Gabor Wavelet Networks for Object Representation

Dissertation

zur Erlangung des akademischen Grades
Doktor der Ingenieurwissenschaften
(Dr.-Ing.)
der Technischen Fakultät
der Christian-Albrechts-Universität zu Kiel

Volker Krüger

CHRISTIAN-ALBRECHTS-UNIVERSITÄT

KIEL

1. Gutachter _____

2. Gutachter _____

Datum der mündlichen Prüfung _____

Abstract

The choice of an object representation is crucial for the effective performance of cognitive tasks such as object recognition, fixation, etc. because how robustly and efficiently vision tasks can be performed depends on the choice of the representation.

In this work we introduce Gabor Wavelet Networks as an effective and efficient object representation. Gabor Wavelet Networks represent objects with sets of weighted Gabor wavelets that are specifically chosen to reflect the properties of the represented objects. The degrees of freedom of each Gabor wavelet are allowed to vary continuously. This is in contrast to the well-known bunch graph approach, also based on Gabor wavelets, where the wavelet parameters are chosen according to a specific discrete scheme that is based on the discrete wavelet transform. The optimized parameter choice of the Gabor Wavelet Networks allows the representation to be very sparse and specific to the represented objects. We will show experimentally that the specificity of the parameters can be exploited for the recognition of faces. Recognition rates are shown to be as high as 97%.

The degrees of freedom of wavelets allow any affine deformation that does not involve shearing. Adding shearing to the degrees of freedom, Gabor Wavelet Networks can easily be deformed affinely. This makes tracking applications very easy.

Gabor Wavelet Networks represent objects through linear combinations of Gabor wavelets. Changing the dimensionality of the linear combination changes the complexity and precision of the representation. Computations based on the representation also vary in their complexity and precision. Controlling the dimensionality of the linear combinations used in vision tasks allows desired degrees of precision or speed to be achieved. This will be referred to as *progressive attention*. Affine variability and progressive attention will be tested in an affine real-time face tracking experiment.

The scalar weights in the linear combination of wavelets can be computed by applying each Gabor wavelet as a filter. The filter is applied to (projected onto) the image only at the position indicated by the wavelet parameters. The relation between the filter responses and the weights is linear, and the responses contain the same visual information as the weights. Therefore, the optimized Gabor Wavelets of a network can be used not only for representation of an object but also for optimized filtering. We have exploited this in a head-pose estimation experiment. Our experiments have shown that the optimized filtering scheme is superior to a filtering scheme in which the filters are homogeneously distributed. The pose-estimation error was as low as $0.20°$.

Acknowledgement

First, I would like to thank Prof. Gerald Sommer very much for his supervision based on his broad experience, for his way of always finding time for discussions, for his encouraging support; and for a cheerful and stimulating research environment. Furthermore, his insistence on clear writing led me to deeper insight as well as to a more precise presentation and discussion of the experiments and their results.

I am grateful to Prof. R. Koch for his interest in my work and his constructive comments.

Special thanks go to Prof. K. Daniilidis for the enjoyable collaboration over many years and for endless and inspiring discussions. He was always an invaluable driving force and source of ideas.

The intensive collaboration and inspiring discussions with Dr. U. Mahlmeister and Dr. T. Bülow gave me great insight into signal and wavelet theory and helped me to see may things from different perspectives. Dr. N. Krüger also collaborated by sharing his experiences with the face recognition problem. For the pleasure of these most creative collaborations, I would like to thank them.

I would like to thank A. Happe for the inspiring discussions and his help with the face recognition experiments. Also I would like to thank S. Bruns for his help with the pose estimation experiments.

I would like to thank Prof. G. Sandini for his support over the years and Prof. A. Rosenfeld and Prof. R. Chellappa for their time and valuable comments.

For numerous discussions and critical and motivating comments at Maryland and during conferences and visits, I would like to thank Prof. Y. Aloimonos, Prof. J. Barron, Prof. R. Herpers, Dr. K. Toyama.

For their collaboration I thank Dr. Pauli, Dr. Perwass, Dr. W. Yu, B. Rosenhahn, S. Buchholz and M. Feldsberg.

