > 0, \sum_i p_i = 1$ and all the other states are kept unchanged: $X_j(t+1) = X_j(t) \quad \forall j \neq i$. Furthermore, $A = (a_{ij}, i, j = 1, \ldots, N)$ is a symmetric $N \times N$ matrix, $r_i \geq 0, i = 1, \ldots, N$ and $f(x)$ is a continuous function that is strictly increasing if $\alpha \leq x \leq \beta$ and $f(x) = \alpha$ if $x \leq \alpha, f(x) = \beta$ if $x \geq \beta$. He shows that the function

$$L(\mathbf{X}(t)) = \sum_{j=1}^{N} \int_0^{X_j(t)} r_j f^{-1}(y) dy - \frac{1}{2} \sum_{j,i=1}^{N} a_{ji} X_j(t) X_i(t) r_j r_i - \sum_{j=1}^{N} r_j b_j X_j(t) \tag{10}$$

is a *supermartingale*, i.e. a stochastic process for which holds:

$$E(L(\mathbf{X}(t+1)) \mid \mathcal{F}_t) \leq L(\mathbf{X}(t)), \tag{11}$$

where L is a measurable function, $\mathcal{F}_t = \sigma(\mathbf{X}(1), \ldots, \mathbf{X}(t))$ the sigma algebra generated by $\mathbf{X}(s), s = 1, \ldots, t$ and $E(\cdot \mid \mathcal{F}_t)$ is the conditional expectation with respect to the sigma algebra $\mathcal{F}_t$. With other words: The function given by (10) might increase occasionally, but in the average L decreases with respect to its dynamical evolution, i.e. L is a sort of "stochastic Lyapunov function" for a dynamical system defined by (9). Note, that when f is differentiable, L is identical with Hopfield's energy function for binary and continuous neurons [Gro88].

Luckily, the heat bath update rule given by (7) is of the same form as (9). If we identify the position i with position $r\alpha$, we can set

$$\begin{aligned} r_i &= 1 \\ a_{ij} &= \frac{1}{J_1 - f_{rr}} \left(\delta_{\alpha\alpha'} f_{rr'} (1 - \delta_{rr'}) - J_1 \delta_{rr'} (1 - \delta_{\alpha\alpha'})\right) \\ b_i &= \frac{J_1 h_r}{J_1 - f_{rr}}, \end{aligned} \tag{12}$$

where δ_{ij} denotes the Kronecker symbol. The constraint function σ (8) corresponds to the nondifferentiable function f in (9). However, f is also bounded from above. This is not the case for σ, which is only bounded from below by 0. Fortunately, the Legendre-Fenchel transformation Feng uses in his proof

$$\int_0^{X_k(t+1)} f^{-1}(y) dy + \int_0^{u_k(t)} f(x) dx = u_k(t) X_k(t+1), \quad t \geq 0, \tag{13}$$

where

$$u_k(t) = \sum_{j=1}^{N} a_{kj} r_j r_k X_j(t) + b_k \tag{14}$$

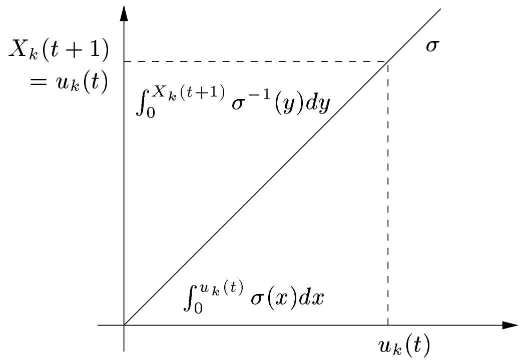

$X_k(t+1)$
$= u_k(t)$
σ
$\int_0^{X_k(t+1)} \sigma^{-1}(y)dy$
$\int_0^{u_k(t)} \sigma(x)dx$
$u_k(t)$

Figure 5: Explanation of the Legendre-Fenchel transformation for σ: The sum of the two integrals corresponds to the area of the square and is therefore equal to $u_k(t) \cdot X_k(t+1)$.

also applies to σ, since $\sigma(u_k(t)) = X_k(t+1)$ for $u_k(t) \geq 0$. Therefore, as can be seen in Figure 5, (13) also holds for σ.

Therefore, we have that L given by (10) is also a supermartingale for the dynamics defined by the proposed stochastical update rule (7). However, it is not clear that the proposed Lyapunov function (10) corresponds to the energy function (4). To show that the proposed stochastical update rule actually minimizes the CLM's energy function, we examine the scalar product of $\Delta\mathbf{x}$ and $\vec{\nabla} E$. Because $\mathbf{x}$ changes only in the randomly selected component $r\alpha$, whereas all the others are kept unchanged, we can write:

$$\begin{aligned}\Delta\mathbf{x} &= x_{r\alpha}(t+1) - x_{r\alpha}(t) \\ &= \frac{J_1(h_r - \sum_{\beta\neq\alpha} x_{r\beta}(t)) + \sum_{r'\neq r} f_{rr'}x_{r'\alpha}(t)}{J_1 - f_{rr}} + \frac{(-J_1 + f_{rr})x_{r\alpha}(t)}{J_1 - f_{rr}} \\ &= \frac{J_1(h_r - \sum_{\beta} x_{r\beta}(t)) + \sum_{r'} f_{rr'}x_{r'\alpha}(t)}{J_1 - f_{rr}} = \frac{1}{f_{rr} - J_1}\vec{\nabla}_{r\alpha} E \end{aligned} \tag{15}$$

Therefore,

$$\langle \Delta\mathbf{x}, \vec{\nabla} E \rangle = \frac{1}{f_{rr} - J_1}(\vec{\nabla}_{r\alpha} E)^2 \leq 0, \tag{16}$$

if $J_1 > f_{rr}$. Note, that (15) is only valid if the activity $x_{r\alpha}$ has not reached its lower boundary $x_{r\alpha} \geq 0$ yet. Otherwise, there are two cases possible:

1. $x_{r\alpha}(t) = 0, x_{r\alpha}(t+1) = 0 \quad \Rightarrow \quad \Delta\mathbf{x} = 0 \quad \Rightarrow \quad \langle \Delta\mathbf{x}, \vec{\nabla} E \rangle = 0$
2. $x_{r\alpha}(t) > 0$, $x_{r\alpha}(t+1) = 0$: In this case, the sign of $\Delta x_{r\alpha}$ is the same as for the unconstrained case, therefore, $\langle \Delta\mathbf{x}, \vec{\nabla} E \rangle \leq 0$ still holds.

So, for each update step $\Delta\mathbf{x}$ and $-\vec{\nabla} E$ point into the same half-plane, which means that the activities do not go "uphill" in the energy function:

These two properties – the existence of a global Lyapunov function plus that the direction of $\Delta\mathbf{x}$ points into the "right" direction – guarantee that the proposed stochastical update rule behaves as desired.

Further Speed Up

Reconsidering the architecture and the grouping assignment property of the Competitive Layer Model, we have that for an attractor state maximally N of the NL neurons are

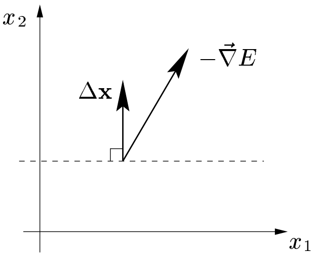


Figure 6: For each update step, the activities do not go "uphill" in the energy function: $\langle \Delta\mathbf{x}, -\vec{\nabla}E \rangle \geq 0$.

active. Furthermore, an examination of (7) shows that the activities $x_{r\alpha}$ reach their maximal values very fast: If all activities are initialized with small random values $x_{r\alpha} \ll h_r$ then the sums in (7) are close to zero. Hence,

$$x_{r\alpha}(t+1) \approx \frac{J_1 h_r}{J_1 - f_{rr}} \approx h_r \tag{17}$$

If then a neuron of the same column r is picked up for an update, then the constraint term $\sum_\alpha x_{r\alpha} = h_r$ is already satisfied and therefore its new value is close to zero. This behaviour was also observed during the simulations. Figure 7 shows a typical development of the activities during a stochastical update run. It can be seen that already after the first time step the activities reach a high value of $\approx h_r$.

If we consider a neuron, which reaches its lower boundary it is likely, that the constraint σ in (7) will be active. Therefore, the expensive computation of the sums would not be used at all. According to Feng [Fen97], the dynamics allows an arbitrary update probability $p_{r\alpha}$. So, the idea is now to omit the computation of $x_{r\alpha}(t+1)$ if $x_{r\alpha}(t) = 0$. However, a complete left out would not be very clever, since activities which were initially set to zero would be doomed to stay there. Therefore, zero activities are updated with a certain probability p_{up}. Figure 8 shows the relation between p_{up} and convergence time t_{Conv}. According to this data a value of $p_{\text{up}} \approx 0.2$ seems to be a good choice. In Figure 9 three typical runs for different values of p_{up} are plotted.

The data show, that the modification of the stochastical update mode gained a further speed up by a factor of ≈ 2.

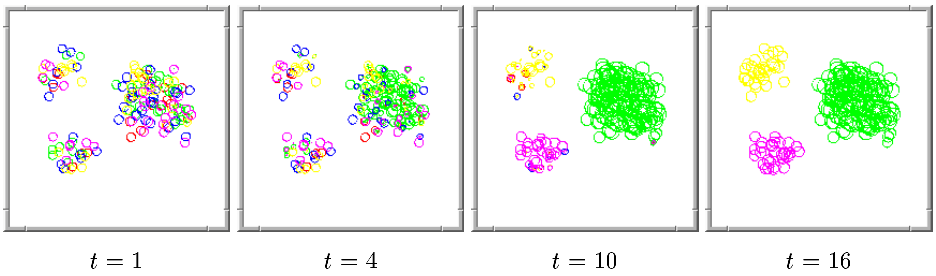


Figure 7: Time development of activities during a stochastical update run: The interaction function for this example was constructed using only the information about the spatial position of the stimuli. (For details see [WSR97]). The values for the times t correspond to the number of NL activity updates. The size of the circles are proportional to the activities of the neurons. It can be seen, that the neurons reach their maximal activity already after the first update step. Note, that this result was obtained with $p_{\text{up}} = 0.2$.

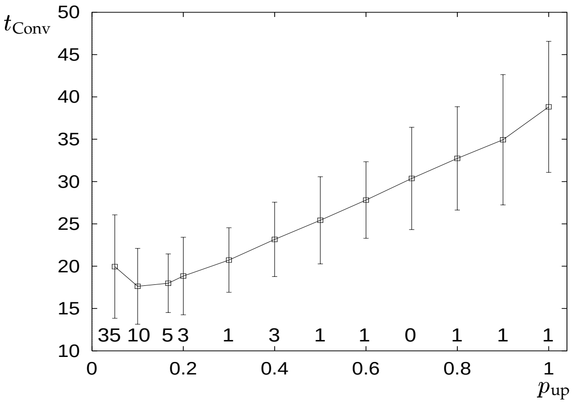


Figure 8: Relation between p_{up} and convergence time t_{Conv}: The squares depict the mean convergence for 100 runs of the CLM applied to the same input as above. The numbers below the errorbars correspond to the number of runs, where the stochastical update did not converge to the global minimum. For too small values of p_{up} the grouping result is getting worse, because no activities are actually updated. Therefore, $\Delta E = 0$ and the system stops although the global minimum is not yet reached.

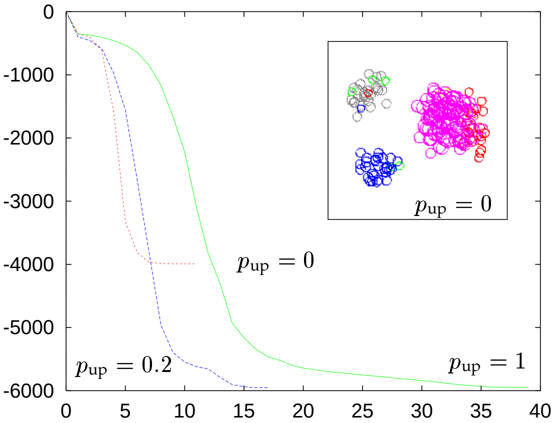


Figure 9: Three typical runs for different values of p_{up}: For $p_{\text{up}} = 0$ the CLM is not able to find the global minimum. The corresponding grouping result is shown in the upper right. For $p_{\text{up}} = 0.2$ the grouping process is about a factor 2 faster than the simple stochastical update with $p_{\text{up}} = 1$.

3 Perceptual Grouping with the CLM

The perceptual grouping of images involves the processing of several features describing the picture's contents. As Bergen [Ber91] says:

> *We have many ways of describing objects of the visual world. We can talk about their size, shape, colour and position in space. [...] Even with all these words, however, our description will be incomplete and unreal without some reference to the texture of the surfaces of the object. They may be smooth or rough, glossy or matt, they may have the more complex qualities of leather, satin, velvet, paper, slate, wood, sand, wax, fur, or hundreds of other substances.*

Consequently, a lot of researchers have discussed the theory of visual texture perception. We propose a model based upon the "Back Pocket Model" which texture perception researchers "routinely pull out from their back pocket" [CL91]. This model applies a set of linear filters motivated by cells found in the visual cortex of mammals. The response of those filters is then used to compute a high dimensional texture feature. We will present further stages of feature processing in order to apply the CLM to the obtained set of attributes. Furthermore, we will construct a feature interaction function which is capable of measuring the textual appearance of image regions and which is used for the perceptual grouping with the CLM.

Taking the texture perception model, the preprocessing of the data and the CLM together we arrive at a model of visual perception which is applied to a variety of test images. These include stimuli found in the classical literature about texture segregation, natural textures taken from the widely cited Brodatz album [Bro66] and a set of images used for the visualization of the Gestalt laws.

Furthermore, it is shown that the proposed system is also applicable to a restricted set of simple scenes of a toy world, where it is able to dissolve ambiguous informations.

Most of the images displayed in the rest of this report contain a grey border. These borders are not part of the pictures. They are just added to separate the figures from the background and to provide a uniform visual appearance for all examples.

3.1 Texture Perception

Nowadays the assumption that the human visual system operates in two modes is a matter of records. These two modes can be described as *preattentive* and *attentive* [Tre86]. In the former mode differences in a few local structural features are detected over the entire visual field. Presumably this mode works in parallel, i.e. the total visual field is searched instantaneously. The preattentive mode then gives important clues for the attentive mode, which in turn works sequential. In this second mode, the attention is concentrated on a small region of interest, and only here a recognition of objects is possible.

Psychophysical studies have revealed, that during the preattentive phase not only basic features such as colour or brightness are detected, but also more complex features like orientation or texture [BJ83]; for a review see [Ber91]. An example for this phenomenon is shown in Figure 10.

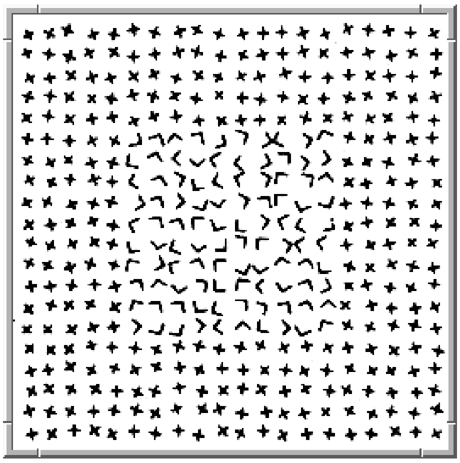


Figure 10: Example of preattentive texture based segregation (taken from [Ber91]): We immediately see a center region which segregates from the region surrounding it. Both regions do not differ in brightness or colour. The segregation is only defined by a difference in local spatial structure. Note, that the exact form of the border cannot be seen instantaneously, in order to construct the precise shape of the central region we probably need attentive mechanisms. The L's and X's are randomly orientated and a small jitter is imposed on their spatial position. Note, that the length of the line elements is the same for both patterns.

Much research on visual texture perception was pioneered by Bela Julesz and his colleagues. Julesz was one of the first who systematically investigated the abilities of humans to discriminate between different textures. His work goes back to 1962 where he studied the perception of random dot textures. Later he proposed, that texture discrimination could be explained in terms of first order differences between local features called *textons* [Jul81]. Textons are defined as elongated blobs of specific color, orientation, length and width, along with their terminators and line crossings. However, later it was shown that the texton-theory is inconsistent with the segregation of certain patterns [Not91].

Recent studies of texture perception have demonstrated the importance of spatial frequency information in connection with texture segregation phenomena. It was shown by various investigators that texture segregation can be explained on the basis of spatial-frequency and orientation-selective linear filters. Therefore, many researchers have proposed computational models based on a standard model called "Back Pocket Model" [BCG90, Cae88, FS89, HPB96, IRSB95, JF91, LB91, MP90, MC93, RW91, Tur86], which uses scale- and orientation-selective mechanisms. (See section 3.3 for a detailed description of the model). These mechanisms – also commonly referred to as channels, pathways, detectors, units or neurons – are believed to exist at a relatively low level in the visual system. Therefore, the next section takes us to early vision models.

3.2 Motivation for the Feature Extraction

3.2.1 Early Vision

The human visual system has drawn the attention of many researchers and the visual cortex is one of the most studied areas in the brain. It is certainly not in the scope of this report to present the tremendous amount of biological information that has been gathered so far. The interested reader is referred to the excellent summaries of Valois & Valois [VV88] and Lennie [Len80]. However, to give a rough idea of its anatomy, a simple sketch of the human visual pathway is depicted in Figure 11.

Of special interest in the light of texture perception are the studies of Hubel and Wiesel which go back to 1959. They found that cells in the primary visual cortex in cat, and the vast majority of those in monkey, respond to lines of only certain orientations. Hubel and Wiesel differentiated between *simple* and *complex* cells. The former only respond to stimuli at a particular location in the visual field and are generally considered linear integrators of luminance within their receptive fields. In contrast, complex cells exhibit a number of fundamentally nonlinear behaviours and also respond across a region much wider than the optimal stimulus. Hubel and Wiesel therefore suggested that complex cells receive

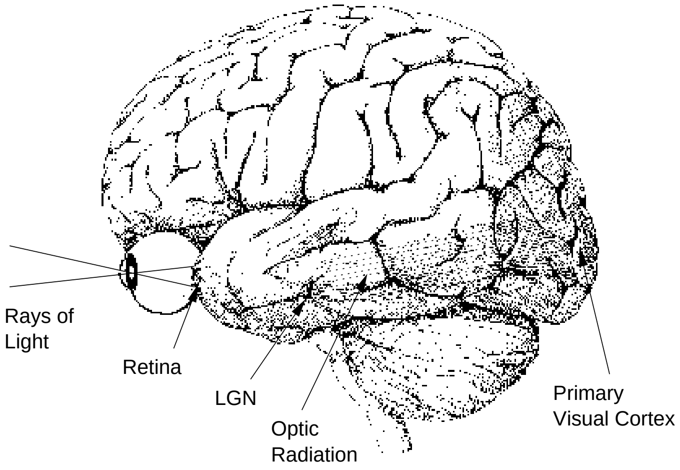


Figure 11: The visual pathway in the human brain: Light falls through the eye on the retina, where cells of different types encode features like colour and brightness. From there the information passes through the optic nerve to the Lateral Geniculate Nucleus (LGN), from where it is projected via the optic radiation to the visual cortex.

inputs from lower level simple cells. However, recent studies have shown, that such a pure hierachical model is probably not justified [MRA97]. In the following we will concentrate on simple cells.

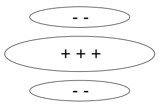


Figure 12: Sketch of the receptive field of a simple cell

Mathematically, a simple cell can be described functionally in terms of its *receptive field*, which depicts the response of that neuron to a small spot of light depending on the stimulus position. An example for the receptive field of a simple cell is shown on the left. The central oval region depicts those locations, where the neuron is excited by a spot of light, the surrounding regions labelled with '-' inhibit the cell's activity. The receptive field of a simple cell is localized in the space-domain (location specific) as well as in the frequency-domain (orientation and scale specific).

In 1946 Gabor showed that analogous to Heisenberg's uncertainty principle in quantum physics, an arbitrary accurate localization in both domains cannot be achieved simultaneously. Gabor proposed a certain class of functions achieving the theoretical lower limit of joint uncertainty in time and frequency:

$$g(t) = e^{-\frac{(t-t_0)^2}{\alpha^2}+i\omega t}, \tag{18}$$

which in complex notation describes the modulation product of a complex exponential wave with frequency ω and a Gaussian envelope of duration α occurring at time t_0.