And, last but not least, I would like to thank F. Maillard for keeping the good spirits up when times of research are dark.

This work was made possible partly by the Deutsche Forschungsgemeinschaft (DFG) with Grant So 322/1-2 u. 1-3.

Contents

Chapter 1

Introduction

It is a crucial question how object information, or image information in general, should be represented for cognitive systems to perform effectively and efficiently. A good representation is a hallmark for robust and successful performance and the choice of a representation has far-ranging consequences for the entire system that relies upon it.

The reasons for this are manifold:

1. The representation implies the distance and similarity measurements. This is important for, e.g., recognition tasks.

2. When dealing with digital images, the representation that encodes the image information usually results in a data reduction, and it is again the type of information representation that determines which image information is relevant, i.e. is encoded, and which is not.

3. Other important properties are invariance properties with respect to perceived object sizes, geometric deformations, and especially illumination (color constancy [Funt *et al.*, 1998; Rock, 1985]).

4. The abstraction capability of the representation has to be mentioned. On the one hand, the representation can take information literally, i.e. it can be data-driven or *appearance-based*. This may be useful in some situations; in other situations it should be avoided: well known are the amusing translations that derive from modern language translation programs*. On the other hand, the representation can be abstract, i.e. model-driven.

5. Also, the representation determines whether geometric information is represented or discarded. How important geometric information is was demonstrated e.g. by [Zeki, 1993],

*like those included, e.g., in *Alta Vista*.

who reports absurdities that happened to humans whose brains lost their ability to represent geometric information.

6. A further aspect is the efficiency of the representation. Low reaction time is of vital importance to many cognitive systems. However, the reaction speed depends on the data that needs to be evaluated and on the number of filters that need to be applied. The possibility of controlling computation speed by controlling the complexity of the data representation should be of great use for the construction of active vision systems.

The above points are only a selection from a variety of points that reveal the role of information representation in cognitive systems.

Various image information representations for artificial cognitive systems have been developed[†]. We want to restrict our consideration to 2-D object representations only, and leave out representations other than for objects and representations of higher dimensionality, such as e.g. [Vetter and Blanz, 1998]. Therefore, when we use the term "object representation" or "object information", we always refer to information that is derived from 2-D image data.

Also, it appears that the object representation approaches that have been used in the context of face detection/recognition have been evaluated most thoroughly. Consequently, most of the object representation approaches that we will consider here were used in the context of face recognition.

Generally, two main types of 2-D object representations appear to exist: *Feature-based representations* and *template-based representations* [Brunelli and Poggio, 1993].

The feature-based approaches as used e.g. in [Cox *et al.*, 1996; Govindaraju, 1996; Xi *et al.*, 1994; Chellappa *et al.*, 1995; Hong, 1991; Nakamura *et al.*, 1991; Yuille, 1991; Zhang *et al.*, 1998; Lyons and Akamatsu, 1998; Delagnes *et al.*, 1995; Denzler and Niemann, 1996; Jaquin and Eleftheriadis, 1995; Blake and Isard, 1998] describe objects through abstraction: An object is represented as a selected collection of abstract features. Simple abstract features are e.g. edges, lines, line segments and points. More complex features may be composed from several simple ones. Also, they can be local gray-value patterns, and even Gabor wavelet jets [Wiskott *et al.*, 1997] can be used as features. This type of representation leads to an abstraction from the image pixel values. In most feature-based approaches, the selection of features as well as their description is given *a priori* through heuristics.

The major drawbacks of abstract object representations are known. Without further explanation, they are the following:

[†] In the following, the term "cognitive system" will always refer to artificial systems.

- Single abstract features are ambiguous. They are not able to uniquely identify in an image the structure they are describing. Therefore, a collection of features always needs to be considered so that their topology adds further important model knowledge.

- Object representation solely through abstract features leads to a loss of valuable image data.

- The choice and description of features are heuristic. Success or failure of a task closely correlates with *a priori* knowledge for the choice of features.