Daugman [Dau85] extended Gabor's work to a two-dimensional frame and demonstrated that 2D Gabor filters have optimal joint resolution in such a sense that they minimize the product of effective areas occupied in the 2D space and 2D frequency domains. A two-dimensional Gabor function $g(x, y)$ and its Fourier transform $G(u, v)$ can be written as:

$$g(x,y) = e^{-\left(\frac{(x-x_0)^2}{\sigma_x^2}+\frac{(y-y_0)^2}{\sigma_y^2}\right)} e^{-2\pi i\omega\,(x-x_0)} \tag{19}$$

$$G(u,v) = e^{-\left(\frac{(u-\omega)^2}{\sigma_u^2}+\frac{v^2}{\sigma_v^2}\right)} e^{-2\pi i\,(x_0(u-\omega)+y_0 v)}, \tag{20}$$

where (x_0, y_0) is the center of the receptive field in the spatial domain, σ_x and σ_y are the widths of the Gaussian envelope along the x and y axes, respectively, ω is the frequency of a complex plane wave along the x-axis, and $\sigma_u = \frac{1}{2\pi\sigma_x}$, $\sigma_v = \frac{1}{2\pi\sigma_y}$. Filters with arbitrary orientations can then be obtained via a rigid rotation of the x-y coordinate system. Note, that (19) describes a complex function, which contains an even symmetric cosine and an odd symmetric sine component. Such a filter is also called a *quadrature filter*. A plot of an even symmetric 2D Gabor is shown in Figure 13. Daugman now proposed that

> *[...] the visual system is concerned with extracting information jointly in the 2D space domain and in the 2D frequency domain, and because of the incompatibility of these two demands, has evolved towards the optimal solution via 2D channels that roughly approximate 2D Gabor filters.*

This suggestion was based on the physiological experiments of Jones and Palmer [JP87], who found that the great majority of their cells could be well fitted by 2D Gabor elementary functions. Since then many researchers have used 2D Gabors as "biological motivated" filters.

In contrast to this opinion there also exist studies which criticize that the lower resolution limit in the conjoint space is only achieved by 2D Gabors in their complex form, whereas the receptive field of simple cells can be described only by real-valued functions [Sto90]. However, there is also evidence that simple cells exist in quadrature-phase, i.e.

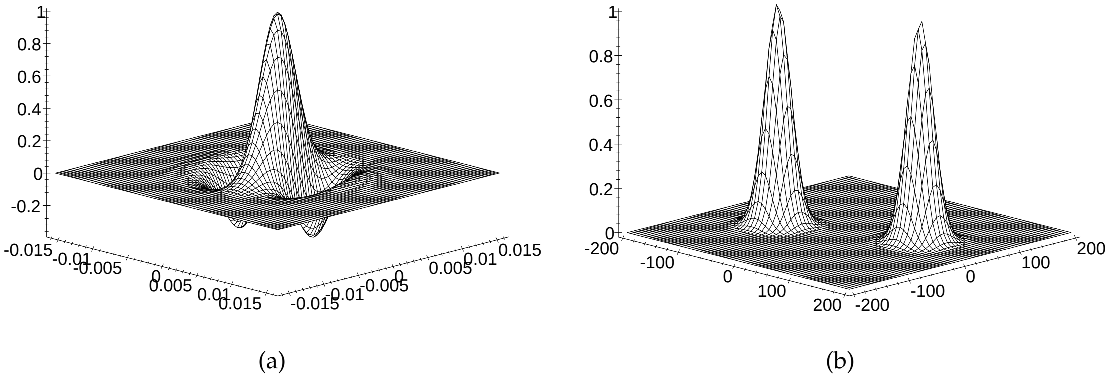


Figure 13: An even symmetric two-dimensional Gabor function (a) and its Fourier transform (b). Note that (a) resembles a receptive field as shown in Figure 12. The figure shows that the size of the receptive field is limited in both domains.

adjacent neurons have spatial receptive fields which share the same location in space and the same orientation preference but differ by 90 deg in their phase [PF81].

Other authors have proposed functions such as differences of Gaussians (DOGs) or differences of offset Gaussians (DOOGs) to model the receptive fields of simple cells [Sto90]. We believe that the precise shape of these functions is not a critical choice. All of the proposed functions exhibit very similar properties. All of them are orientation and scale sensitive and should not make much differences for the outcome of any computational model which uses “simple cells” as feature extractors.

Because Gabor functions are mathematically simple and heavily used in computer vision systems, we will follow the same way and concentrate on 2D Gabor functions as a model for simple cells in primary visual cortex.

3.2.2 The Watson Model

In the section above we have motivated the class of functions which we use to model the receptive fields of simple cells. The question is now, how these feature extractors should be used to process a given input image.

Reexamining the definition (19) or (20) of the 2D Gabor functions, we see that they have four degrees of freedom if we just want to characterize the form of the receptive field and omit its spatial location (x_0, y_0). For the frequency domain these are $\omega, \sigma_u, \sigma_v$ and the orientation Θ. See Figure 14 for a visualization of these parameters in the frequency domain.

A common way to describe the size of the receptive fields is to measure their frequency and orientation *bandwidth*, which are defined as:

$$B_{\mathrm{f}} = \log_2 \left(\frac{\omega + \sqrt{\ln 2}\,\sigma_u}{\omega - \sqrt{\ln 2}\,\sigma_u} \right) \tag{21}$$

$$B_{\Theta} = 2 \tan^{-1} \left(\frac{\sqrt{\ln 2}\,\sigma_v}{\omega} \right), \tag{22}$$

where B_{f} is measured in octaves and B_{Θ} in degrees.

Daugman [Dau88] and others [BCG90, MM96c, JF91] have proposed that an ensemble of simple cells is best modelled as a family of self-similar 2D Gabor filters. We can construct

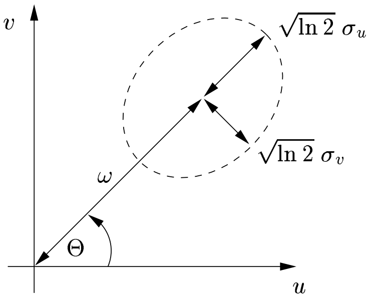


Figure 14: Degrees of freedom of a 2D Gabor function in the frequency domain: The dashed ellipse indicates the half-amplitude contour of the Gaussian (corresponding to one peak in Figure 13(b)) and is characterized by σ_u and σ_v. The frequency ω denotes the distance from the origin, and the angle Θ the orientation of the Gabor.

such a set of self-similar functions, commonly referred to as *Gabor wavelets* by a scaling and rotation of the x-y coordinate frame:

$$\begin{aligned} g_{mn}(x,y) &= g(x',y'), \\ x' &= a^{-m}(x\cos\Theta_n + y\sin\Theta_n) \quad \text{and} \\ y' &= a^{-m}(-x\sin\Theta_n + y\cos\Theta_n), \end{aligned} \tag{23}$$

where $a > 1$ is the scaling factor, $\Theta_n = \frac{n2\pi}{K}$, $n = 1, \ldots, K$ is the rotation angle, where K is the total number of orientations, and $m = 1, \ldots, S$, where S is the number of scales.

Watson [Wat87] and others [Lee96, Dau88] have demonstrated that a set of filters arranged in a daisy-like pattern as shown in Figure 15 is able to give a good description of an image coding scheme. It is "good" in such a sense that a sparse sampling of the phase space (which is spawned by m, n, x_0 and y_0) is sufficient for a complete representation of arbitrary image data. Furthermore, physiological data suggests that the cortical sampling density is far greater than the density required to obtain a complete representation. Lee [Lee96] concludes in his paper that

> *the visual cortex is primarily concerned with extracting and computing perceptual information [...] The simple cells, modelled by Gabor wavelets [...] facilitate these computations by providing an ideal medium for representing surface texture and surface boundary with high resolution.*

3.3 Obtaining the Features for Perceptual Grouping

So far we have motivated a construction scheme for a set of filters which is suitable as an image representation code. Because such a set of filters is able to reconstruct an image without essential loss of information, the filter responses necessarily contain the information for texture segregation. This has lead to a general model that "texture perception researchers routinely pull out from their back pocket to make sense of new instances of preattentive texture segregation" [CL91]. This Back Pocket Model generally consists of three stages:

1. A variety of linear filters is applied to the input image, resulting in a set of neural images, one for each filter. They are commonly referred to as *channels*.

2. Local transformation of each neural image by some sort of energy measure (e.g. taking the square or absolute value).

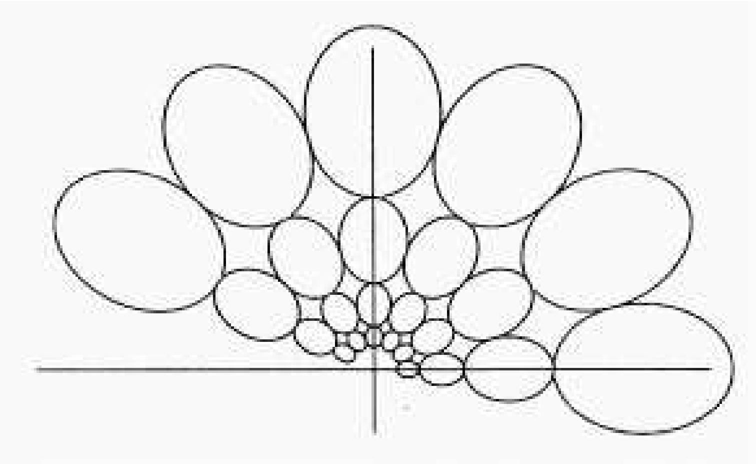


Figure 15: 2D Gabor filter arrangement in a daisy-like pattern (taken from [MM96c]): As in Figure 14 the ellipses indicate the half-peak contours of the Gabor filters in the frequency domain. The figure describes 6 orientations at 4 different scales. The advantage of this construction scheme is that almost the whole frequency space is covered, whereas the individual filters have as little overlap as possible. Note, that one lobe of the Gabors is omitted. This can be done without loss of information for even symmetric filters. (The reason for this is that the Fourier transform of a real-valued image has conjugate symmetry. Therefore, the product of the symmetric filter and the Fourier transform of the image also has conjugate symmetry and therefore contains redundant information.)

3. Passing of the entire set resulting from step 2 to a segmentation system whose purpose is to partition the visual field.

It is not clear, how exactly these steps have to look like in order to achieve results consistent with human texture perception. Furthermore, there is evidence, that the sort of nonlinearity described in step 2 is not sufficient to account for certain texture segregation phenomena.

In the following sections we will present the steps described above in more detail and propose some modifications to the model.

3.3.1 Channel Extraction

As we have seen in section 3.2.2, a set of self-similar 2D Gabor filters arranged on a daisy-like pattern in the frequency domain provides a good method to represent visual information in terms of early vision mechanisms. The question arises which parameters exactly we should use to generate such a set of filters, i.e. how many scales and orientations and which type of filters (symmetric, asymmetric or both) we should use.

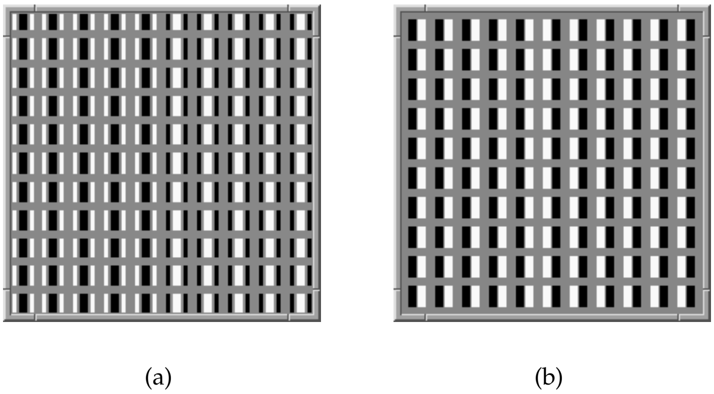
Even Symmetric vs. Odd Symmetric Mechanisms

There is strong evidence due to the results of Malik and Perona [MP90], that preattentive texture segregation is mainly based on even symmetric mechanisms. Based on experimental results [RHC88] they construct two different texture patterns, from which only one segregates preattentively. Then they show, that for the segregation only even symmetric mechanisms are relevant. We repeat their arguments in the following part.

Artificial textures can be easily constructed from so-called *micropatterns*. These in turn can be described by their symmetry:

1. Two micropatterns M_1 and M_2 are said to be *xy-mirror symmetric* (xy-ms) if

$$M_1(x) = -M_2(-x) \tag{24}$$



(a) (b)

Figure 16: Two artificial textures constructed of mirror symmetric micropatterns: **(a)** is constructed of xy-ms pairs as defined in (24) and segregates preattentively: the left hand side differs significantly from the right hand side, **(b)** is constructed of y-ms pairs as defined in (25). Note, that the border region between the two textures "pops out", but the left hand side of the texture does not segregate from the right hand side.

2. Two micropatterns M_1 and M_2 are said to be *y-mirror symmetric* (y-ms) if

$$M_1(x) = M_2(-x) \tag{25}$$

See Figure 16 for an example of such textures.

In order to achieve a segregation effect we need some sort of measure which gives us a different description for different textures. Since we do not want a descriptor which varies much if applied to a uniform texture, this measure has to work on an area at least the size of one micropattern. Otherwise, only local differences within one texture element would be measured. A widely used method to obtain such a measure is to compute the statistical properties of the signal, such as the mean value or standard deviation. In context of neural nets this approach is commonly referred to as *spatial pooling*, i.e. there is one neuron that sums up the output of several other units from a lower level. The actual pooling process used in our model is described in section 3.3.3.

Considering an xy-ms pair of texture elements, such as depicted in Figure 16(a), we see that a convolution with an even symmetric filter preserves xy-mirror symmetry. On the other hand, a convolution with an odd symmetric filter, turns it into a y-ms pair, as can be seen in Figure 17. Now, any two patterns with y-mirror symmetry necessarily have identical spatial averages – this follows directly from (25). This is not the case for xy-ms pairs, which have spatial averages of opposite sign. Therefore, to segment a texture consisting of xy-mirror symmetric pairs, we have to rely on even symmetric mechanisms. Note, that the above comments still hold, if we apply a nonlinear scaling function to the output of the filter responses.

The situation is reversed if we consider a y-ms pair of texture elements as shown in Figure 16(b). Convolution with an even or odd symmetric filter yields a pair of y-ms or xy-ms patterns, respectively – see Figure 18. Hence, segregation of y-ms patterns can only occur if the visual system uses the different outputs of odd symmetric channels.

Figure 16(a) segregates preattentively, Figure 16(b) does not. Therefore, Malik and Perona [MP90] state: "One could conclude from this result that odd symmetric mechanisms are not utilized in texture perception but that even symmetric are." They report that they have not found any textures for which odd symmetric mechanisms were necessary.

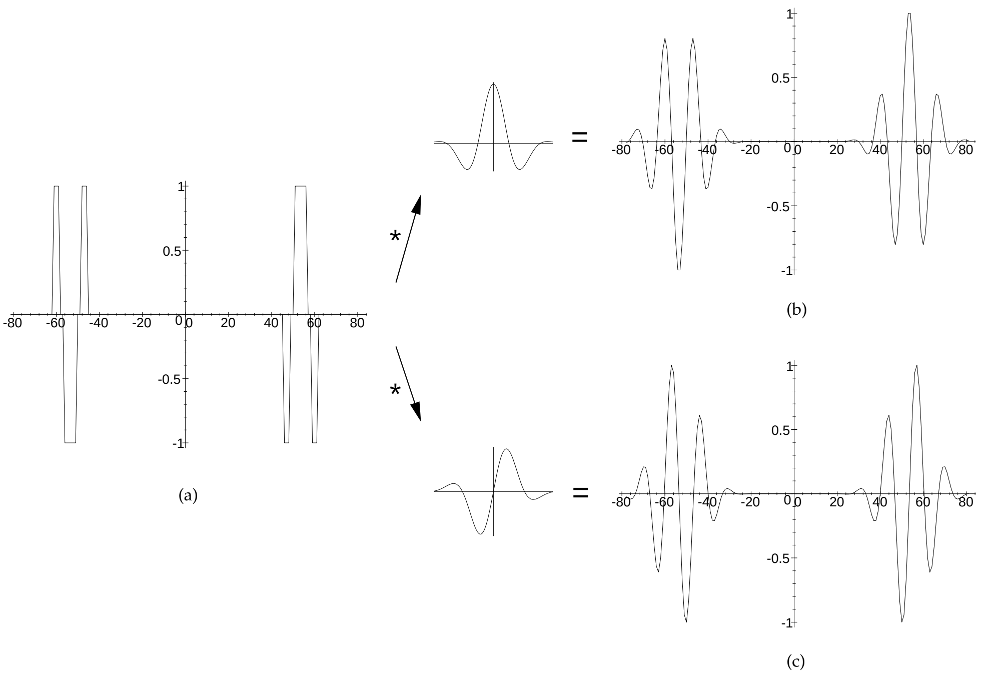


Figure 17: Convolution of an xy-ms pair of micropatterns with an even and odd symmetric filter, respectively: **(a)** shows the cross section of an xy-ms pair of micropatterns (see Figure 16(a)) along the x axis, **(b)** denotes the convolution of (a) with an even symmetric filter as depicted in the upper middle, **(c)** the convolution of (a) with an odd symmetric filter as shown in the lower middle. Note that (b) is also xy-ms, whereas the odd symmetric convolution "reversed" the symmetry and yielded a y-ms pair, where both patterns necessarily have identical spatial averages.

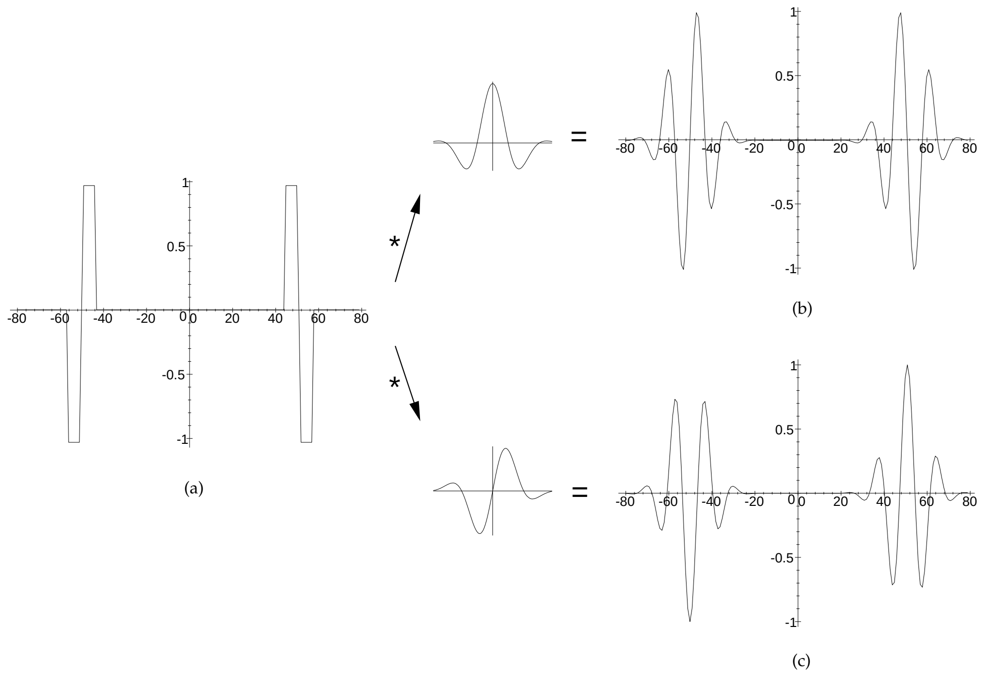


Figure 18: Convolution of an y-ms pair of micropatterns with an even and odd symmetric filter, respectively: **(a)** shows the cross section of an y-ms pair of micropatterns (see Figure 16(b)) along the x axis, **(b)** denotes the convolution of (a) with an even symmetric filter as depicted in the upper middle, **(c)** the convolution of (a) with an odd symmetric filter as shown in the lower middle. Again, the convolution with an even symmetric filter preserved the symmetry, whereas the odd symmetric filter reversed it.

This is the same observation we made in our experiments, and we therefore exclude odd symmetric mechanisms from our model.

An important behaviour of simple cells is that they have zero d.c. response, i.e. they do not respond to uniform luminance. This is also necessarily the case for odd symmetric filters, since $\int_{-\infty}^{\infty} g(x)\ dx = 0$, for any odd symmetric function g. On the other hand, for even filters we generally cannot guarantee this property. In order to reduce the computational complexity, we apply the set of filters in the frequency domain. By explicitly setting each filter at $(u, v) = (0, 0)$ to zero we force that their responses have zero mean and do not respond to uniform luminance. A filter response to a given input image is then equivalent to a convolution of the corresponding receptive field with this input.

Number of Scales and Orientations

We still have not specified the number of scales and orientations we should use for the filter bank. According to physiological experiments [VYH82, VAT82] the median frequency and orientation bandwidths of simple cells are 1.4 octaves and 40 deg, respectively. Because we only use even symmetric filters, two filters of orientations Θ and $\Theta + 180$ deg are equivalent. By choosing 5 equally spaced orientations we therefore arrive at a filter bank with their optimal stimuli separated by $180/5 = 36$ deg.