On the other hand, template-based representations, like [Jolliffe, 1986; Loeve, 1955; Costen *et al.*, 1996; Sirovitch and Kirby, 1987; Kirby and Sirovich, 1990; Edelman *et al.*, 1992; Craw *et al.*, 1999; Moghaddam and Pentland, 1997; Turk and Pentland, 1991; Rowley *et al.*, 1998; Yang and Huang, 1994; Matas *et al.*, 1999] are completely data-driven. The template-based representation uses in its simplest version a gray-value template of the object. But sophisticated variations like PCA-related approaches [Jolliffe, 1986; Loeve, 1955] also exist. In contrast to feature-based representations, a template-based representation is a holistic representation, where the object is treated as a whole. Prior knowledge is needed here only for segmentation of the object from the background. A rudimentary segmentation may result in significant instabilities with respect to background variations.

The major known drawbacks of template-based representations relative to abstract representations are the following:

- Geometrical deformations of abstract object information are relatively easy to handle, but the problem of aligning template and image into a common coordinate system appears to be a major problem for template-based approaches.

- Abstract representations are mostly robust with respect geometric deformations or illumination, contrast and background changes, but these changes lead to great instabilities in template-based approaches.

- On the other hand, the feature-based approach can adapt the number of features used to the needs of the problem, but the holistic approach prohibits this to a certain degree.

To summarize, abstraction from pixel gray-values introduces valuable robustness with respect to illumination variations, etc., but valuable image information is lost. On the other hand, relying on the pixel information only while preserving the image data leads to serious instabilities.

An object representation that combines feature-based and template-based characteristics is the approach of [Wiskott *et al.*, 1997], where the features are selected *a priori*, while the description of each feature is subject to training.

We will introduce an object representation that can be described as both feature-based and template-based. The representation is feature-based because an object is represented as a collection of features and their relative positions. The only information that is stored in this representation are these features. The features are found through optimization; no prior knowledge is used. The representation is template-based because one can work with the representation as a template. The collection of features allows complete reconstruction. Almost no image information is lost. However, since the template is composed from the set of features, the number of features used allows the precision of the template to be controlled, ranging from a coarse representation to an almost photo-realistic one.

To be specific, we want to introduce *Gabor Wavelet Networks* (GWNs) as a 2-D object representation framework. GWNs are feature-based because an object is represented as a collection of specially parameterized and weighted Gabor wavelets. The only information that is stored in this representation is the parameter vectors of each Gabor wavelet.

GWNs are template-based because the represented object can be completely reconstructed by the weighted sum of the wavelets. In this sense, the object can literally be viewed as the sum of its features. The use of Gabor wavelets introduces the model for the object features. Furthermore, the GWN framework supplies all the algorithms needed to cope with illumination change, affine deformation, segmentation and alignment of the representation to an object in an image.

GWNs combine the advantages of both representations:

- The robustness with respect to geometric deformations and illumination changes is inherited from the feature-based representation.
- GWNs are robust with respect to ambiguities of single features because GWNs inherit the holistic view of an object from the template-based approach.
- The loss of valuable image data is avoided, except for the mean gray value and the contrast, which are normalized.
- The number of features that are used to describe an object can be adapted according to need. This also allows the precision of the description to vary between coarse and photo-realistic.

In the following, we will discuss various approaches to object representation. In Section 1.2 we will summarize the contribution of the thesis and in Section 1.3 we will give an outline of the thesis' structure.

1.1 Related Work

In this section we will present the major and most recent publications that present approaches to 2-D object representation. We should mention that there exist a great variety of object representations, but we shall discuss only those that are precise enough to allow the recognition of individual objects or of object classes.

1.1.1 Template-Based Approaches

One of the most successful approaches to template-based 2-D object representation is based on Principal Component Analysis (PCA) [Jolliffe, 1986; Loeve, 1955], such as the eigenface approach [Turk and Pentland, 1991] and various enhancements [Moghaddam and Pentland, 1997; Zhao *et al.*, 1998]. The eigenface approach has shown its advantages in the context of detection [Sung and Poggio, 1994] and recognition [Phillips *et al.*, 1998]. Its major drawbacks are its sensitivity to perspective deformations and illumination changes [Belhumeur *et al.*, 1997; Craw *et al.*, 1999]. PCA approximates texture only; geometrical information is not evaluated. Furthermore, the alignment of object images into a common coordinate system is still a problem. Another PCA-based approach is the active appearance model (AAM) [Cootes *et al.*, 1998]. This approach enhances the eigenface approach considerably by including geometrical information. This allows alignment of image data into a common coordinate system; the alignment technique can be elegantly formulated within the AAM framework. Recognition and tracking applications have also been done within this framework [Edwards *et al.*, 1998]. An advantage of this approach is the ability to model, in a photo-realistic way, almost any face, gesture and gender. However, this is an expensive task. In fact, use of varying precision levels in order to spare computational resources and to restrict consideration to the data actually needed for a certain application seems not be easy. Generally, eigenface approaches encode information on a pixel basis. This is also true for the active appearance approach, but a further level of abstraction is achieved via the appearance parameters. The papers [Rowley *et al.*, 1998; Poggio and Beymer, 1995] represent other template-based approaches, where object representations are found implicitly through application of artificial neural networks (ANNs). The inputs to the ANNs are subsampled gray-value images of the object or object class. In [Yang and Huang, 1994; Matas *et al.*, 1999], templates and subsampled versions of the templates are directly used,

and the authors optimize their correlation approaches by using, e.g., geometric knowledge.

1.1.2 Feature-Based Approaches

In [Cox *et al.*, 1996; Govindaraju, 1996; Reisfeld and Yeshurun, 1998; Yow and Cipolla, 1997], a feature-based representation for face detection is introduced. Face knowledge is represented through rudimentary line descriptions and an explicit description of their relations. Features are detected in an image through spatial filters, and filter responses are grouped according to geometric and gray-value constrains. Probabilistic frameworks are used to reinforce probabilities and to evaluate the likelihood that the candidate is a certain object. In [Herpers and Sommer, 1998; Lam and Yan, 1996; Xi *et al.*, 1994], single object features are explicitly modeled through static line models. In [Blake and Isard, 1998], the well-known active contour models are presented. In [Zhang *et al.*, 1998], differences between geometric features and wavelet features are discussed.

1.1.3 The Bunch Graph Approach

The bunch graph approach [Wiskott *et al.*, 1997], as mentioned above, combines characteristics of feature-based and template-based representations. The approach is based on the discrete wavelet transform: A set of Gabor wavelets is applied at a set of hand-selected prominent object points, so that each point is represented by a set of filter responses, called a *jet*.

During a training phase, jets are computed on a set of training images, but always for the same set of prominent object points. After the training phase, an entire set of jets exists for each prominent object point. The sets of jets, together with their relative positions, define a *bunch graph*.

In order to represent a novel image, the bunch graph first searches automatically for the prominent object points by using the set of stored jets for each point. At the detected object features, new jets are computed and added to the bunch graph.

For recognition of an object, the best-matching jets are selected from the bunch graph and a voting strategy is used for final identification.

1.1.4 Histogram-Based Approaches

Histogram-based approaches are presented in [Funt *et al.*, 1998; Schiele and Crowley, 2000; Swain and Ballard, 1991]. In these approaches objects are described and characterized by vectors of local feature measurements, such as color, Gaussian derivatives, etc. Multidimensional

histograms are used to approximate the probability density function for local and global appearance. Histograms discard geometric information. Furthermore, foreground-background segmentation is not possible with this representation, and the approach is not very robust with respect to background variations when the segmentation is not done properly.

1.2 Contribution

This dissertation presents GWNs as an object representation approach that is both feature-based and template-based. Even though Wavelet Networks were already introduced by [Zhang and Benveniste, 1992], they have hardly been used or even mentioned in active research. We will argue that the potential of GWNs has been underestimated. This thesis contributes a thorough evaluation of their properties as well as their advantages and disadvantages for real applications.

In detail, GWNs supply a representational and algorithmic framework for the design of typical appearance-based visual applications that are used, e.g., in Human-Computer-Interaction (HCI), such as face tracking, face recognition and gaze detection. For these three tasks the framework offers a unified approach to

- the representation of image data and
- the formulation of algorithms.

The algorithmic framework is strictly 2-D appearance-based. This means that model and template knowledge, which is represented by the GWNs, is evaluated on the basis of object appearance in a 2-D gray-value image, and the algorithms do not rely on any hand-selected model knowledge.