In order to achieve a good coverage of the frequency domain with little overlap between each filter we use the following scheme to construct the proposed daisy-like pattern:

$$\omega_m = \left(\frac{3}{4}\omega_{\text{max}}\right) 2^{-m+1} \tag{26}$$

$$\sigma_m = \omega_m \frac{2^{B_{\text{f}}} - 1}{\sqrt{\ln 2}\ (2^{B_{\text{f}}} + 1)}, \tag{27}$$

where ω_m is the frequency of the filter corresponding to scale m, $m = 1, ..., S$, ω_{max} is the highest spatial frequency in the input image, σ_m is the width of the corresponding Gaussian, and B_{f} is the frequency bandwidth. The different orientations are obtained via a rigid rotation of the (u, v) coordinate frame analogous to (23).

By choosing $S = 3$, $B_{\text{f}} = 1$, and an aspect ratio of one, i.e. $\sigma_u = \sigma_v$, we obtain a set of Gabor filters as shown in Figure 19. The figure shows, that the pattern actually achieves an almost coverage of the frequency domain. By applying this filter bank to an input image and summing up all responses, we obtain a reconstruction of this input, where only those spatial frequencies are missing which are not covered by our set of filters. An evaluation of this reconstruction can be used as a heuristic to estimate the numbers of scales we should use.

Figure 20 shows three different reconstructions of the texture in Figure 16(a). The first was obtained by a filter bank using only two scales, i.e. the same set as shown in Figure 19 was used but the innermost ring denoted by C was omitted. The second and third reconstruction used three and four scales, respectively. It can be seen, that two scales are not sufficient to capture the structure of the original image. The center region of the micropatterns is too large to be covered by the receptive fields. Three scales give a still somewhat distorted image, but in contrast to two scales the structure of the micropatterns is completely captured. The usage of four scales does not significantly improve the quality of the reconstruction. Since in our experiments no textures with larger structures than shown in this example were used, we will rely on a filter bank consisting of three scales.

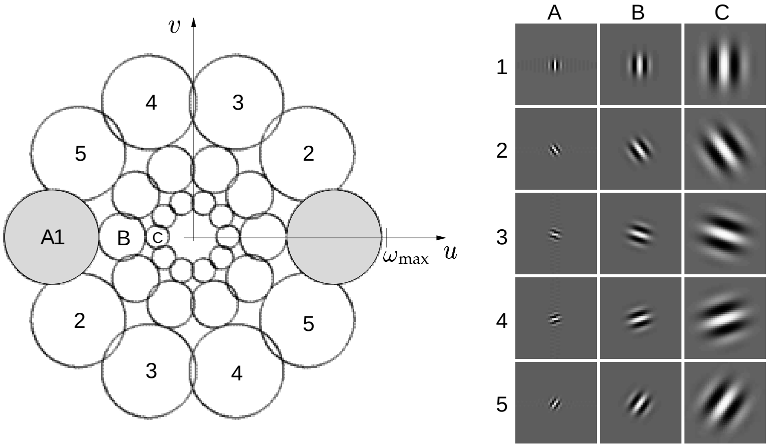
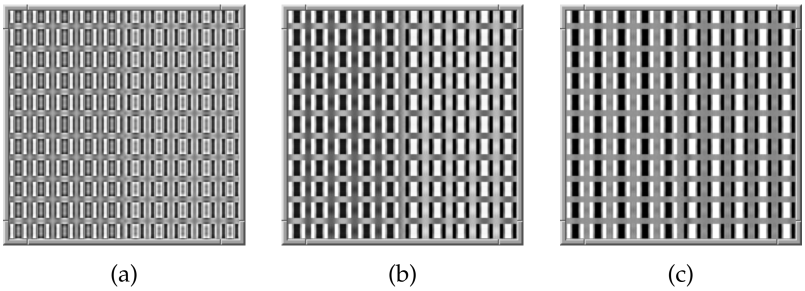


Figure 19: The set of 2D Gabor filters used for the feature extraction: On the left hand side the daisy-like pattern of the frequency domain is shown. Again, the circles denote the half-peak contours of each 2D Gabor. Their corresponding receptive fields in the spatial domain are depicted on the right. Note, that small receptive fields in the spatial domain have a large counterpart in the frequency domain, and vice versa. This also expresses the uncertainty relation and shows that information content in spatial and frequency domain are inversely related.



(a) (b) (c)

Figure 20: Reconstruction of an input image: The images (a)–(c) were obtained by the summation over the responses of all filter elements to the input shown in Fig. 16(a) For visualization issues, the reconstructions were scaled to full contrast. In **(a)** only two scales were used, **(b)** was obtained using the three different scales shown in Fig. 19, and in **(c)** a further scale was added to the set. See the text below for an interpretation.

Taking everything together, we apply a set of 15 even symmetric 2D Gabor filters with forced zero d.c. response to a given input image. In the frequency domain these filters are arranged in a daisy-like pattern with 5 orientations and 3 scales. Therefore, for each input image we obtain a set of 15 filter response images, or channels, as shown in Figure 21.

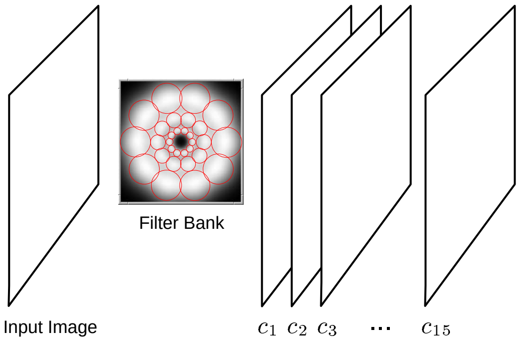


Figure 21: Obtaining the channel responses: The filter bank is applied to a given input image, resulting in 15 different filter responses, or channels, c_1 to c_{15}. The sum over all c_i almost completely reconstructs the input image.

3.3.2 The Nonlinearity

As we will see in this section, we need to incorporate some nonlinear mechanism in order to reproduce certain aspects of human texture segregation.

Let $g(x, y)$ and $G(u, v)$ be the receptive field of the filter in the spatial and frequency domain, respectively. Furthermore, let $m(x, y)$ and $M(u, v)$ be a texture pattern and its Fourier transform. Because we use zero d.c. filters, $G(0, 0) = 0$. Therefore, the product GM is zero at $(0, 0)$ as well. This means that the average of $g{*}m$ is also zero. Consequently, if we average the filter response over a region large enough, we obtain an average response of zero for any pattern. This can be seen in Figure 22, where two filter responses to the different textures of Figure 16(a) and their corresponding averages are shown.

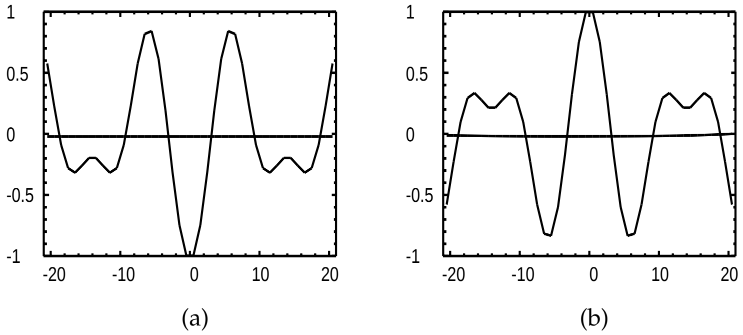


Figure 22: Average response of one of the proposed filters to the two different texture regions of Figure 16(a): **(a)** shows the crossection along the x axis of the response to the left texture area. The straight line at zero corresponds to the mean value taken over a region about the size of the filter's receptive field. **(b)** shows the same information for the right texture area. Note, that the average response is the same for both regions. Therefore, a segregation is not possible.

The average response of the zero d.c. filter is the same for both textures. Therefore, a segregation is not possible. This circumstance provides a motivation to add some sort of nonlinearity to the channels. According to Malik and Perona [MP90] there are at least two physiologically plausible choices for such a nonlinearity:

1. A nonlinear contrast response function, or
2. Lateral interactions within and among different channels

Albrecht and Hamilton [AH82] measured the responses of 247 neurons recorded from the striate cortex of monkeys and cats as a function of the contrast intensity of luminance-modulated sine-wave gratings. They describe a qualitative contrast response function that applied to the majority (some 80–90%) of the measured cells:

> *In general then, as the contrast of a grating increases, the response of a striate cell increases in a relatively linear fashion. [...] The slope covers a restricted contrast range (generally less than 1 log unit of contrast). At approximately 50–60% of the maximum response of a cell, the function begins an accelerating compression. Ultimately, the response totally saturates.*

In order to mimique this behaviour, we introduce a sigmodial contrast transfer function (CTF), which operates on each channel. The actual function we use is a hyperbolic tangent as shown in Figure 23.

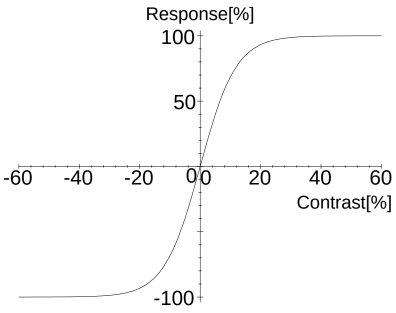


Figure 23: Contrast response function: A hyperbolic tangent is used to mimique the contrast response function of simple cells. For high contrast this function causes a saturation effect. In the first 10% of the contrast range, the response increases linearly, then it quickly approaches the saturation level.

If we now apply this contrast transfer function to each channel, we avoid the problem that the filter responses become zero if spatially pooled over a larger region. The same texture pattern as above is used to demonstrate the effects of the nonlinear CTF in Figure 24.

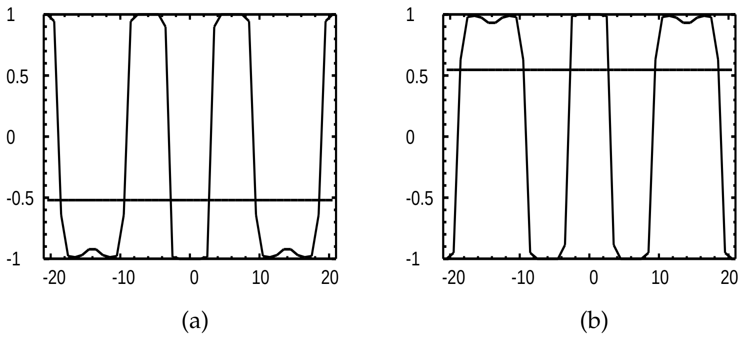


Figure 24: Effects of the nonlinear CTF: Comparing (a) and (b) with the responses shown in Figure 22, we see that the nonlinear stretching of the curves has changed the average values of the signals which can now be used to discriminate between the two patterns. Note, that the xy-mirror symmetry of the patterns causes average values of opposite sign. A y-ms texture pattern would still not segregate due to the reasons mentioned in section 3.3.1.

The usage of the CTF is also motivated by empirical evidence that an early nonlinearity is necessary in order to explain human texture perception [GBS92, LB91]. A similar contrast response function is also used in the models of Caelli [Cae88] and Jain & Farrokhina [JF91].

3.3.3 From Gabor Responses to Texture Features

After the application of the CTF we arrive at a set of 15 nonlinear transformed response images. The question is now, how to extract a meaningful texture description from this set. Manjunath and Ma [MM96c] propose a method based upon the statistical properties of the response signals. For each channel they compute the unnormalized mean and standard deviation over the whole image:

$$\mu_{mn} = \iint |c_{mn}(x,y)| \, dx dy \tag{28}$$

$$\sigma_{mn} = \sqrt{\iint \left(|c_{mn}(x,y)| - \mu_{mn}\right)^2 dx dy}, \tag{29}$$

where c_{mn} is the response in channel mn corresponding to scale m and orientation n. Taking all channels together, they arrive at a $2KS$-dimensional feature vector

$$\bar{\mathbf{h}} = (\mu_{11}, \sigma_{11}, \mu_{12}, \sigma_{12}, \ldots, \mu_{SK}, \sigma_{SK})^T, \tag{30}$$

where S and K are the total number of scales and orientations, respectively. They use this feature vector to compute the global difference in the textual appearance of *whole* images. In contrast to this, our aim is to detect local textual differences *within* a single image. A common approach to establish this is to divide the image into a set of small overlapping rectangular blocks centered on a regular grid. Hofmann et al [HPB96], for example, used 32×32 sized blocks on a 64×64 grid for 512×512 images. Then for each block the texture feature vector is computed and associated with the corresponding grid position.

The more grid positions we use, the more accurate is the localization of texture borders. Maximum accuracy is achieved if the grid resolution is equal to the resolution of the digital image in pixels. Note, that in this case the computation of the mean value over a small block of size $M \times N$ is equivalent to a convolution of the image with an $M \times N$ filter kernel with entries $\frac{1}{MN}$. Such a convolution in turn corresponds to a smoothing of the image data. In digital image processing it is well known, that the convolution with rectangular filter masks only leads to suboptimal smoothing results [Jäh93, Ver91]. Much better results are obtained with Gaussian kernels. Therefore, we arrive at the following texture features:

$$\mu_{mn}(x,y) = c_{mn}^{\mathrm{CTF}}(x,y) * gs_{mn}(x,y) \tag{31}$$

$$\sigma_{mn}(x,y) = \sqrt{\left(c_{mn}^{\mathrm{CTF}}(x,y) - \mu_{mn}(x,y)\right)^2 * gs_{mn}(x,y)}, \tag{32}$$

where $*$ denotes the convolution operation, c_{mn}^{CTF} is the response in channel mn after the nonlinear scaling with the contrast transfer function, and gs_{mn} is the corresponding Gaussian filter kernel given by

$$gs_{mn}(x,y) = e^{-\frac{x^2+y^2}{\rho_{mn}}}, \tag{33}$$

where ρ_{mn} is the width of the smoothing filter.

Note, that in contrast to (28) and (29) we do not use the absolute value of the channels response. The reason for this is clear, if we reconsider Figure 24: The two patterns only segregate due to their different signs. By taking the absolute value of the response, this difference would be lost, and thus the textures would not segregate.

A critical choice is the filter width ρ_{mn}. Since texture is a quality which cannot be associated with a single point, but a certain region of the image, a more reliable measurement of texture features calls for large sizes. On the other hand, an accurate localization of texture borders demands smaller sizes. In our experiments we found a heuristical value of three times the size of the receptive field's Gaussian envelope to be a good choice. The relation of the sizes is shown in Figure 25.

Summing everything up, we arrive at the following stages for the feature extraction process:

1. Compute the set of 2D Gabor filters tuned to 5 different orientations and 3 scales according to the daisy-like pattern scheme described by (26) and (27).

2. For each element of the filter bank, compute its response image, or channel, c_{mn} to the given input.

3. Apply the nonlinear contrast transfer function to each channel c_{mn}, yielding c_{mn}^{CTF}.

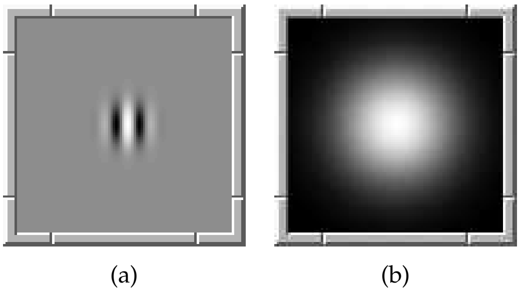


Figure 25: Size of Gaussian smoothing filter in relation to its corresponding receptive field: **(a)** shows the channel's receptive field in the spatial domain, **(b)** shows the Gaussian filter which is used to smooth the response in that channel. The width of the smoothing Gaussian is three times the size of the receptive field's Gaussian envelope.

4. Compute the texture features μ_{mn} and σ_{mn} for each of the 15 channels according to (31) and (32).

Therefore, we get 30 images which summarize the relevant properties of the textual appearance of the input. Hence, for each position (x, y) in the input image, we get a 30-dimensional feature vector $\mathbf{h}(x, y)$ describing the local texture at that point:

$$\mathbf{h}(x,y) = (\mu_{11}(x,y), \ldots, \mu_{53}(x,y), \sigma_{11}(x,y), \ldots, \sigma_{53}(x,y))^T \tag{34}$$

To give an idea of how the feature images actually look like, an example is shown in Figure 26.

3.4 Grouping of Texture Features with the CLM

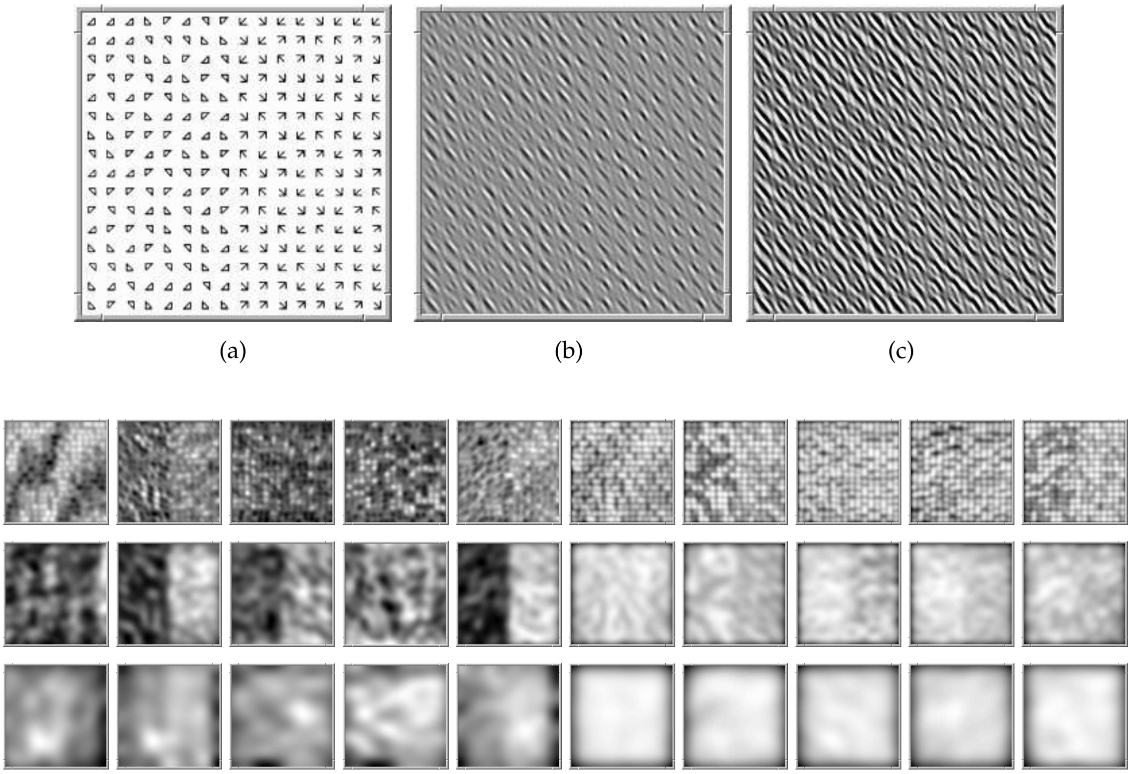
So far we have motivated the feature extraction process which generates the set of feature vectors $\mathbf{h}(x, y)$ with $M_x \times M_y$ elements, where M_x and M_y denote the width and height of the input image, respectively. For a description of the feature extraction in context to perceptual grouping with the CLM, reconsider section 2.1.1.

In order to apply the CLM, we need to find a suitable pairwise interaction function $f_{rr'}$ as described in section 2.1.2. Since $f_{rr'}$ expresses the similarity of two stimuli we can think of $f_{rr'}$ as a kind of distance metric defined on $\mathcal{M}$, the set of feature vectors. As we will see in the next section, it is a nontrivial problem to find a good distance measure on a high dimensional space.

3.4.1 The Curse of Dimensionality

The term "curse of dimensionality" was first introduced in 1961 by Bellman who used it to describe the complexity of computations where the number of computations exceeds the available computing power [HH96]. Recently, it is also used in context with classification tasks and refers to the difficulties associated with the exponential growth of hypervolume as a function of dimensionality. In a high dimensional space, data samples quickly become "lost" in the wealth of space.