In detail, we will show that

1. GWNs offer both a template-based and a feature-based object representation,
2. GWNs are able to cope with affine object deformations and with changes in illumination and contrast,
3. the algorithmic framework for alignment of the object representation with an object image is inherent in the representation,
4. the representation is sparse and efficient.

In various experiments we will further demonstrate the following:

1. GWNs can be used efficiently for face tracking. Face tracking will be realized in an appearance-based manner while the GWN framework offers the needed algorithmic basis for affine tracking through so-called *superwavelets*.

2. GWNs allow an object to be represented with any desired precision, from coarse to almost photo-realistic. This will allow computation speed and the computational precision to be controled. We will introduce the term *progressive attention* for the variability in perception precision. The same term has been introduced by [Zabrodsky and Peleg, 1990] for image compression.

3. GWNs offer a sparse representation of image data. The sparseness is achieved because the Gabor wavelets introduce a model for local image features. Objects that are represented with a GWN can be considered as weighted collections of local image features. Data reductions of up to 98% can be achieved through this representation.

4. The sparseness of the representation and the use of a model for the local image features lead to very specific representations of objects where the optimized parameters of each of the Gabor wavelets reflect the structure of the represented object. Through this specificity, accurate recognition can be realized. Algorithms for recognition are natural parts of the GWN framework. Without any further heuristics, the recognition rate is as high as 97% for a small database with large facial expression variations.

5. GWNs represent object data through a weighted sum of specially parameterized Gabor wavelets. We will show that the weights and the filter responses are linearly related. The Gabor wavelets can be used not only for reconstruction, but also for image filtering. The linear relation implies that the Gabor wavelets are not only optimized for reconstruction, but also for filtering. GWNs therefore offer an optimized scheme for filtering images, i.e. for optimized extraction of image data through a small set of Gabor filters that are given by a GWN.

6. We will present an appearance-based pose estimation approach in which input images are filtered with optimized filters. The reduced set of filters speeds up computation and training time of a subsequently applied ANNs and the pose estimation results are excellent.

1.3 Thesis Outline

Chapter 2 gives a background and an introduction to GWNs. Section 2.1 starts with a general introduction to wavelets and related terms. We then give a general introduction to wavelet

networks (Subsection 2.1.5) followed by a general introduction to Gabor filters (Subsection 2.1.6). Finally, Section 2.2 gives an extensive introduction to GWNs and to important constraints on the Gabor functions used. The remaining sections of Chapter 2 will discuss how GWNs are optimized (Subsection 2.3), how weights are computed (Subsection 2.4), and how different GWNs can be compared (Section 2.5). Furthermore, distance measures within wavelet space will be discussed (Subsection 2.5.2). Also, the term *superwavelet* will be introduced; it will allow any affine deformation of image data that is represented by a GWN (Section 2.6).

In Chapter 3 we will discuss the most important properties of GWNs, including (1) the relation between a GWN and the represented object (Section 3.1), (2) the property of variable representation precision of the GWN, from coarse to almost photo-realistic (Section 3.2), and (3) the property of optimized filtering through the linear relation between wavelet weights and filter responses (Section 3.3).

Chapters 2 and 3 will cover all be important topics and properties of GWNs that have been investigated and that are needed for the applications described in the thesis.

Chapters 4 through 6 discuss how appearance-based tracking (Chapter 4), appearance-based face recognition (Chapter 5), and appearance-based pose estimation (Chapter 6) can be realized with techniques that are part of the GWN framework. With minor exceptions, these chapters are independent of each other and depend solely on Chapters 2 and 3.

In Chapter 4 we first introduce the *progressive attention scheme*. We then will exploit this scheme and show that progressive attention allows control of tracking speed and tracking precision. Progressive attention and affine tracking with superwavelets will be treated thoroughly in various experiments.

In Chapter 5 we introduce a novel face recognition approach that exploits the sparseness of the GWN representation. In experiments we show how faces can be recognized in spite of affine deformations, variations in facial expression, and illumination.

In Chapter 6 we exploit the optimized filtering scheme that is offered by the GWN for pose estimation. The estimation results achieved are better than the results of any other pose estimation approach that is known to us. The approach again exploits techniques that are part of the GWN framework. The speed is high and can be controlled through the progressive attention property.

Finally, Chapter 7 closes with a summary of the main topics and a discussion of further issues that remain to be investigated.

The existing literature on image representation and on tracking, recognition and pose estimation is large and varied. Reviews of the literature will consequently be spread throughout the dissertation. Chapter 2 contains the relevant citations for wavelets, wavelet networks and re-

lated topics. Chapters 4 through 6 contain, each in one of its early sections, a report of relevant background and related work, with selected citations and related topics.

Each experimental chapter is concluded with final remarks about the applications.

Chapter 2

Introduction to Gabor Wavelet Networks

In this chapter we will give an extensive and thorough introduction to Gabor Wavelet Networks. The actual introduction will start in Section 2.2 and will include a discussion of

- the relationship between filter responses and wavelet coefficients,
- distance measurements between
  - different wavelet networks and
  - different sets of wavelet coefficients, derived from the same wavelet network,

  as well as
- different norms.

First, however, we will start with an introduction to the wavelet transform itself from which GWNs are derived. This will be done in Section 2.1 to the extent needed in order to introduce GWNs.

2.1 Foundations

In this section we give a short introduction to the wavelet transform. The wavelet transform is often referred to as the *wavelet decomposition*, thus emphasizing the fact that the wavelet transform decomposes a function into a superposition of wavelets. We will use the term *wavelet transform* hereafter.

We will start with the continuous 1-D wavelet transform and continue with the discrete 1-D wavelet transform for orthogonal and non-orthogonal frames. Then we will extend the 1-D

transform to the 2-D wavelet transform. After that, we will review very briefly the two wavelet approaches most commonly taken today to image processing and image representation.

2.1.1 The 1-D Continuous Wavelet Transform

In this subsection we follow mainly the notation of [Daubechies, 1992].

A function $\psi \in \mathbb{L}^2(\mathbb{R})$ that satisfies

$$0 < C_\psi = 2\pi \int_{\mathbb{R}} \frac{\|\bar{\psi}(\omega)\|^2}{\|\omega\|} d\omega < \infty \tag{2.1}$$

is called an *admissible wavelet*. Here $\bar{\psi} = \frac{1}{\sqrt{2\pi}} \int e^{j\omega t}\psi(t)dt$ is the Fourier transform of ψ. Equation (2.1) is often referred to as the *admissibility condition*. In the following, when we use the term *wavelet* we assume that the wavelet is admissible. The admissibility condition ensures that the Fourier transform of ψ decays sufficiently quickly when approaching zero. The admissibility condition is important for the derivation of the *resolution of identity* formula (2.3).

For any function $f \in \mathbb{L}^2(\mathbb{R})$ the continuous 1-D wavelet transform is given by

$$\begin{aligned} (L_\psi f)(a,b) &= \frac{1}{\sqrt{|a|}} \int_{\mathbb{R}} f(t)\psi\left(\frac{t-b}{a}\right) dt \\ &= \langle \psi_{a,b} \ , \ f \rangle \ , \end{aligned} \tag{2.2}$$

with $a \in \mathbb{R} \setminus \{0\}$, $b \in \mathbb{R}$, $\psi_{a,b}(t) = \frac{1}{\sqrt{|a|}}\psi\left(\frac{t-b}{a}\right)$ and $\langle \cdot \ , \ \cdot \rangle$ denoting the $\mathbb{L}^2(\mathbb{R})$-inner product. The function ψ is often called the *mother wavelet* and the functions $\psi_{a,b}$ are called *wavelets*.