In our case this problem is caused by the 30-dimensional feature vector we use to describe the textual appearance in an image. As can be seen in Figure 26 just a few features contain the necessary information to segment the different textures. Ideally, only these vector components should be incorporated into the interaction function $f_{rr'}$. This cannot be done, because it is not clear a priori which features have a high discriminatory power for a given input image. On the other hand, if we take all vector



(a) (b) (c)

(d)

Figure 26: Example for the set of feature images: **(a)** shows the input image consisting of different micropatterns, **(b)** illustrates channel c_{22}, which is the response of a filter tuned to a diagonal orientation from the upper left to the lower right (corresponding to B2 in Fig. 19), **(c)** shows the response after transformation with the nonlinear CTF. In **(d)** the feature images μ_{mn} and σ_{mn} for $m = 1, \ldots, 3$, $n = 1, \ldots, 5$ are shown. The organization is as follows: The three rows correspond to the three scales with the first row describing the smallest scale. In each row, the first five images display μ_{mn}, the last five σ_{mn}. Note, that mainly μ_{25} contributes to the segregation of the two different textures. For visualization issues each image was scaled to the interval [0,255].

elements into account for the calculation of the distance measure, then for each element with low discriminatory power, noisy information is added. This can severely reduce the segmentation performance – see also Pichler et al [PTH96] for a discussion on this topic. Therefore, we seek for a representation of the feature vectors in a lower dimensional space without loosing much of the information contained in the data.

Principle Component Analysis

Principal Component Analysis (PCA) is a statistical technique for calculating the directions of maximal variance of a set of data points in some high dimensional space. Consider a set of n-dimensional data samples $\{\mathbf{x}_r\}_{r=1,\ldots,N} \subset \mathbb{R}^n$, where N is the number of samples. The covariance matrix of the data set is given by

$$\mathsf{C} = \frac{1}{N} \sum_{r=1}^{N} (\mathbf{x}_r - \bar{\mathbf{x}})(\mathbf{x}_r - \bar{\mathbf{x}})^T, \tag{35}$$

where

$$\bar{\mathbf{x}} = \frac{1}{N} \sum_{r=1}^{N} \mathbf{x}_r \tag{36}$$

is the mean of the data vectors. The elements C_{ij} describe the covariance of the vector components x_i and x_j. Since C is a symmetric $N \times N$ matrix, it may be written in the form

$$\mathsf{C} = \mathsf{GLG}^T, \tag{37}$$

where L is the diagonal matrix of eigenvalues λ_i of C, and G is an orthogonal matrix whose columns are the normalized eigenvectors $\mathbf{v}_i$ of C with length $||\mathbf{v}_i|| = 1$; $\mathbf{v}_i$ is then called the ith *principal component* of the data sample.

The principal components $\{\mathbf{v}_i\}_{i=1,\ldots,n}$ form an orthogonal basis set of the n-dimensional vector space. Furthermore, the principal component $\mathbf{v}_1$ corresponding to the largest eigenvalue λ_1 is the direction of greatest variance, $\mathbf{v}_2$ is the direction of second greatest variance, and so on – see also Figure 27.

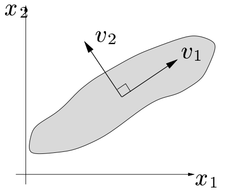


Figure 27: Illustration of the principal components of a 2-dimensional data distribution: The elongated blob denotes a set of 2D data points in the x_1x_2 space. The vectors v_1 and v_2 correspond to the direction of greatest and smallest variance of the data set, respectively. They are called the principal components.

Since the $\mathbf{v}_i$ form an orthogonal basis set, we can represent the data sample in terms of this new basis as

$$\mathbf{w}_r = \mathsf{G}^T\mathbf{x}_r, \tag{38}$$

The transformation given by (38) is commonly referred to as the *Karhunen-Loeve Transformation* (KHL). The components of the $\mathbf{w}_r$ are uncorrelated. The idea is now to summarize most of the variability of the data using only the principal components with the highest variances, and therefore reducing the dimensionality.

The "proportion of total variation" [MKB79] explained by the first k principal components is given by $(\lambda_1 + \cdots + \lambda_k)/(\lambda_1 + \cdots + \lambda_n)$. Therefore, if we reconsider the example of the feature vectors shown in Figure 26, we obtain from the PCA, that the first 4 principal components of $\{\mathbf{h}(x, y)\}$ contribute to 75% of the overall variation in the data – see also Figure 28.

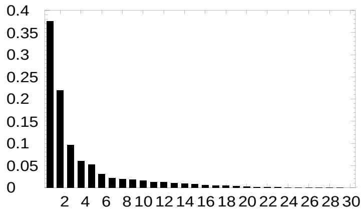
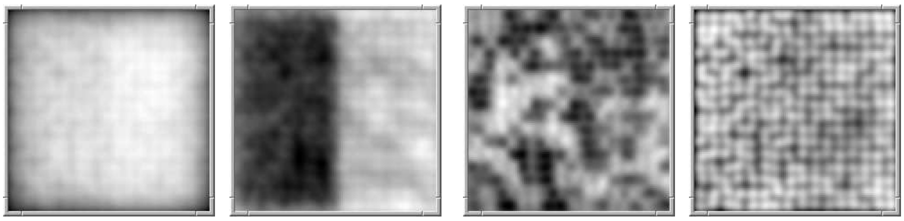


Figure 28: Eigenvalues corresponding to the principal components of the 30-dimensional feature space shown in Figure 26. The first four principal components contribute to 75% of the overall variation in the data. Note, that the bars were scaled, so that their total height is equal to one.

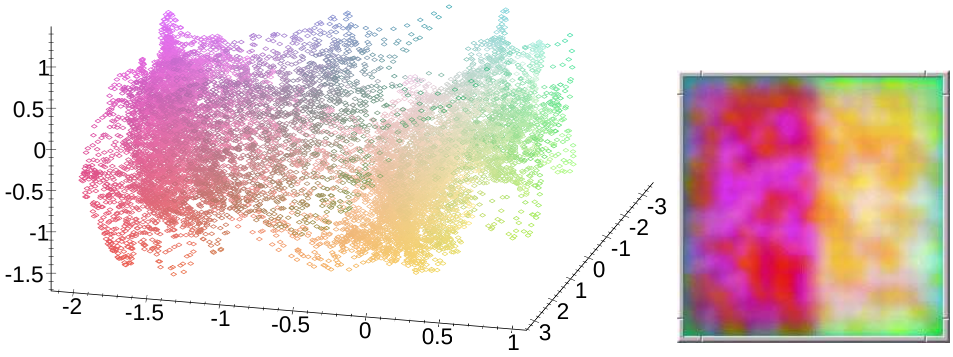
We get additional information about the relevance of these four components, if we take a look at the feature vectors after the Karhunen Loeve Transformation: As shown in Figure 26 the set of the 30-dimensional feature vectors $\mathbf{h}$ correspond to 30 feature images. After the transformation of $\mathbf{h}$ according to (38) we obtain a new set of 30 images, which now describe the set of feature vectors in terms of their principal components. The first 4 of these images are shown in Figure 29.



(a):1st (b):2nd (c):3rd (d):4th

Figure 29: Orthogonal projection of the feature vectors: **(a)** corresponds to the projection of $\mathbf{h}(x, y)$ as defined in (34) onto its first principal component. It can be seen, that this linear-combination of feature vector components separates the outermost border region from the rest of the texture. This is due to the fact, that the texture elements at the border do not have any neighbours, and therefore produce a different local texture description. In **(b)** it can be nicely seen, that the feature set contains a direction, where the two different textures segregate very well. In **(c)** noisy information predominates. The visible structure in **(d)** corresponds to the micropatterns themselves: The small bright speckles indicate the position of a single texture element.

To gain a better intuition of the topological structure of the transformed data set, we plot the first 3 components in $\mathbb{R}^3$ as shown in Figure 30. Though two components contain noisy information, a clear structure is visible and allows a sensible texture segmentation.



(a) (b)

Figure 30: **(a)**: 3D-plot of the first 3 components of the data set after the KHL transformation. The colour of the points is chosen according to their position in the transformed space: The red channel corresponds to the first principal component and goes from the back to the front, the green channel corresponds to the second p.c. and goes from the left the right, and the blue channel corresponds to the third p.c. and goes from the bottom to the top. In **(b)** the colour of the points is mapped back into the image space: Each pixel at position (x, y) is coloured according to the colour of its feature vector in **(a)**.

Another interesting aspect of the PCA is that it provides us with a method to see which linear combination of 2D Gabor filters produces a good segmentation of two textures. The projection onto the second p.c. as shown in Figure 29(b) is a linear combination of 30 features corresponding to the mean and variance of the response of 15 2D Gabor filters, respectively. If we now map the coefficients of this linear combination onto the daisy-like pattern of the filter arrangement, we get two images indicating which features contributed mostly to the segregation of the texture pattern: One for the features that were obtained from the mean responses, the other for the features obtained from the responses' variance. As can be seen in Figure 31, the linear combination of μ_{22}, μ_{23}, and μ_{25} with an emphasize on μ_{25} is responsible for the segregation of Figure 26(a).

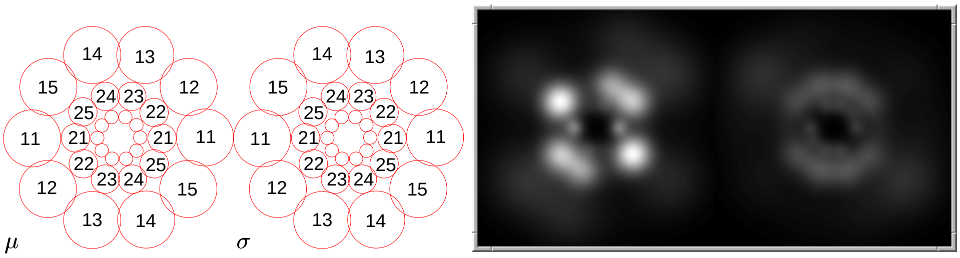


Figure 31: Linear combination of 2D Gabor filters in the frequency domain: The left and right part of the image show the arrangement of filters whose mean responses and variances contributed to Figure 29(b), respectively. It can be seen, that μ_{25} is the most important feature for the segregation of the texture in Figure 26(a).

So, the PCA has proven to be a valuable technique in order to reduce the high dimensional data set. Prior to the KHL-transformation only 1 component of 30 others contained a high discriminatory information, afterwards, this relation was reduced to 1:4. Furthermore, the PCA supplies us with a method to visualize the high dimensional data and to get an intuitive understanding of its organization. The profit of the PCA is not restricted to the discussed example of Figure 26, but also applies to all other investigated examples as shown in sections 3.5 –3.6.

Resolution Reduction

As we have seen, the PCA provides a method to significantly reduce the dimensionality of the feature space we use to describe the local textual appearance of an image. Therefore, for a 256×256 sized digital image, we arrive at a set of 65536 4-dimensional feature vectors $\mathbf{p}_r = (\mathbf{v}_1 \cdot \mathbf{h}(x,y), \mathbf{v}_2 \cdot \mathbf{h}(x,y), \mathbf{v}_3 \cdot \mathbf{h}(x,y), \mathbf{v}_4 \cdot \mathbf{h}(x,y))^T$, where r corresponds to the position (x,y), $\mathbf{h}(x,y)$ is the feature vector defined by (34), and $\mathbf{v}_i$ is the ith principal component of the set $\{\mathbf{h}(x,y)|x = 1,\dots,M_x, y = 1,\dots,M_y\}$, where M_x and M_y are the width and height of the input image, respectively.

The computational complexity of the simulation of the CLM's dynamic is of the order $\mathcal{O}(N^2)$, where N is the number of feature vectors. Since the grouping of 180 stimuli took approximately 3 seconds, the grouping of 65536 stimuli would take $65536^2 \cdot 3/180^2$ sec ≈ 4.6 days. Therefore, we need to reduce the number of feature vectors in order to attain a practicable utilization of the CLM for textual grouping. We achieve a drastical reduction of the set of feature vectors by the following subsampling technique:

We can interpretate $\{\mathbf{p}_r\}_{r=1,\dots,N}$ as a set of 4 images, where the first component of $\mathbf{p}_1$ and $\mathbf{p}_N$ denotes the upper left and lower right pixel of Figure 29(a), respectively. The second components of the $\mathbf{p}_r$ correspond to Figure 29(b) and so on – see also Figure 32

Each of these images is then subsampled by a factor 8 according to the scheme depicted in Figure 32. Hence, we arrive at a new set of 4 images with a resolution of $256/8 \times 256/8 = 32 \times 32 = 1024$ pixels each. From this stack of reduced images we extract the set of feature vectors $\{\mathbf{m}_r\}_{r=1,\ldots,1024}$ which is then used as the input for the CLM. Analogous to Figure 30 the subsampled space is plotted in Figure 33.

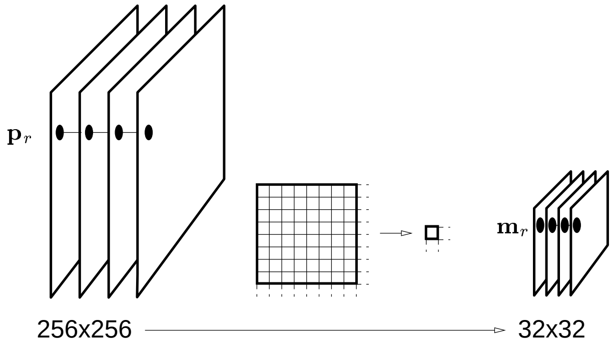
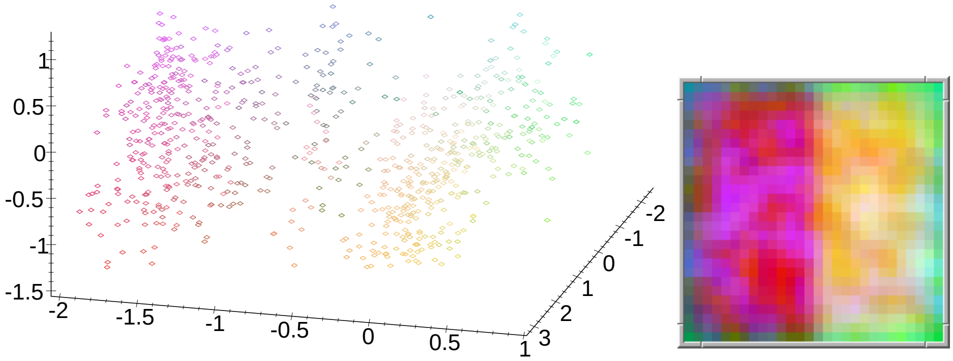


Figure 32: The process of the feature vector subsampling: The stack of large images shown on the left hand side correspond to the set of feature vectors $\mathbf{p}$ obtained from the PCA. The subsampling is done in the following way: The 256×256 images are divided into 1024 8×8 sized regions. For each of these regions the mean value is computed and is assigned to the corresponding pixel of the smaller subsampled image.



(a)

(b)

Figure 33: 3D-plot of the subsampled feature data: This figure shows the same information as Figure 30 for the subsampled feature data. Note, that the structure of the cloud is completely preserved, while the density was drastically reduced. In **(b)** it can be seen, that the image space looks somewhat pixelized due to the subsampling of the data.

3.4.2 The Interaction Function

We are now in the position to construct the interaction function for the extracted feature vectors $\mathbf{m}_r$. We will use a distance measure based upon the one Manjunath and Ma propose for an application, which we will briefly summarize below. For details see [MM96c, MM96a].

The Measure for Texture Similarity

Manjunath and Ma present a system which is able to retrieve patterns from a database containing 1856 textured images each of the size 128×128 pixels. The idea is to submit a query pattern and ask the system to retrieve all images from the database which "look similar". They propose a distance measure based upon the global texture features defined by (28) and (29). The distance between two image patterns i and j is then given by

$$d(i,j) = \sum_m \sum_n d_{mn}(i,j), \tag{39}$$

where

$$d_{mn}(i,j) = \left| \frac{\mu_{mn}^{(i)} - \mu_{mn}^{(j)}}{\alpha(\mu_{mn})} \right| + \left| \frac{\sigma_{mn}^{(i)} - \sigma_{mn}^{(j)}}{\alpha(\sigma_{mn})} \right|, \tag{40}$$

$\alpha(\mu_{mn})$ and $\alpha(\sigma_{mn})$ are the standard deviations of the respective features over the entire database, and are used to normalize the individual vector components.

They compare this distance measure based on Gabor filter image representation and demonstrate that the proposed method outperforms several other multiscale texture features. Therefore, it has proven to be able to express the textual similarity of different images very well. For this reason we will adopt this type of metric for the usage with the CLM to evaluate the similarity between two feature vectors $\mathbf{m}_r$ and $\mathbf{m}_{r'}$. We generalize the proposed metric to an arbitrary *Minkowski* norm of the following type:

$$d_{\text{text}}(r,r') = \sqrt[n]{\sum_{i=1}^{4} \left(\frac{|m_r^i - m_{r'}^i|}{\sqrt{\alpha(m^i)}} \right)^n}, \tag{41}$$

where m_r^i is the ith component of vector $\mathbf{m}_r$, $\alpha(m^i)$ is the standard deviation of the ith component of all 1024 $\mathbf{m}_r$, and n is the dimension parameter of the Minkowski norm. For $n = 1$ it resembles the so-called *Cityblock* metric, for $n = 2$ it is equivalent to the Euclidean distance, and for $n = \infty$ it is called the maximum norm. If the vectors $\mathbf{m}_r$ and $\mathbf{m}_{r'}$ differ in only one component, then (41) results in the same distance for all n. On the other hand, if the vectors differ in more than one component, then the Cityblock metric with $n = 1$ yields the largest distance.

During our simulations we observed, that for a single grouping problem the quality of the segmentation does not depend on n. The parameters of the interaction function as described later in section 3.4.2 can always be tuned in such a way, that a sensible grouping occurs. However, we found that the usage of the Cityblock metric made it easier to control the set of parameters in such a way, that it can be kept constant for a large class of input images. Therefore, we choose $n = 1$ in (41). This is also consistent with the observations of Hofmann et al [HPB96] who state, that "small n often yields superior results".

Note, that in contrast to Manjunath and Ma we normalize each feature with the square root of its standard deviation. This results in normalization factors closer to 1. We found that this choice leads to a better segmentation performance especially for textures consisting of micropatterns.

Incorporation of the Gestalt Principle "Proximity"

As we have seen in section 1, one of the most basic Gestalt principles is the Law of Proximity. This principle is also relevant in context with texture segregation: Because texture is an attribute which always stretches over a certain region, neighbouring pixels are very likely to belong to the same texture category. If we neglect this information, then noise in the feature vectors can easily lead to speckle-like noise in the segmented images. Therefore, we also have to take the information about the spatial location of the texture features into consideration. In the literature there are several ways employed to establish the principle of proximity.

Hofmann et al use so-called *Objective Functions* which describe the relation of two texture regions. One element in this set of functions is a cost function $\mathcal{H}^{\text{top}}$ which introduces "topological costs". These are based on the four-connected neighbourhood of image blocks and "glue" these blocks together. For details see [HPB96]. They furthermore suggest a post-processing of the labelled images. That is, after an initial grouping of the feature vectors, a smoothing operation like a median filter enforces the spatial constraints and eliminates the noise.

Jain and Farrokhnia [JF91] propose an incorporation of the spatial adjacency information directly into the clustering process. They achieve this by including the spatial coordinates of the pixels as two additional features.

Wersing [WSR97] has shown that the principle of proximity can be expressed by an interaction function between two feature vectors. We therefore adopt this kind of interaction and include it as an integral component of the new interaction function as shown below.

Construction of the Interaction Function

So far we have motivated a distance measure on the set of feature vectors $\mathbf{m}_r$ which measures the textual similarity of two image regions. Furthermore we reasoned, that we need to incorporate the principle of proximity. We therefore propose an interaction function composed of two parts: The first part corresponds to the Gestalt principle of Similarity and the second to the principle of Proximity:

$$f_{rr'} = e^{-\frac{d^2_{\text{text}}(r,r')}{R^2_{\text{sim}}}} + c_{\text{prox}}\, e^{-\frac{|\mathbf{x}_r - \mathbf{x}_{r'}|^2}{R^2_{\text{prox}}}} - k, \tag{42}$$

where d_{text} is the distance measure defined in (41) with $n = 1$, c_{prox} is a coupling constant between the two Gestalt principles, R_{sim} is the distance, for which two image regions are said to be similar, $\mathbf{x}_r$ is the position from where in the input image $\mathbf{m}_r$ was taken, R_{prox} is the radius which controls the proximity grouping, and k is the inhibitory constant.

The summation of the two components corresponds to a logical *or* which combines the two principles: $f_{rr'}$ is positive if the two image regions corresponding to r and r' have a similar textual appearance *or* if they are close together. The exact grouping behaviour is controlled by the set of 4 parameters:

1. R_{sim} corresponds to the range of the interaction in terms of textual similarity. If R_{sim} is too large, then even for large distances d_{text} the first part of the interaction function is close to 1. Thus, the CLM cannot differentiate between different texture regions, resulting in a single perceptual group for the whole image. On the other hand, if R_{sim} is too small, then even for similar textures, the first part becomes zero. The grouping behaviour then depends on the parameters controlling the proximity interaction.

2. c_{prox} controls the coupling between the two parts of the interaction function. The larger it is chosen, the more the Gestalt principle of Proximity will dominate.

3. R_{prox} corresponds to the range of the proximity interaction. Here, the same notes as for R_{sim} are holding: If it is too large, then the whole image will be grouped into one perceptual group. If it is too small and c_{prox} is large, than the image will be splitted up into many small subgroups.

4. k is the global inhibitory constant. It has to be choosen in such a way, that $f_{rr'}$ is neither completely positive nor negative ([WSR97]).

During our simulations we found that the following heuristically chosen parameters give a good overall grouping performance:

1. $R_{\text{sim}} = 6.6$. This rather random appearing number has no intuitive meaning. It is related to the extracted feature vectors and the distance measure d_{text}.

2. $c_{\text{prox}} = 0.6$ is choosen because a larger value causes a domination of the Proximity principle and a split up of large regions containing the same texture. Since k is choosen $k = 0.5$, the interaction is still positive for regions very close together even if they do not appear similar. In this way, small local differences in texture are levelled.

3. $R_{\text{prox}} = 0.63$. The vectors describing the spatial location of the features are scaled in such way, that x- and y-coordinate are always in the interval $[0, 1]$. Therefore, the proposed value $R_{\text{prox}} = 0.63$ causes a proximity interaction as shown below in Figure 34(b).

4. $k = 0.5$. Together with $c_{\text{prox}} = 0.6$ this results in values of $f_{rr'} \in [-0.5, 1.1]$

To get an idea of how the two components of the interaction function are working together, its assembly is depicted below in Figure 34. The effect of the incorporation of the Law of Proximity into the feature interaction function can be seen in Figure 35.

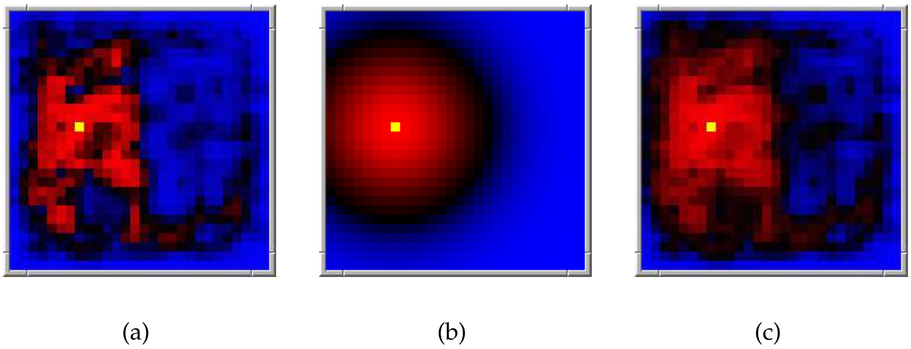


Figure 34: Composition of the feature interaction function: **(a)** shows the part of $f_{rr'}$ based on the texture distance measure as defined in (41): The texture distance of the region denoted by the yellow point to all other positions in the image is plotted. Red and blue code a positive and negative interaction, respectively. The range of the pure texture interaction is from -0.25 to 0.75. **(b)** shows the topological component only. Due to the coupling constant of $c_{\text{top}} = 0.6$ the values range from -0.25 to 0.35. In **(c)** the sum over both components is depicted, resulting in a range of -0.5 to 1.1.

3.4.3 Obtaining Perceptually Grouped Images

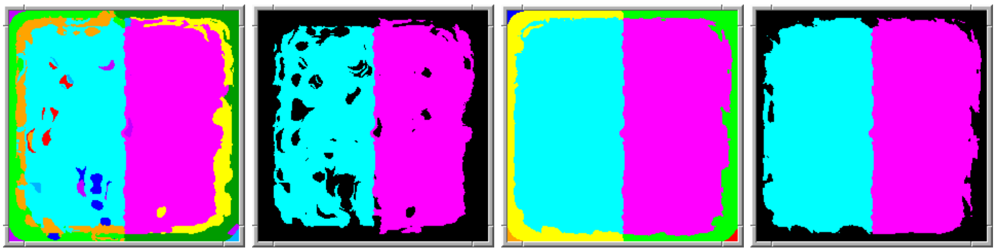
Interpreting the CLM's Output

Once the dynamic of the CLM has reached an equilibrium point, for each feature vector we get the number of its active layer as an output from the CLM. Since we know the position, from where in the image the feature vector $\mathbf{m}_r$ was taken from, we can label the corresponding points in the image space with this output. By coding each label with a different colour, we obtain an image, where different colours denote different perceptual groups.

According to Wersing [WSR97] for an equilibrium point of the dynamics either holds

$$x_{r\alpha} = h_r + \frac{1}{J_1} \sum_{r'} f_{rr'} x_{r'\alpha} \quad \text{or} \quad x_{r\alpha} = 0.$$

Therefore, $0 < x_{r\alpha} < h_r$ means, that there is a significant amount of activities in the layer α which belong to dissimilar feature vectors, because in that case the sum over all r' is negative. Consequently, only layers with activities greater than h_r correspond to clear and distinct perceptual groups. This motivates a heuristical threshold of $1.2h_r$ we found in our experiments. It proved to be a good value to discriminate between significant groupings and layers containing noisy and ambiguous information. An example for the thresholding effect and the benefits of the Proximity principle are shown in Figure 35.



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

Figure 35: Effects of Proximity principle and Thresholding: We used a CLM with 10 layers for the perceptual grouping of the image shown in Figure 26(a) on page 26. The image regions are coloured according to the scheme described in the text above. In **(a)** the principle of Proximity is not incorporated into the interaction function, and no threshold is applied to the CLM's output. It can be seen that the left hand side of the grouped image suffers from speckle-like noise. This is due to the fact, that the corresponding region in the input image appears less homogeneous than the right hand side. In **(b)** only those regions are labelled, where the corresponding activities are greater than the proposed threshold of $1.2h_r$. The erroneous classifications are removed, but now holes are disturbing the output. In **(c)** the benefits of the Gestalt principle of Proximity can be seen: The noise is removed and only 7 of the 10 layers are actually used. In **(d)** the threshold removes the outermost region of the image, where border effects lead to different feature vectors. Only the two distinct texture areas remain. Note, that all images in this example were obtained with the resolution enhancement technique described below.

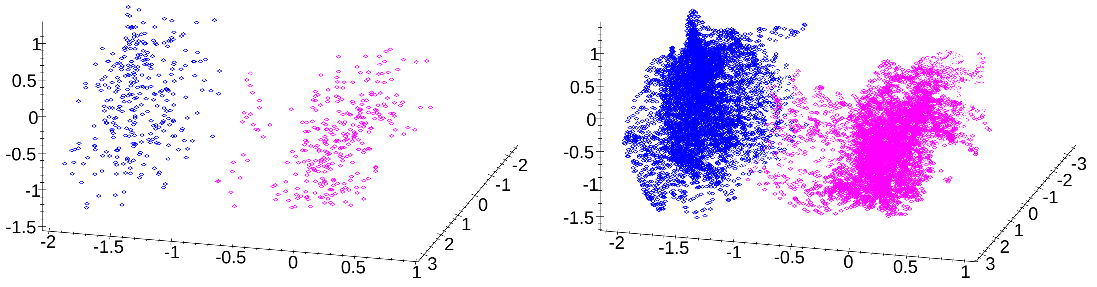
Resolution Enhancement

Due to the resolution reduction described above we have lost information about the precise localization of the feature vectors. Consequently, texture boundaries in the input image of 256×256 pixels can only be detected with an accuracy allowed by the subsampled 32×32 sized images. Therefore, we propose a method to improve this rather small localization accuracy of 8 pixels.

The CLM only labelled the subsampled data set of the 1024 feature vectors $\mathbf{m}_r$ as shown in Figure 36(a). What we actually seek is a grouping of all the 65536 feature vectors $\mathbf{p}_i$. Since the subsampling preserved the structure of the data set, we can also label the 65536 $\mathbf{p}_i$ in the following way: For each $\mathbf{p}_i$ its nearest neighbour $\mathbf{m}_{\text{near}}$ is found such that

$$|\mathbf{p}_i - \mathbf{m}_{\text{near}}| + |\mathbf{x}_i - \mathbf{x}_{\text{near}}| = \min, \tag{43}$$

where $\mathbf{x}_i$ and $\mathbf{x}_{\text{near}}$ are the normalized positions from where in the images $\mathbf{p}_i$ and $\mathbf{m}_{\text{near}}$ were taken, respectively. I.e. the lower left position in the images corresponds to $\mathbf{x} = (0,0)$ and the upper right to $\mathbf{x} = (1,1)$. Each $\mathbf{p}_i$ is then labelled according to the label of its nearest neighbour $\mathbf{m}_{\text{near}}$. The distance metric in (43) is motivated as follows: By using only the distance $|\mathbf{p}_i - \mathbf{m}_{\text{near}}|$ we would introduce errors in those regions of the data distribution, where different labelled clouds are meeting. For this case the additional information about the spatial position of the feature vectors allows a more accurate labelling. The usage of this technique allows the location of texture borders with the same high resolution as of the input images, as can be seen in Figure 49 on page 43.



(a) (b)

Figure 36: Grouping of 1024 feature vectors $\mathbf{m}_r$ and labelling of 65536 original feature set $\mathbf{p}_i$: In **(a)** each feature vector is coloured according to its active layer in the CLM, **(b)** shows the distribution of the feature vectors extracted from the high resolution input image and their labelling according to the proposed nearest neighbour algorithm.

3.5 Application to Textures

So far we have motivated the feature extraction process, the construction of the interaction function for the CLM, and a method to obtain perceptually grouped images with the same high resolution as of the input images. Taking all these stages together we arrive at an artificial model of visual perception which describes the organization of visual stimuli according to the two Gestalt principles of Similarity and Proximity.

In this section we present the set of images the model is applied to. It can be divided into three main parts:

1. Artificial textures are created using different types of micropatterns. Following this way, we are able to construct well defined test images and to "get a feeling" about the models behaviour.

2. For the testing of real world textures we use a database of textured images containing samples from the popular Brodatz album [Bro66]. Since many texture segmentation algorithms are tested with these images, we are able to compare the models performance with other texture segmentation models.

3. In section 1 we presented a few images which the Gestaltists used to illustrate their proposed Gestalt laws. We also apply the model to this type of images to see whether the results are consistent with their predictions.

Note, that *all* results presented in this section were obtained using the same set of parameters as described in section 3.4.2. In certain cases we additionally apply the model with other parameters to demonstrate their effects on the grouping result. In all of these cases the change of parameters is explicitly described. Furthermore, if not otherwise noted, the CLM is applied with a constant number of 10 layers.

3.5.1 Artificial Textures

Mirror Symmetric Micropatterns

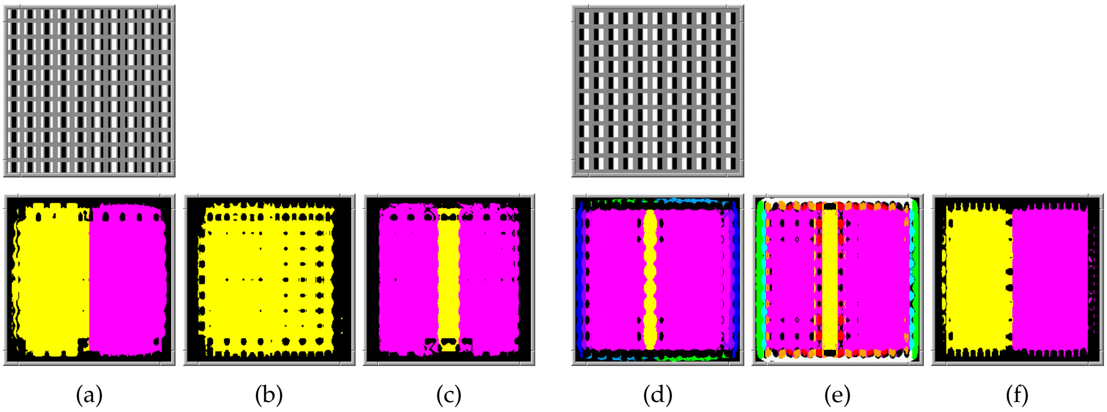


Figure 37: Grouping of textures consisting of mirror symmetric micropatterns: **(a)** and **(d)** show the grouping result of our model using even symmetric 2D Gabor filters with an applied nonlinearity. The corresponding input images are shown in the row above. The achieved grouping behaviour is consistent with human perception: In **(a)** the two regions segregate and in **(d)** the border between the two textures "pops out" whereas the regions themselves do not segregate preattentively. **(b)** and **(e)** show the result if the nonlinearity is omitted. In the first case, the two textures cannot be discriminated any more. In the second case, the texture border still pops out. In **(c)** and **(f)** only odd symmetric mechanisms were used. For a discussion see the text.

First, we show the perceptual organization the model produces if applied to artificial textures consisting of mirror symmetric micropatterns. In section 3.3.1 and 3.3.2 we used these patterns to motivate the usage of pure even symmetric mechanisms and the need for a nonlinearity, respectively. The results shown in Figure 37 are consistent with human perception: The first pattern segregates and the second pattern does not. Note, that the model also produces a "pop-out" effect of the texture border for the second one. The results also show, that the nonlinearity is essential to segment the two different pattern areas in the first image. In contrast, it is not essential for the "pop-out" effect in the second image.

An interesting effect can be observed if we change the type of 2D Gabor filters for the feature extraction. The processing pathway for this experiment was identical as before, but the even symmetric 2D Gabors were exchanged by odd symmetric Gabors with otherwise the same parameters. In this case, the results are reversed: The "pop-out" effect occurs for the first pattern, and the second segregates. Since humans observe the "pop-out" effect only for the second pattern, this might be an evidence, that in human texture perception indeed only even symmetric mechanisms are utilized.

Other Common Micropatterns

In the following we present the grouping results of other artificial textures commonly found in the literature. Figure 38 for example, shows the perceptual organization of the image presented in the introduction to this section.

The image shown in Figure 39(a) is composed of four different regions. Because orientation is an important criterion to detect similarity, the regions with upright T's and L's look similar. Consequently, the untrained observer usually sees only three different regions in this figure.

Interestingly enough, the region with equal orientated X's and L's segregates well to the human observer. To get an idea of how our model segregates the yellow and cyan region in Figure 39(b) we plot the first four principal components of the texture feature vectors analogous to Figure 29. As we see in Figure 40, mainly the differences of the feature vectors along the direction of the 3rd principle component are responsible for the segregation of the two regions. If we inspect the corresponding linear combination of the 2D Gabor filters in the frequency domain, we see, that mainly the component μ_{25} of the feature vectors $\mathbf{h}(x, y)$ as defined in (34) contributes to the segregation.

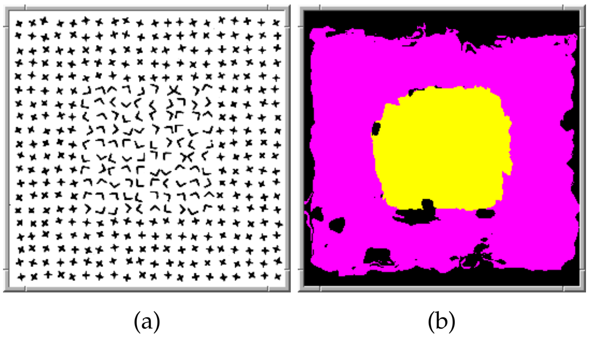


Figure 38: Grouping of two textures consisting of L's and X's: **(a)** is the same image as shown in the introduction of this section on page 12. The grouping result in **(b)** shows that the model is able to differentiate between the two regions of the image. Note, that although the principle of Proximity is incorporated into the feature interaction function, noise leads to small holes in the lower region of the image.

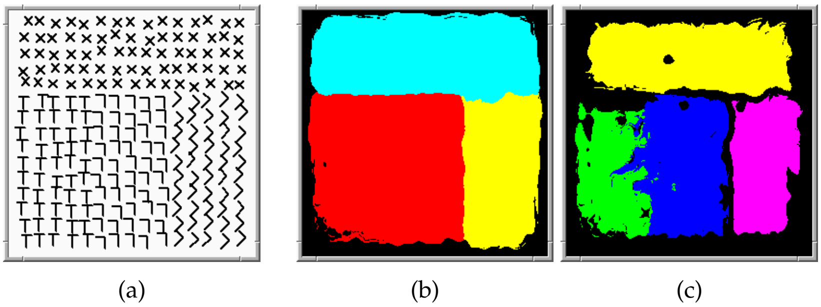


Figure 39: Grouping of textures consisting of L's T's and X's: **(a)** shows the input image consisting of four different regions. It was constructed by placing the patterns on a rectangular grid with a grid size of 16 pixels and a superimposed random jitter with a standard deviation of 1.6 pixels. In **(b)** the output of the CLM is shown. Similar to the untrained human observer the CLM only distinguishes between three different regions. In **(c)** we changed the parameter R_{sim} of (42) to a smaller value, such that difference in texture plays a larger role. In this case the model perceptually organizes the image in four distinct groups.

The corresponding receptive field of the filter from which μ_{25} is extracted is sensitive to gratings orientated from the upper right to the lower left. If we project this receptive field on the input image, as indicated in Figure 41(a), we see that in the region with tilted L's a lot of neurons with this receptive field have zero response. This is not the case for the region constructed of the X's. If we try to position the fields between the X's, there is always some part of the pattern extending into the inhibitory region of the receptive field. Therefore, the average response in the X-region should be lower. This is indeed the case, if we inspect μ_{25} alone, as plotted in Figure 41(b).

Therefore, our model segments the two regions not because of different responses to the micropatterns themselves, but due to different responses to the background generated by the different patterns. One might speculate that this is also the case for the human observer.

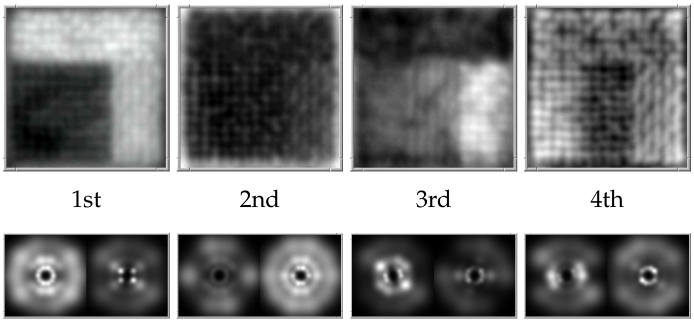


Figure 40: First four principal components of the texture features extracted from Figure 39(a). It can be seen that the 3rd p.c. is responsible for the segregation of the yellow and cyan region in Figure 39(b). The corresponding linear combination of 2D Gabor filters in the frequency domain is shown in the row below. For a description of how these pictures were generated, reconsider section 3.4.1.

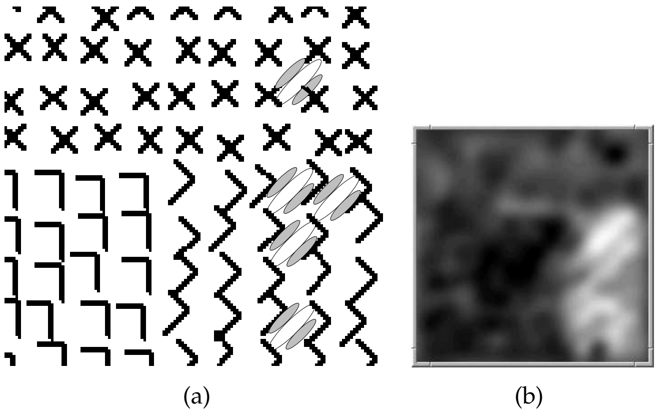


Figure 41: The sketch in **(a)** indicates, that there are a lot of neurons in the tilted L region whose receptive fields generate a zero response. This is not the case for the X region. In **(b)** the average response in the corresponding channel B5 is shown – see Figure 19 for the naming of the channels.