The corresponding inverse wavelet transform, *resolution of identity* formula, or Calderòn equation that reconstructs a function f from its wavelet coefficients is [Calderón, 1964; Grossmann and Morlet, 1984]

$$\begin{aligned} f(t) &= \frac{1}{C_\psi} \int_{\mathbb{R}} \int_{\mathbb{R}} (L_\psi f)(a,b) \frac{1}{\sqrt{|a|}} \psi\left(\frac{t-b}{a}\right) \frac{dadb}{a^2} \\ &= \frac{1}{C_\psi} \int_{\mathbb{R}^+} \int_{\mathbb{R}} \langle \psi_{a,b} \ , \ f \rangle \ \psi_{a,b}(t) \ \frac{dadb}{a^2} \ . \end{aligned} \tag{2.3}$$

Eq. 2.3 was first proved in 1964 [Calderón, 1964]. The integration with respect to a and b is done over the entire *continuous phase space*. The parameters a, b are continuous over $\mathbb{R}$ and control the dilation and translation of the mother wavelet function ψ. The term *phase space* is borrowed from physics and refers to two-dimensional time-frequency space, considered as a geometric whole [Daubechies, 1990]. It can be seen that when the integral in eq. (2.1) diverges the function f cannot be reconstructed. For details of the proof see, e.g., [Calderón, 1964; Daubechies, 1992; Louis *et al.*, 1994].

Equation (2.3) can be interpreted in two ways: It shows

1. that a function f can be uniquely represented in terms of its wavelet coefficients $(L_\psi f)(a,b)$ and that there is a one-to-one correspondence between functions $f \in \mathbb{L}^2(\mathbb{R})$ and vectors in the infinite-dimensional vector space over the wavelets $\psi_{a,b} = \psi\left(\frac{t-b}{a}\right)$.

2. that a function f can be written as a superposition of the wavelets $\psi_{a,b}$.

2.1.2 The 1-D Discrete Wavelet Transform

It is known that the representation $(L_\psi f)(a,b)$ of eq. (2.2) is highly redundant and that the continuous phase space can be discretized without loss of information [Chui, 1992; Daubechies, 1992; Mallat, 1998]. In this sense, let $S \subset \mathbb{R} \setminus \{0\} \times \mathbb{R}$ be a discrete set. Then, $B_\psi = \{\psi_{m,n} \mid (m,n) \in S\}$ defines a *discrete family of wavelets*. The set S can be understood as a (not necessarily homogeneous) sampling grid of the phase space.

Using the family of wavelets B_ψ, the wavelet coefficients $(L_\psi f)(m,n) = \langle \psi_{m,n} \ , \ f \rangle$ for $(n,m) \in S$ are calculated by applying eq. (2.2). In eq. (2.3), the double integral is then replaced by a double sum. However, there does not exist a direct discrete version of (2.3). Hence before we can write a function f in terms of its discrete wavelet coefficients $(L_\psi f)(m,n)$, we have to introduce some more concepts.

Obviously, for a given wavelet function ψ, how well a function $f \in \mathbb{L}^2(\mathbb{R})$ can be represented by its discrete wavelet coefficients $(L_\psi f)(m,n)$, $(m,n) \in S$, depends on the sampling grid S or, equivalently, on the discrete family of wavelets B_ψ. In order to quantify this, the term *frame* needs to be introduced. It is usually defined in a more general manner [Daubechies, 1992], but we define it here according to our needs:

Definition 1
*Let $\psi \in \mathbb{L}^2(\mathbb{R})$ be a wavelet, S a sampling grid, and $B_\psi = \{\psi_{m,n} \mid (m,n) \in S\}$ a discrete family of wavelets. We say that B_ψ constitutes a **frame** if there exist constants $A > 0$ and $B < \infty$ such that for all $f \in \mathbb{L}^2(\mathbb{R})$*

$$A\|f\|^2_{\mathbb{L}^2} \leq \ \sum_{(m,n)\in S} |\langle \psi_{m,n} \ , \ f \rangle_{\mathbb{L}^2}|^2 \ \leq B\|f\|^2_{\mathbb{L}^2} \ , \tag{2.4}$$

*where $\|f\|^2_{\mathbb{L}^2} = \int_{-\infty}^{\infty} |f(x)|^2 dx$ (which is referred to as the **energy** of f). A and B are called **frame bounds**.*

When a discrete family of wavelets forms a frame, it provides a complete and lossless representation of *any* function $f \in \mathbb{L}^2$ [Daubechies, 1990].