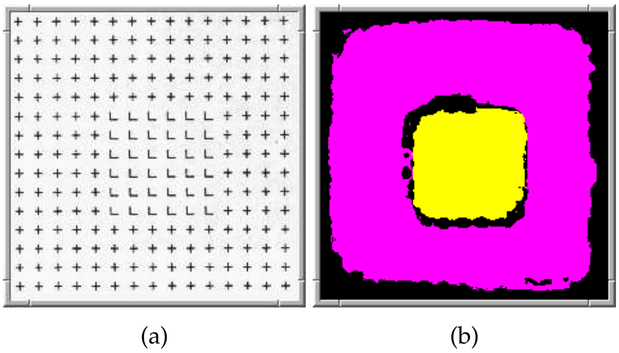
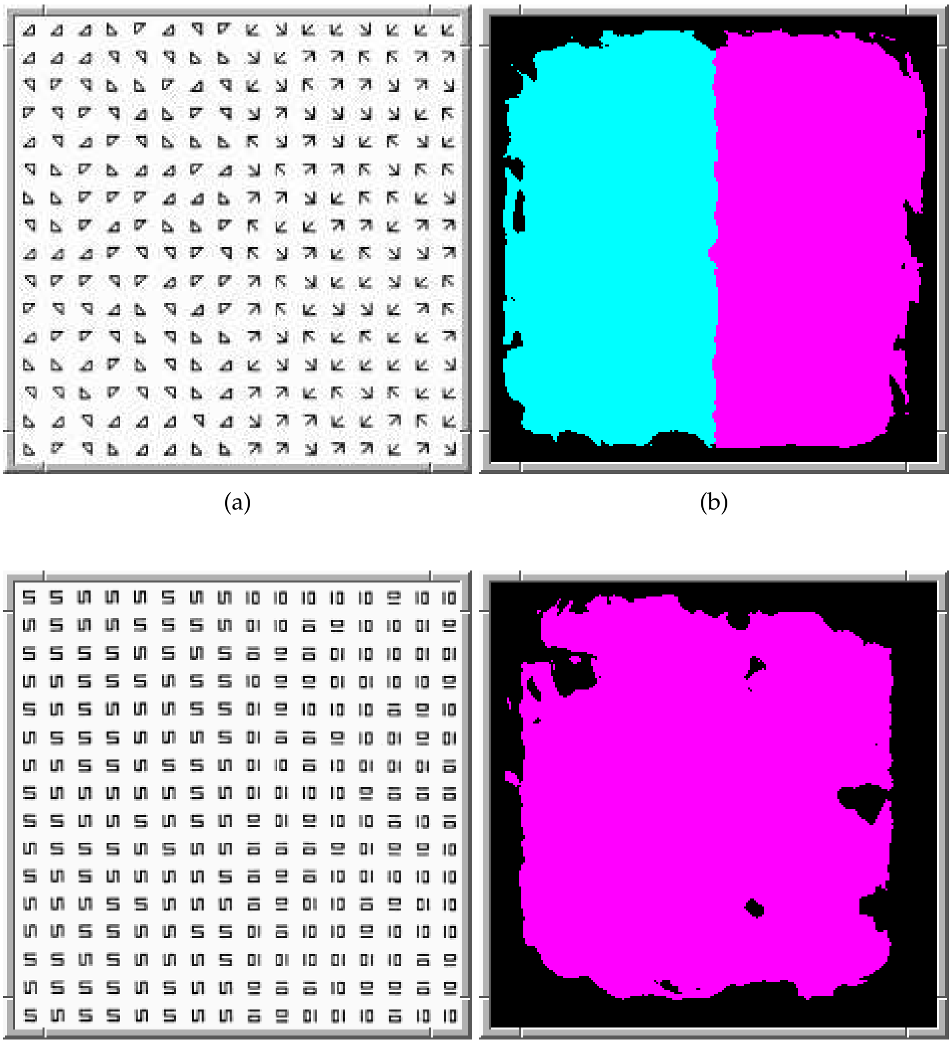


Figure 42: Grouping of texture consisting of L's and +'s. **(a)**: Though the orientation of the micropatterns and the length of their bars are the same for both regions they segregate well. (b): Grouping result according to the texture perception model.

Comparison to Psychophysical Studies

In this section we compare the model's performance if applied to textures used in psychophysical experiments. In these experiments artificial textures are constructed in order to study human texture perception.

Figure 43 shows two textures taken from Julez et al [JGV78]. Both textures have identical second order statistics. The first does segregate preattentively and is therefore a counter-example to the original Julez conjecture that texture pairs with identical second order statistics cannot be discriminated. The second texture does indeed not segregate preattentively.



(a) (b)

(c) (d)

Figure 43: Two textures with identical second order statistics (taken from [JGV78]): Experiments have shown that **(a)** segregates preattentively, **(c)** on the other hand does not. The grouping results of the CLM in **(b)** and **(d)** show that our model is consistent with these experiments.

Nothdurft studied human texture discrimination using patterns with different orientated line segments. He systematically measured the influence of structure density on human segregation performance. He found that texture discrimination depends not only on form, but also on the spacing of texture elements. Humans commonly fail to segment widely spaced texture elements, despite their instantaneous segregation in close arrangements. For details see [Not85]. The types of textures used for the experiments and the grouping results obtained with the CLM are shown in Figure 44.

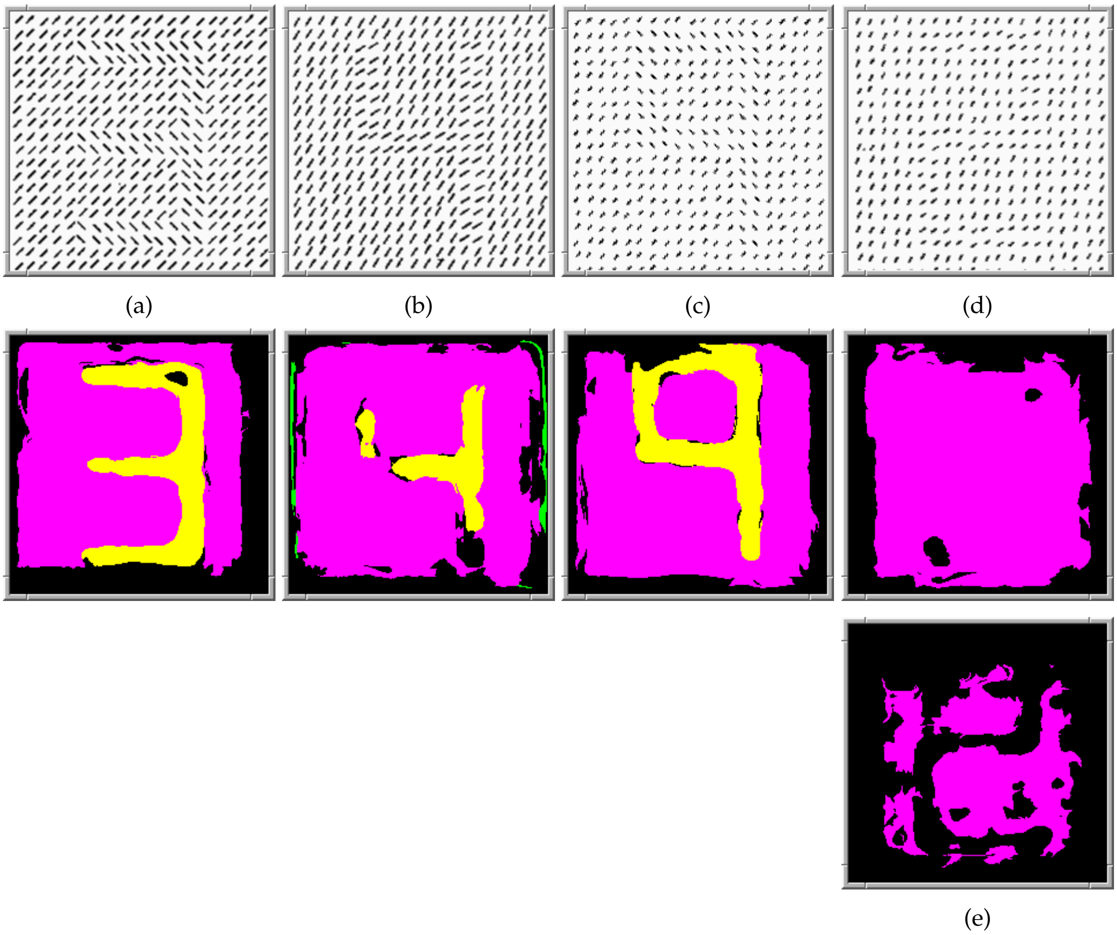


Figure 44: Dependence on orientation difference and line length for the segregation of equally spaced line arrays (taken from [Not85]): The patterns contain global figures in which line orientation differs from that in the surrounding texture field by 90° ((a),(c)) and 24° ((b),(d)). For each difference, patterns with different line lengths are shown ((a),(b): 0.8 raster width, (c),(d): 0.5 raster width). Nothdurft found, that textures with strong differences in line orientation ((a),(c)) can be discriminated down to shorter line lengths than textures with smaller orientation difference ((b),(d)). The corresponding perceptually grouped images obtained by our model are shown in the second row. The results are consistent with Nothdurft's observations: The model is able to segment (c), but fails to detect the number in (d), where the short line segments have less orientation difference. In (e) the threshold applied to the CLM's output was increased to $2.2h_r$. It can be seen that the activity pattern in the active layer contains the structure of the texture. This is due to the fact that the interaction function produces smaller interactions for pairs from different regions. Nevertheless, the model is not able to segment the pattern using the global set of parameters.

Nothdurft also presents another interesting pair of textures in [Not85]. The principle for the construction of these textures is based upon the Craik-Cornsweet illusion from luminance perception, which is illustrated in Figure 45. The illusion is caused by the fact that the human visual system has only limited sensitivity for absolute luminance levels. Subthreshold variation of luminance may remain undetected and areas displaying identical luminance values on an absolute scale may appear different when the noticeable luminance contrast to neighbouring areas is different. The analogous textures corresponding to Figure 45 are shown in Figure 46. The following description is taken from [Not85]:

> *From periphery to centre of the pattern in (a) orientation of single line elements changes continuously, with a sudden step in the opposite direction at points midway between periphery and centre. As far as texture is concerned, the central square appears to be homogeneous and obviously distinct from the background, even though lines in the centre of the square have the same orientation as lines at the pattern's edges. [...] The influence of local, rather than global structure variation is illustrated in (b). Although texture differences are identical, as far as the global variation is concerned they become perceptible only in the pattern with a sufficiently strong local structure gradient.*

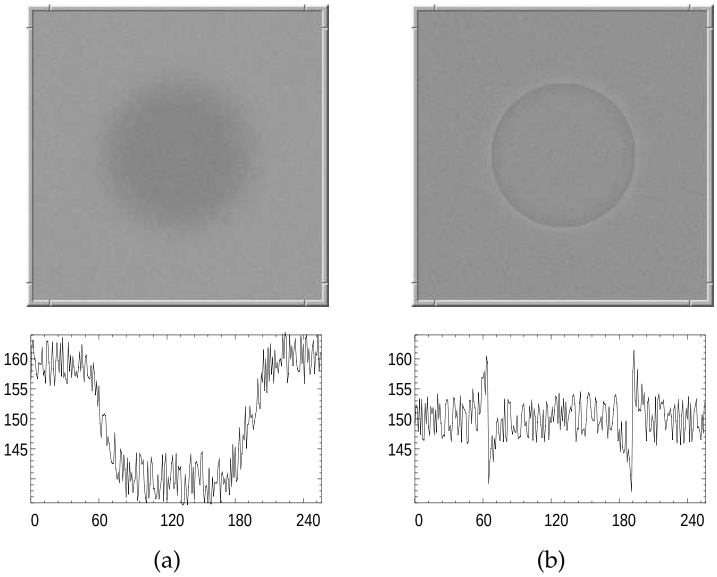


Figure 45: The Craik-Cornsweet illusion for luminance perception: **(a)** shows a pattern, where the luminance changes continuously from the border to the center, in **(b)** the luminance distribution is characterized by a strong discontinuity along the circle's radius. White noise was added to both images as indicated in the plot below the figures. The steady fixation of the left figure from a near distance causes the pattern to disappear. The changing of the fixation point causes the figure to abruptly appear again and to disappear after a while of steady fixation. This is not the case for the right figure, which remains present even after long steady fixation.

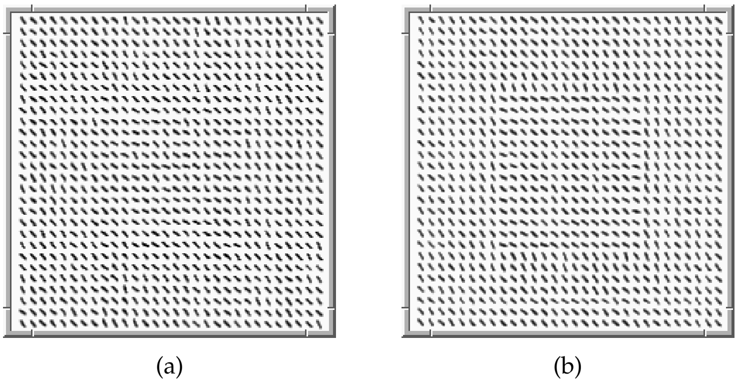
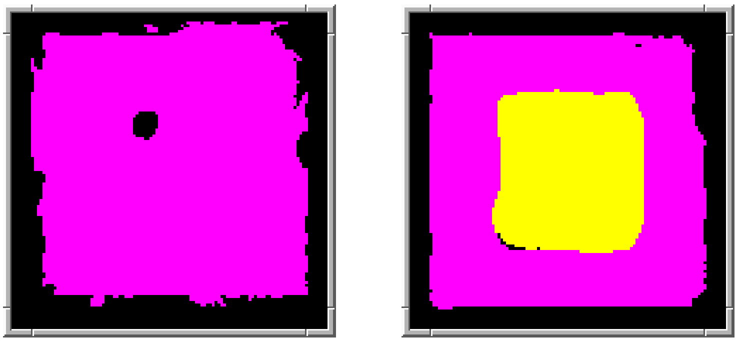


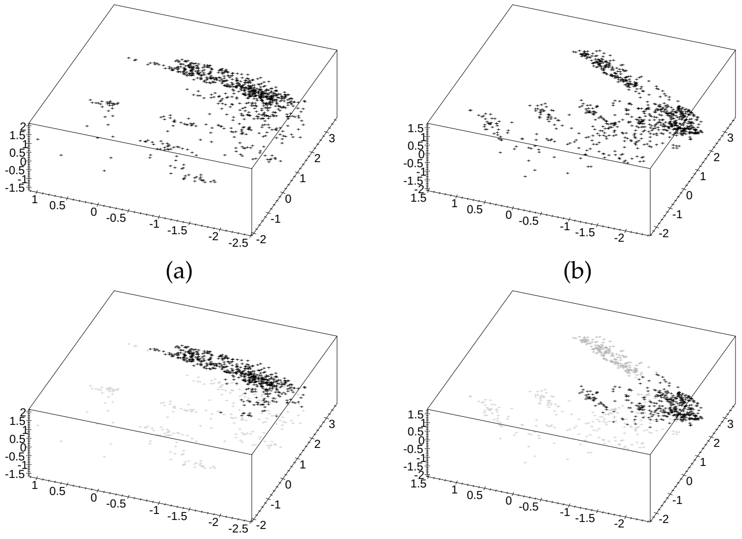
Figure 46: Craik-Cornsweet illusion for textured images (taken from [Not85]): Lines inside and outside the central square differ slightly in their average orientation. In **(b)** lines are arranged to generate a maximal texture gradient: In texture, both areas appear to be homogeneous and distinct. In **(a)** the pattern was reorganized and the same lines were distributed randomly (separately for each texture area): In consequence, the texture border disappears.

As can be seen in Figure 47 the perceptual organization found by our model is consistent with Nothdurft's observations. To understand why the model is able to mimique the human perception, we take a look at the distribution of the texture feature vectors. Figure 48 shows, that the continuous change of orientation in Figure 46(a) corresponds to a continuous data distribution in the feature space. In contrast, the feature space corresponding to Figure 46(b) reflects the discontinuity generated by the step in orientation change as a visible gap in the data distribution. This gap causes the CLM to separate the large cluster into two smaller ones as can be seen in Figure 48(c) and 48(d).



(a) (b)

Figure 47: Grouping result for Craik-Cornsweet textures: **(a)** and **(b)** show the grouping results of Figure 46 (a) and (b), respectively. Also in this example, the perceptual organization achieved by the model is consistent with human perception.



(a) (b)

(c) (d)

Figure 48: Feature vectors corresponding to Craik-Cornsweet textures and grouping result obtained with the CLM: The top row shows the first 3 components of the feature vectors plotted in $\mathbb{R}^3$. It can be seen that the structure of the two distributions is very similar. In **(a)** the main cloud stretches from the upper left to the lower right region in the back of the cube. In **(b)** the distribution is almost the same, but a gap is visible in the cloud of data. This gap corresponds to the discontinuous change in the orientation of the line segments. In **(c)** and **(d)** it can be seen that the CLM's clustering behaviour does not only depend on the size of the individual groups, but also on their structure: The gap causes a segmentation of the large cluster into two smaller ones. The grey points in (c) and (d) correspond to those activities whose output is smaller than the proposed threshold of $1.2h_r$.

3.5.2 Natural Textures

In this section we present the application of the CLM to a set of images constructed of natural textures taken from the popular Brodatz [Bro66] album. This album contains photographs from natural textures, such as water, grass, leather, sand, bricks and so on. Test images from this album are of common usage in the field of texture segmentation. We use images from a database taken from Hofmann et al [HPB96] which contains pictures each assembled of five different textures. We will present a few examples, which show the most important properties of our model if applied to this kind of natural images.

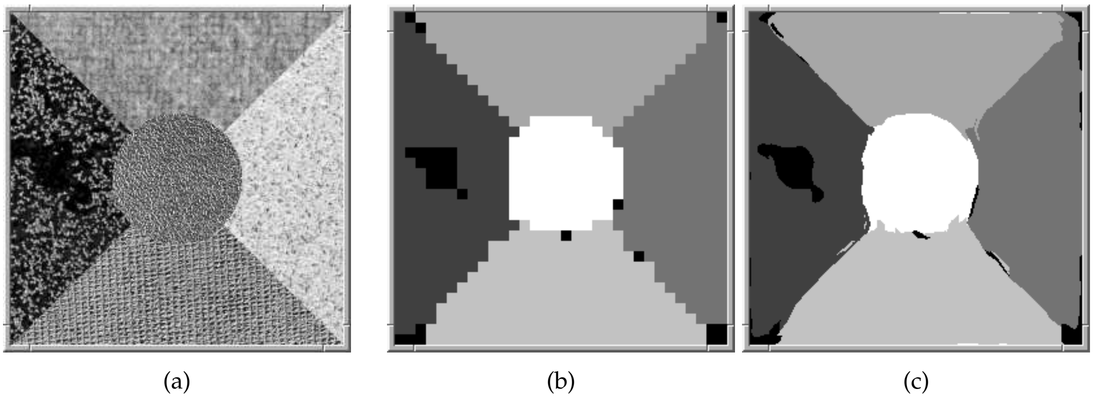
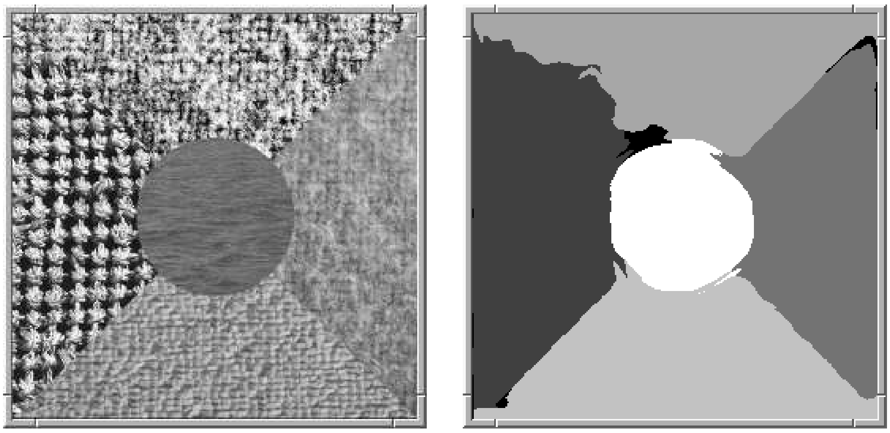


Figure 49: First example for natural texture segregation: **(a)** shows a test image taken from [HPB96]. In **(b)** the grouping result on the subsampled feature vectors is displayed, **(c)** shows that the resolution enhancement as described in section 3.4.3 provides a more accurate localization of the texture boundaries. Note that the black area in the left region does not depict a salient group. The activities in this area do not exceed the threshold of $1.2h_r$, which expresses that this area is dissimilar to the rest of the region. A visual inspection of the input image shows indeed, that the texture looks significantly different at this position. This grouping behaviour could be used in industrial inspection to detect flaws in certain kinds of material.

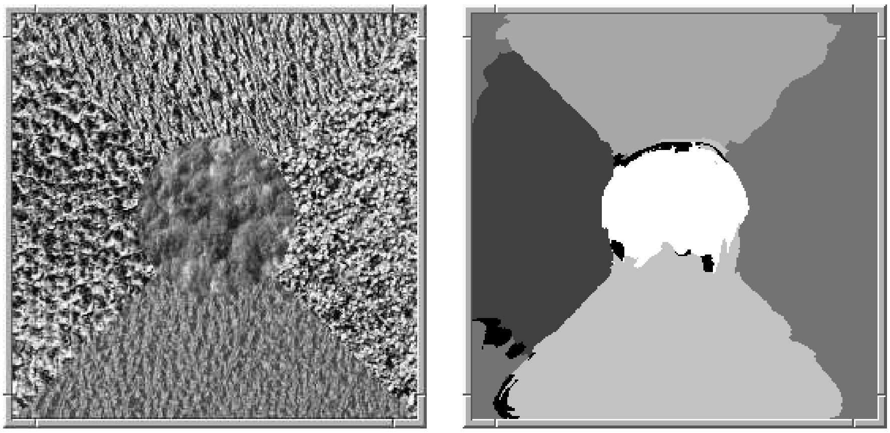
The images presented in this section show, that the grouping results obtained with the CLM in connection with the proposed feature extraction mechanisms are generally in good accordance with human texture perception. In some cases the model has difficulties to detect the borders of different textures properly. These misclassifications are occurring in those examples, where different texture regions have a great similarity in visual appearance. As the results of Figure 49 and 52 show, also local differences within otherwise uniform textures are detected.

Figure 51 shows, that the combination of the two principles of Similarity and Proximity is not sufficient to produce satisfying results in every case. The reason for this is that the principle of Proximity only utilizes absolute distances. The construction of the feature interaction function according to (42) does not include any information about the connectedness of two regions, which in turn is an important principle of human perception: We only tend to group those areas together which are connected or share a common region.



(a) (b)

Figure 50: In this example also for the human observer the upper left region of the image looks ambiguous. Without prior knowledge about the pattern arrangement it is hard to say, where one region ends and the other begins. This is also reflected by the grouping result of the CLM in **(b)**, which shows that the texture description we use is not able to discriminate between the two textures in the upper left region.



(a) (b)

Figure 51: This example shows that the model is not able to differentiate between the four corners of the image. Furthermore, the border between the center and lower texture is not detected properly. Both effects are caused by the great similarity of the textures in the corresponding regions. The splitting of the left region is probably caused by the different contrasts within the texture: The lower part has significantly lower contrast than its central region. Note, that the incorporation of the Law of Proximity does not forbid such a grouping result: The upper right region has the same distance from the lower right as the erroneous labelled lower left region. What would help in this case is a principle of Connectedness – which is actually a further Gestalt principle: Only those regions should be grouped together, which form a connected area.

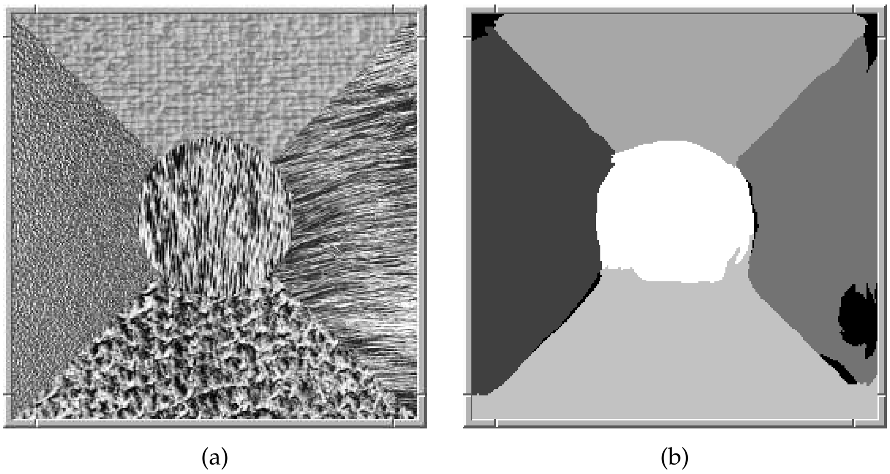

(a) (b)

Figure 52: In this example the nonuniform texture in the right region causes a hole in the grouped image. Also note, that the border between the lower and the center texture is not detected properly, because the two textures look too similar.

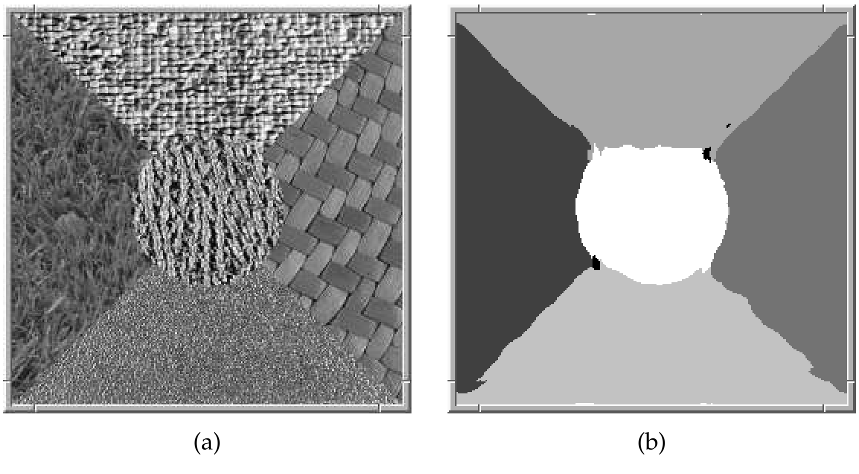

(a) (b)

Figure 53: This example shows, that textures with different scales are properly segmented. The structure in the texture of the right region is of a much larger scale than the structure in the lower texture. Despite the border of the upper and center region, all other borders are detected with high accuracy.

To demonstrate which features are used for the segmentation of natural textures, we use the same technique as described above for Figure 39. In Figure 54 the first 4 principal components of the feature vectors extracted from Figure 53 are plotted. The corresponding linear combinations of the features are depicted below. As can be seen in this example, the variance in the channels is an important criterion for the segmentation of natural textures. This is also consistent with the observation of Manjunath and Ma [MM96c], who noticed that "the use of the σ_{mn} feature in addition to the mean improves the retrieval performance considerably".

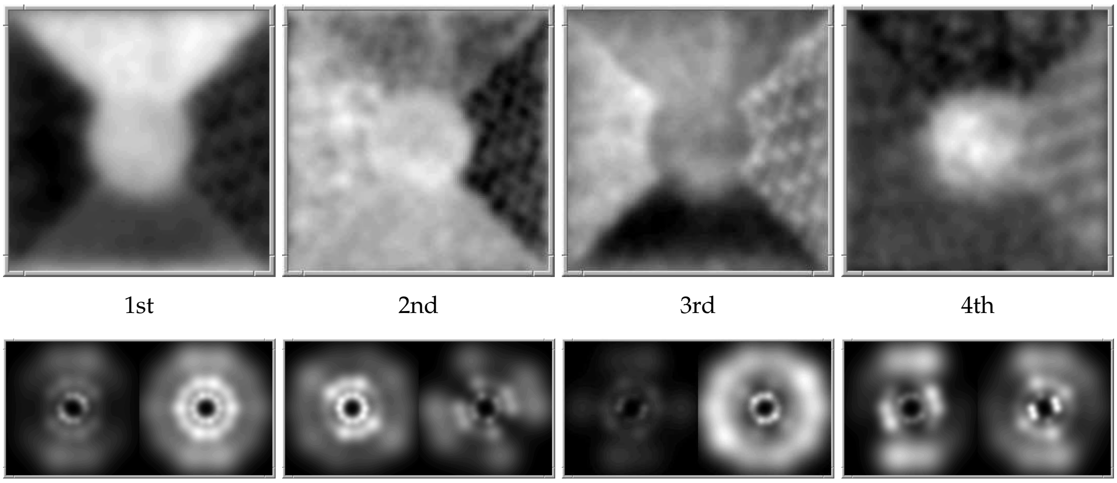


Figure 54: Principal components of feature vectors extracted from Figure 53 and their corresponding linear combinations in the feature space: As the latter show, the features generated from the variance in the channels contribute more to the segmentation of the image as in the case of micropattern textures (compare with Figure 40 on page 38).

3.5.3 Gestalt Laws

In this section we will present the grouping results of the CLM if applied to some of the images which illustrate the Gestalt laws proposed by the Gestalt psychologists – see also section 1.1.1. All images in this section were created using the *xfig*-drawing program.

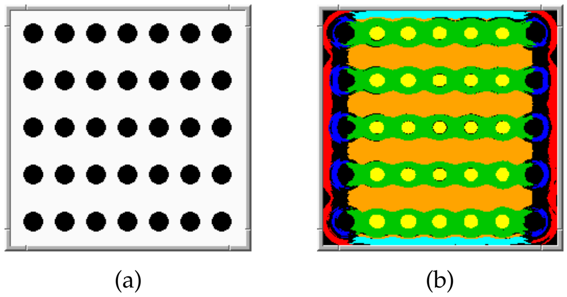


Figure 55: (a) shows an image which is often used to illustrate the Law of Proximity [Rob97]]: Because the horizontal spacing between the elements is smaller than the vertical spacing, we perceive five horizontal groups. In **(b)** the grouping result obtained with our model is shown. It can be seen, that despite of the border region, there are three distinct groups: The first (magenta) corresponds to the elements themselves, the second (green) connects the elements together, forming a horizontal structure. The third group (orange) expresses the horizontal structure generated from the background. Therefore, the perceptual grouping of this image mirrors the human introspection very well.

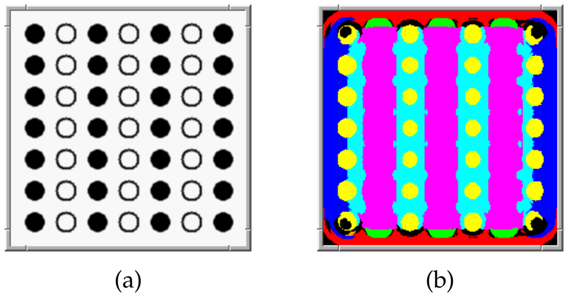


Figure 56: (a) is a common example to visualize the Law of Similarity [Rob97]: In contrast to the figure above, the spacing between the elements is constant, but the elements themselves consist of different stimuli arranged in a vertical structure. We therefore perceive vertical groups. The grouping result in **(b)** shows that our model also produces vertical arranged patterns. Note, that the filled elements form a salient group (yellow), but the open elements do not.

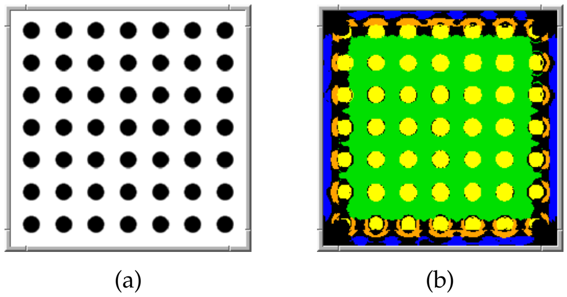


Figure 57: This figure shows equally spaced stimuli which do not differ in appearance. Since there is no information in this image indicating any vertical or horizontal structure, we just perceive black dots on a uniform background. As **(b)** shows, this is exactly the way, how our model groups the image.

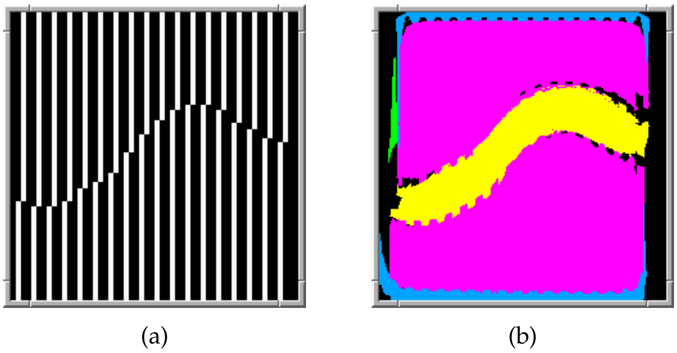


Figure 58: In **(a)** a pattern of broken lines is shown. Because the points marked by the discontinuities lie on a smooth line, we perceive a wavelike illusory contour. The grouping result obtained with the CLM also shows this wavelike contour. For a description of how this result is actually achieved, see the text.

As the results show, the perceptual grouping behaviour of our model is also in these cases consistent with human perception. An example of special interest is depicted in Figure 58. Although the principle of Connectedness or Good Continuation is not incorporated into our model, the grouping result is in good accordance with human perception. To understand this result, we again apply the proposed method of the inspection of that linear combination of features, which causes the segregation. As can be seen in Figure 59(a), relevant features are those which are extracted from channels sensitive to horizontal gratings. The mean features do not contribute to the segregation, because even after the nonlinearity the average response is close to zero in all regions (compare the black and white areas of Figure 59(b) which denote negative and positive response, respectively). The variance feature on the other hand separates those points where the lines are broken very well from the otherwise homogeneous background. The variance of the signal is computed over a region greater than the channel's receptive field and the spacing of the lines is rather dense. Therefore, the mechanism which produces a continuous line of connected stimuli is part of the feature extraction and not of the grouping process itself. This is also supported by Figure 60, where the spacing of the broken lines is increased, such that the range of the largest receptive field used within the feature extraction is smaller than this spacing. As can be seen in the grouping result, our model is then not able anymore to connect the elements. Furthermore, the large spacing of the lines destroys the illusion of a homogeneous background. Now the lines themselves constitute a salient group. Again, this result does not disagree with the human perception of Figure 60. Compared to Figure 58 the perception of a continuous wavelike line is indeed reduced significantly.

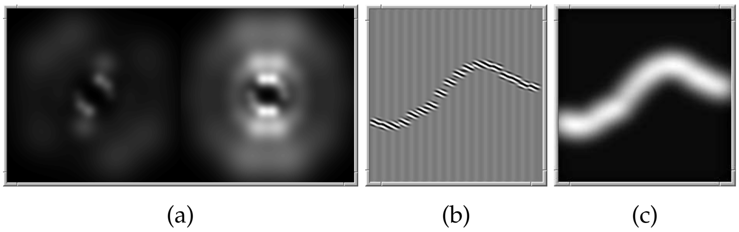


Figure 59: **(a)**: Linear combination of 2D Gabor filters corresponding to 1st principle component. Relevant features are those extracted from channels sensitive to horizontal gratings. **(b)**: channel response after the nonlinearity, and **(b)**: the corresponding feature σ_{23} according to (32). For a discussion, see the text.

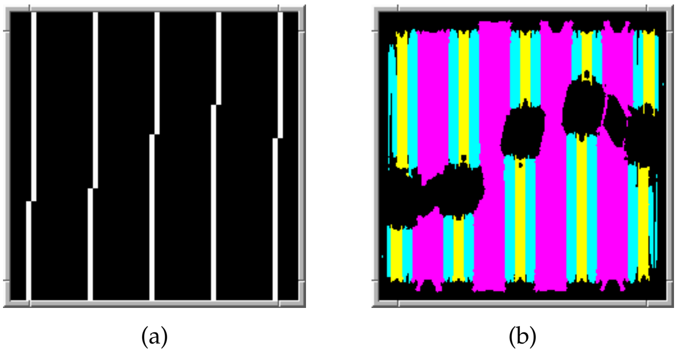

(a) (b)

Figure 60: **(a)** shows the same figure as 58, but with increased line spacing. **(b)** shows the corresponding grouping result. The homogeneous background of 58 disappears and the vertical structure of the lines becomes visible. Furthermore, the broken line elements are not connected anymore.

3.6 Application to Baufix Scenes

In this section we will present the grouping results if we apply our model to some simple scenes from a "Baufix" toyworld. We propose a modification of the feature extraction process and will show that for a certain class of images the modified model is able to achieve a reasonable perceptual grouping.

Consider the image shown in Figure 61(a). It shows some pieces of the wooden set of construction parts which belong to the "Baufix" world. Human beings have no difficulties to identify three crossing bars. However, Figure 61(b) shows that our model does not achieve a grouping of the image consistent with our experience. We gain more insight into the reason why this happens, if we inspect the principal components of the feature vectors, on which the grouping process is based.

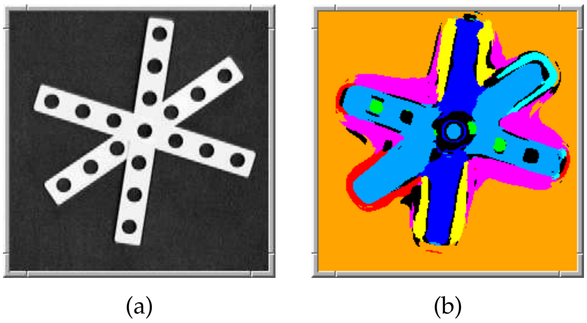
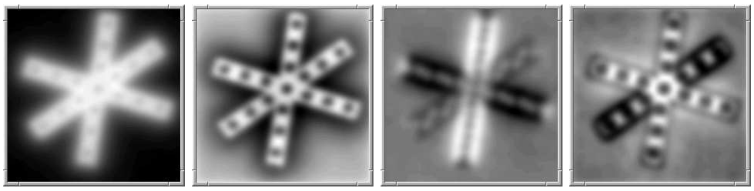

(a) (b)

Figure 61: **(a)** shows an image of three crossing Baufix bars, and **(b)** the grouping result achieved by our model. It can be seen that this perceptual organization is not consistent with human perception.

As we can see in Figure 62, the feature vectors along the direction of the first principle component separate the three pieces from the background. The inspection of the eigenvalues (see Figure 63) shows that the first p.c. contributes to approximately 80% of the variation in the data. Therefore, the texture features mainly describe the separation of the objects from the background. So, the idea is to take all those feature vectors corresponding to the background from the data set to "leave more space" for the other features to structure the available volume in which the data is clustered by the CLM. Therefore, we propose the following modification to the model:



1st 2nd 3rd 4th

Figure 62: The first 4 principle components of the feature vectors extracted from the Baufix image.

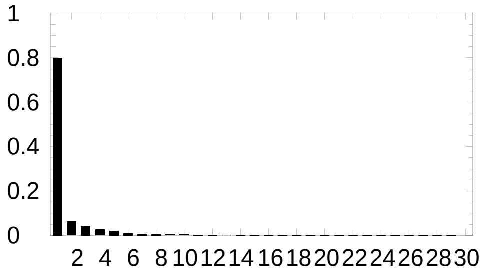
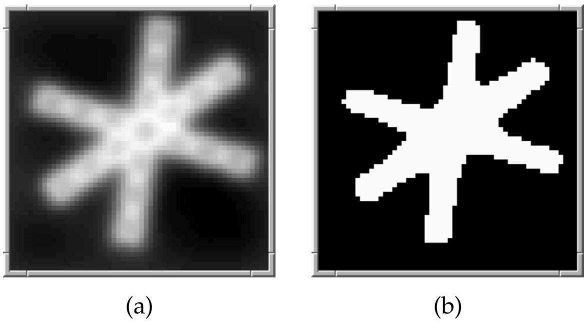


Figure 63: Eigenvalues corresponding to principle components depicted in Figure 62

Since the zero d.c. filters do not respond to the uniform luminance, we can assume that the channel responses in the background are very low and therefore use this information to segment the objects. We compute the Euclidean length of the feature vector $\mathbf{h}(x, y)$ as defined in (34) and apply a threshold operation to separate regions with high from those with low response. Figure 64 shows the result of this operation. Using this technique we also include the "holes" into our thresholded image, which would be lost by applying a simple threshold operation to the input image itself. The reason for this is that the holes are covered by long range 2D Gabors which give a significant response in this regions.



(a) (b)

Figure 64: Thresholding of the input image based on the Euclidean length of feature vectors $\mathbf{h}(x, y)$: In **(a)** the Euclidean length of the feature vector is coded in greyvalues and scaled to the interval [0,255]. In **(b)** a simple threshold operation was applied to **(a)** and shows those areas of the image which are "regions of interest".

The next stages of feature processing are exactly the same as described in section 3.4. The only difference is that we omit the those feature vectors which describe the background of the image. After the KHL of the data set, we obtain the first 4 principle components as shown in Figure 65.

If we now apply our model with the type of interaction function as defined by (42) and increase the sensitivity for Similarity by decreasing R_{sim}, we obtain a grouped image as

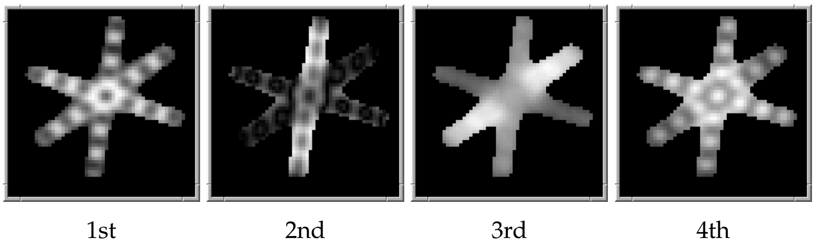


Figure 65: The first 4 principle components after the elimination of the background information.

shown in Figure 66.

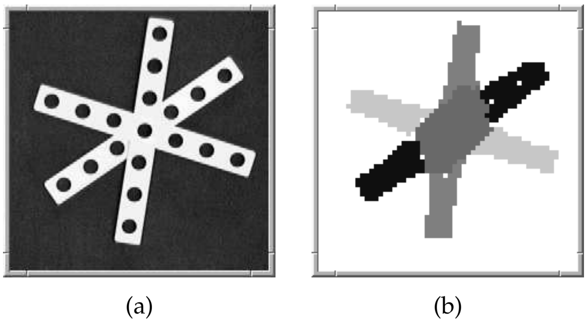
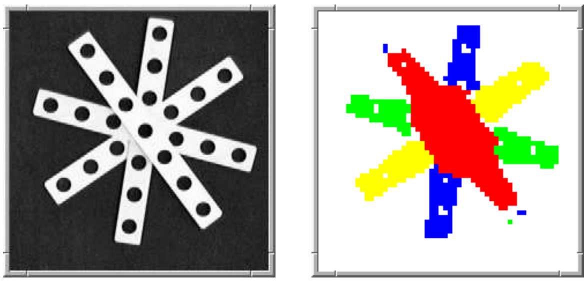


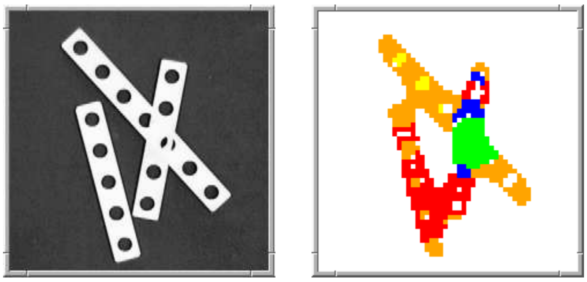
Figure 66: Grouping of Baufix image: **(a)** shows the input image again, and in **(b)** the grouping result obtained with the CLM is shown. It can be seen that this perceptual organization describes the image contents very well: The system identifies four salient groups. Three of them correspond to the three bars and the 4th depicts the center of the cross which belongs equally to all pieces.

Also, the next image shows that an arrangement of four crossing bars is grouped in a sensible way. However, as can be seen in Figure 68 generalization to arbitrary organized Baufix elements is quite poor. The reason for this lies in the set of feature vectors we use for the perceptual grouping: They are especially designed to measure the textual appearance in an image. As we see in Figure 66 and 67 good results are achieved if the single pieces are separated by a large enough distance which exceeds the range of the 2D Gabor filters (the ends of the bars are far away from each other, only in the relatively small center region the pieces are close resp. overlapping). In this cases, the feature vectors are able to measure the orientation of a single piece. On the other hand, if the Baufix parts are close together, the texture features – which are computed over a certain region – are not able to describe a single element. Instead they describe the appearance of a region which contains more than one element and are therefore not a good choice to describe the perceptual organization of a general Baufix scenario.



(a) (b)

Figure 67: Grouping of four crossing Baufix bars: Also for this example a reasonable perceptual organization is achieved.



(a) (b)

Figure 68: Grouping of arbitrary organized Baufix elements: As can be seen in this image, generalization to arbitrary Baufix scenes is quite poor. For a discussion, see the text.

4 Discussion and Outlook

4.1 Summary

In this report we have described a neuro-dynamical system which models the visual perception of images based upon the Gestalt principles of Proximity and Similarity.

In section 2 we have described the recurrent Competitive Layer Model (CLM) which was first introduced by Ritter [Rit90] as a model for spatial feature linking. We have proposed the "Heat Bath Update Rule", a new dynamics for the CLM and have shown that a Lyapunov function constructed by Feng [Fen97] also applies to the proposed stochastical dynamics.

In section 3 we have motivated the feature extraction in order to obtain the input stimuli for the CLM by early vision mechanisms found in the visual cortex of mammals. We have applied a bank of linear filters to each point of the input image and obtained a set of 15 images containing the responses of each filter.

Based on Malik and Perona [MP90] we have presented arguments that only even symmetric mechanisms are utilzed in preattentive texture segregation and that we need some sort of nonlinearity to model human perception.

In order to obtain a description of textual appearance, for each pixel of the input image we have constructed a 30-dimensional feature vector based on the statistical properties of the filter responses. Furthermore we have shown that the reduction of dimensionality by a Karhunen Loeve Transformation produces a set of features which is more reliable for a stable texture segmentation.

Because the computational complexity of the simulation of the CLM's dynamics is of the order $\mathcal{O}(N^2)$, we have proposed a subsampling mechanism which significantly reduces the amount of data which is then grouped by the CLM. This grouping is characterized by a pairwise feature interaction function composed of two parts: The first is based upon a distance metric on the texture feature space and corresponds to the Gestalt Law of Similarity, and the second incorporates the Law of Proximity using the Euclidean distance of the feature vectors spatial positions. To achieve a grouping result with higher resolution than of the subsampled data, we have used the output of the CLM and a nearest neighbour classificator to label the original feature space.

By putting everything together we have constructed a model of visual perception which was applied to a variety of test images. The results have shown that the model is able to reproduce a large amount of phenomena related to human texture perception.

An important advantage of the Competitive Layer Model over other models which are used in texture segmentation is, that the number of salient groups does not have to be specified beforehand. Only the maximal expected number of distinct groups is given as a parameter to the CLM. Practically, this number is chosen so high that it does not limit the grouping performance. The CLM then dynamically allocates that number of layers, which are necessary to describe the image.

The application to "Baufix" scenes has revealed that the texture features are not sufficient to produce stable grouping results for a larger set of such inputs. In the next section we will as an outlook present the introduction of a "local association field", which might be incorporated into the model to produce more stable grouping results for images consisting of Baufix elements.

4.2 Outlook

In the feature interaction function we used for the perceptual grouping only the Gestalt principles of Similarity and Proximity were incorporated. Another important principle is the Gestalt law of "Good Continuation". As a first test we have implemented a feature interaction function based on the orientation of line elements. Using the response of 5 different orientated Gabors we were able to determine the orientation of a stimulus with an accuracy of about 5 degrees. Based on the studies of Field et al [FHH93] we have implemented a "local association field" as depicted in Figure 69.

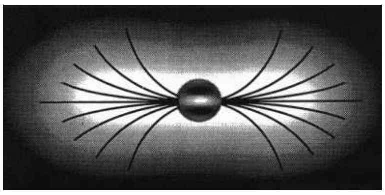


Figure 69: The association field as proposed by Field et al (taken from [FHH93]): Features are interacting positive with neighbouring features if lying on a smooth line – denoted by the black curves.

The first results of the grouped images as shown in Figure 70 look quite promising, and a further investigation might show that an interaction function based on such an "association field" is able to produce reliable results if applied to images from the Baufix scenario, where the orientation of edges plays a significant role.

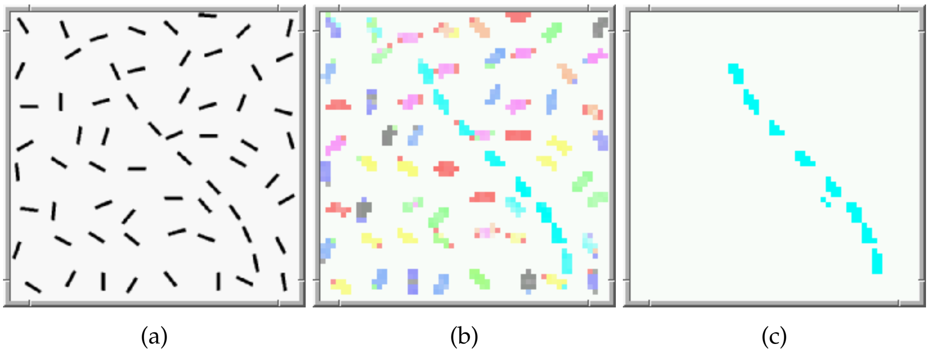


Figure 70: Grouping of orientated stimuli with an interaction function based on a "local association field": (a) shows the input image consisting of orientated line elements. In (b) the grouping result of the CLM is shown, and (c) shows the output after the application of a threshold function which leaves only salient groups.

References

[AH82] Duane G. Albrecht and David B. Hamilton. Striate cortex of monkey and cat: Contrast response function. *Journal of Neurophysiology*, 48(1):217–237, 1982.

[BCG90] Alan Conrad Bovik, Marianna Clark, and Wilson S. Geisler. Multichannel texture analysis using localized spatial filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(1):55–73, January 1990.

[Ber91] James R. Bergen. Theories of visual texture perception. In David Regan, editor, *Spatial Vision*, volume 10 of *Vision and Visual Dysfunction*, pages 114–134. Boca Raton, 1991.

[BJ83] James R. Bergen and Bela Julesz. Rapid discrimination of visual patterns. *IEEE Transactions on Systems, Man, and Cybernetics*, 13:857–863, 1983.

[Bro66] P. Brodatz. *Texture: A Photographic Album for Artists and Designers*. Dover, NewYork, 1966.

[Cae88] Terry M. Caelli. An adaptive computational model for texture segmentation. *IEEE Transactions on Systems, Man and Cybernetics*, 18(1):9–17, February 1988.

[CL91] Charles Chubb and Michael S. Landy. Orthogonal distribution analysis: A new approach to the study of texture perception. In Michael S. Landy and J. Anthony Movshon, editors, *Computational Models of Visual Processing*, pages 291–301. MIT Press, 1991.

[CR97] Matteo Carandini and Dario L. Ringach. Predictions of a recurrent model of orientation selectivity. *Vision Research*, 1997. in press.

[Dau85] John G. Daugman. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *Journal of the Optical Society of America A*, 2(7):1160–1169, July 1985.

[Dau88] John G. Daugman. Complete discrete 2D Gabor transforms by neural networks for image analysis and compression. *IEEE Transactions on Acoustincs, Speech and Signal Processing*, 36(7):1169–1179, July 1988.

[DHW94] Dennis Dunn, William E. Higgins, and Joseph Wakeley. Texture segmentation using 2D Gabor elementary functions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(2):130–149, February 1994.

[Fen97] Jianfeng Feng. Lyapunov functions for neural nets with nondifferentiable input-output characteristics. *Neural Computation*, 9:43–49, January 1997.

[FHH93] David J. Field, Anthony Hayes, and Robert F. Hess. Contour integration by the human visual system: Evidence for a local "association field". *Vision Research*, 33(2):173–193, 1993.

[Fie87] David J. Field. Relations between the statistics of natural images and the response properties of cortical cells. *Journal of the Optical Society of America A*, 4(12):2379–2394, December 1987.

[FS89] I. Fogel and D. Sagi. Gabor filters as texture discriminator. *Biological Cybernetics*, 61:103–113, 1989.

[GBS92] Norma Graham, Jacob Beck, and Anne Sutter. Nonlinear processes in spatial-frequency channel models of perceived texture segregation: Effects of sign and amount of contrast. *Vision Research*, 32(4):719–743, 1992.

[Gro88] S. Grossberg. Nonlinear neural networks: Principles, mechanisms, and architectures. *Neural Networks*, 1:17–61, 1988.

[Hay94] Simon Haykin. *Neural Networks: a comprehensive foundation*. Macmillan, New York, 1994.

[HBS92] Peter J.B. Hancock, Roland J. Baddeley, and Leslie S. Smith. The principal components of natural images. *Network*, 3:61–70, 1992.

[HH96] Rudolf Hanka and Thomas P. Harte. Curse of dimensionality: Classifying large multi-dimensional images with neural networks. In *Proceedings of the IEEE European Workshop on Computer Intensitive Methods in Control and Signal Processing*, Prague, 1996.

[Hir89] Morris W. Hirsch. Convergent activation dynamics in continuous time networks. *Neural Networks*, 2:331–349, 1989.

[HPB96] Thomas Hofmann, Jan Puzicha, and Joachim Buhmann. A deterministic annealing framework for unsupervised texture segmentation. Technical Report IAI-TR-96-2, University of Bonn, 1996.

[HS74] Morris W. Hirsch and Stephen Smale. *Differential equations, dynamical systems, and linear algebra*. Academic Press, New York, 1974.

[IRSB95] Giacomo Indiveri, Luigi Raffo, Silvio P. Sabatini, and Giacomo M. Bisio. A neuromorphic architecture for cortical multilayer integration of early visual tasks. *Machine Vision and Applications*, 8:305–314, 1995.

[Jäh93] Bernd Jähne. *Digitale Bildverarbeitung*. Springer-Verlag, Berlin, 1993.

[JF91] Anil K. Jain and Farshid Farrokhina. Unsupervised texture segmentaion using Gabor filters. *Pattern Recognition*, 24(12):1167–1186, December 1991.

[JF96] Anil K. Jain and Farshid Farrokhina. A self-organizing network for hyperellipsoidal cluserting (hec). *IEEE Transactions on Neural Networks*, 7(1):16–29, January 1996.

[JGV78] B. Julez, E.N. Gilbert, and J.D. Victor. Visual discrimination of texture with identical third-order statistics. *Biological Cybernetics*, 31:137–140, 1978.

[JP87] J. Jones and L. Palmer. An evaluation of the two-dimensional Gabor filter model of simple receptive fields in cat striate cortex. *Journal of Neurophysiology*, 58:1233–1258, 1987.

[Jul81] Bela Julesz. Textons, the elements of texture perception and their interaction. *Nature*, 290:91–97, 1981.

[KKM95] Frederick A.A. Kingdom, David Keeble, and Bernard Moulden. Sensitivity to orientation modulation in micropattern-based textures. *Vision Research*, 35(1):79–91, 1995.

[KM97] Osamu Koseki and Fumitaka Matsubara. Cluster heat bath method on a Quasi-One-Dimensional Ising Model. *Journal of the Physical Society of Japan*, 66(2):322–325, February 1997.

[KMB82] J.J. Kulikowski, S. Marcelja, and P.O. Bishop. Theory of spatial position and spatial frequency relations in the receptive fields of simple cells in the visual cortex. *Biological Cybernetics*, 43:187–198, 1982.

[Kof62] Kurt Koffka. *Principles of Gestalt psychology*. Routledge & Paul, London, 1962.

[LB91] Michael S. Landy and James R. Bergen. Texture segregation and orientation gradient. *Vision Research*, 31(4):679–691, 1991.

[Lee96] Tai Sing Lee. Image representation using 2D Gabor wavelets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(10):959–971, October 1996.

[Len80] Peter Lennie. Parallel visual pathways: A review. *Vision Research*, 20:561–594, 1980.

[Low86] David G. Lowe. *Perceptual organization and visual recognition*. Kluwer Academic Publishers, Boston, 1986.

[MC93] B.S. Manjunath and Rama Chellappa. A unified approach to boundary perception: Edges, textures, and illusory contours. *IEEE Transactions on Neural Networks*, 4(1):96–107, January 1993.

[MKB79] K.V. Mardia, J.T. Kent, and J.M. Bibby. *Multivariate Analysis*. Academic Press, London, 1979.

[MM96a] W.Y. Ma and B.S. Manjunath. Image indexing using a texture dictionary. Technical report, University of California at Santa Barbara, 1996.

[MM96b] W.Y. Ma and B.S. Manjunath. Texture features and learning similarity. In *Proceedings of IEEE Int. Conference on Computer Vision and Pattern Recognition*, June 1996.

[MM96c] B.S. Manjunath and W.Y. Ma. Texture features for browsing and retrieval of image data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8):837–842, August 1996.

[MN92] Rakesh Mohan and Ramakant Nevatia. Perceptual organization for scene segmentation and description. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(6):616–634, June 1992.

[MP90] Jitendra Malik and Pietro Perona. Preattentive texture discrimination with early vision mechanisms. *Journal of the Optical Society of America A*, 7(5), May 1990.

[MRA97] B.W. Mel, D.L. Ruderman, and K.A. Archie. Translation-invariant orientation tuning in visual 'complex' cells could derive from intradendritic computations. In preparation. See `http://quake.usc.edu/publications.html`, 1997.

[Not85] H.C. Nothdurft. Sensitivity for structure gradient in texture discrimination tasks. *Vision Research*, 25(12):1957–1968, 1985.

[Not91] H.C. Nothdurft. Different effects from spatial frequency masking in texture segregation and texton detection tasks. *Vision Research*, 31:299–320, 1991.

[PC89] T.S. Parker and L.O. Chua. *Practical Numerical Algorithms for Chaotic Systems*. Springer Verlag, New York, 1989.

[PF81] D.A. Pollen and S.E. Feldon. Phase relationship between adjacent simple cells in the visual cortex. *Science*, 212:1409–1411, 1981.

[PS94] Uri Polat and Dov Sagi. The architecture of perceptual spatial interactions. *Vision Research*, 34(1):73–78, 1994.

[PTH96] Olaf Pichler, Andreas Teuner, and Bedrich J. Hosticka. A comparison of texture feature extraction using adaptive Gabor filtering, pyramidal and tree structured transforms. *Pattern Recognition*, 29(5):733–742, 1996.

[PTVF92] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C*. Cambridge University Press, New York, 2nd edition, 1992.

[RHC88] I. Rentschler, M. Hubner, and T. Caelli. On the discrimination of compound Gabor signals and textures. *Vision Research*, 28:279–291, 1988.

[Rit90] Helge Ritter. A spatial approach to feature linking. In *International Neural Network Conference Paris*, 1990.

[RMS92] H.J. Ritter, T.M. Martinez, and K.J. Schulten. *Neuronale Netze*. Addison-Wesley, 1992.

[Rob97] Adrian Robert. From contour completion to image schemas: A modern perspective on Gestalt psychology. Technical report, Department of Cognitive Science, University of California, San Diego, February 1997.

[RW91] Todd Reed and Harry Wechsler. Spatial/spatial-frquency representations for image segmentation and grouping. *Image and Vision Computing*, 9(3):175–193, June 1991.

[SNS95] David C. Somers, Sacha B. Nelson, and Mriganka Sur. An emergent model of orientation selectivity in cat visual cortical simple cells. *Journal of Neuroscience*, 1995. in press.

[SSC95] Anne Sutter, George Sperling, and Charles Chubb. Measuring the spatial frequency selectivity of second-order texture mechanisms. *Vision Research*, 35(7):915–924, 1995.

[Sto90] David G. Stork. Do Gabor functions provide appropriate descriptions of visual cortical receptive fields? *Journal of the Optical Society of America A*, 7(8):1362–1373, August 1990.

[Tre86] Anne Treisman. Features and objects in visual processing. *Scientific American*, 255:106–125, November 1986.

[Tur86] M.R. Turner. Texture discrimination by Gabor functions. *Biological Cybernetics*, 55:71–82, 1986.

[VAT82] Russell L. De Valois, Duane G. Albrecht, and Lisa G. Thorell. Spatial frequency selectivity of cells in macaque visual cortex. *Vision Research*, 22:545–559, 1982.

[Ver91] David Vernon. *Machine Vision*. Prentice Hall, New York, 1991.

[Ves93] Franz Vesely. *Computational physics: Einführung in die computative Physik*. WUV Universitätsverlag, Wien, 1993.

[VT83] Karen K. De Valois and Roger B.H. Tootell. Spatial-frequency inhibition in cat striate cortex cells. *Journal of Physiology*, 336:359–376, 1983.

[VV88] Russell L. De Valois and Karen K. De Valois. *Spatial Vision*. Oxford University Press, New York, 1988.

[VYH82] Russel L. De Valois, E. William Yund, and Norva Hepler. The orientation and direction selectivity of cells in macaque visual cortex. *Vision Research*, 22:531–544, 1982.

[WAJA89] Andrew B. Watson and Jr. Albert J. Ahumada. A hexagonal orthogonal-oriented pyramid as a model of image representation in visual cortex. *IEEE Transactions on Biomedical Engineering*, 36(1), January 1989.

[Wat86] Andrew B. Watson. Ideal shrinking and expansion of discrete sequences. Technical report, NASA - National Aeronautics and Space Administration, January 1986.

[Wat87] Andrew B. Watson. Efficiency of a model human image code. *Journal of the Optical Society of America A*, 4(12):2401–2417, December 1987.

[Wat90] Andrew B. Watson. Algotecture of visual cortex. In C.B. Blakemore, editor, *Vision: Coding and efficiency*, pages 391–410. Cambridge University Press, 1990.

[Wer96] Heiko Wersing. A neurodynamical model of perceptive grouping. Diplom Thesis, Universiät Bielefeld, 1996.

[WSR97] Heiko Wersing, Jochen J. Steil, and Helge Ritter. A layered recurrent neural network for feature grouping. In M. Hasler W. Gerstner, A. Germond and J.-D. Nicoud, editors, *Proc. International Conference on Artificial Neural Networks Lausanne*, pages 439–444, 1997.

[Zhe95] Yong-Jian Zheng. Feature extraction and image segmentation using self-organizing networks. *Machine Vision and Application*, 8:262–274, 1995.