In order to provide more detail, we introduce some additional terms: B_ψ is called *orthonormal* in $\mathbb{L}^2(\mathbb{R})$ if for $\psi_i, \psi_j \in B_\psi$

$$\langle \psi_i \ , \ \psi_j \rangle_{\mathbb{L}^2} = \delta_{i,j} = \begin{cases} 1, & \text{if } i = j \\ 0, & \text{if } i \neq j \ . \end{cases}$$

A frame B_ψ is called a *basis* for $\mathbb{L}^2(\mathbb{R})$ if for all $f \in \mathbb{L}^2(\mathbb{R})$ the linear combination $f = \sum_k c_k(f)\psi_k$ is unique. A family of functions in $\mathbb{L}^2(\mathbb{R})$ that is both orthogonal and a basis is called an *orthogonal basis*.

The expression $\frac{A+B}{2}$ measures the redundancy of the frame while $\frac{B}{A}$ measures its tightness [Lee, 1996]. A frame is called *tight* when $B = A$.

For frame bounds $A = B = 1$ and $\|\psi_i\| = 1$, the family of functions B_ψ forms an orthogonal basis of $\mathbb{L}^2(\mathbb{R})$, and any function $f \in \mathbb{L}^2(\mathbb{R})$, can be uniquely written as

$$\begin{aligned} f(t) &= \frac{2}{A+B} \sum_{(m,n)\in S} (L_\psi f)(m,n)\psi\left(\frac{t-n}{m}\right) \\ &= \frac{2}{A+B} \sum_{(m,n)\in S} \langle \psi_{m,n} \ , \ f \rangle \psi_{m,n}(t) \end{aligned} \tag{2.5}$$

Even for frame bounds $0 < A \leq B < 2$, B_ψ can still be considered to be an orthogonal frame and eq. (2.5) is fairly exact. For frame bounds $A = B > 1$, eq. (2.5) is exact, but B_ψ no longer constitutes a basis, so that the linear combination in eq. (2.5) may not be unique. In cases where B_ψ does not constitute a tight frame, i.e. $A < B$, we have to write f in terms of the *dual frame* $\tilde{B}_\psi = \{\tilde{\psi}_{m,n} \mid (m,n) \in S\}$:

$$\begin{aligned} f(t) &= \sum_{(m,n)\in S} \langle \tilde{\psi}_{m,n} \ , \ f \rangle \ \psi_{m,n}(t) \\ &= \sum_{(m,n)\in S} \langle \psi_{m,n} \ , \ f \rangle \ \tilde{\psi}_{m,n}(t) \ . \end{aligned} \tag{2.6}$$

The two families of functions B_ψ and $\tilde{B}_\psi$ are called dual when for each $\psi_i \in B_\psi$ and $\tilde{\psi}_j \in \tilde{B}_\psi$ we have

$$\langle \psi_i \ , \ \tilde{\psi}_j \rangle_{\mathbb{L}^2} = \delta_{i,j} = \begin{cases} 1, & \text{if } i = j \\ 0, & \text{if } i \neq j \ . \end{cases} \tag{2.7}$$

For (infinitely) large sets S, the dual wavelets in eq. (2.6) can be computed only approximately [Daubechies, 1990].

It should be kept in mind that with a frame we can reconstruct *any* function $f \in \mathbb{L}^2(\mathbb{R})$. In this thesis, we are not interested in finding a wavelet representation for *every* function $f \in \mathbb{L}^2(\mathbb{R})$. Instead, we will deal with only a small subset of $\mathbb{L}^2(\mathbb{R})$, so that we will not actually have to deal with wavelet families that constitute frames. But eq. (2.6) still holds for non-frame wavelet families and allows to approximately reconstruct a function f with minimal error with respect to the $\mathbb{L}^2$ Norm (i.e. in the mean square sense). We will return to this important issue later.

It may be mentioned that for the discrete wavelet transform, the function f and the wavelet ψ are *continuous* functions. It is the phase space that is discrete here. This becomes clearer when we look at eqs. (2.2) and (2.5). In eq. (2.2) the integral remains discrete because the wavelet coefficient is calculated by integration over the *continuous* function parameters *at discrete* phase space coordinates. Consequently, in eqs. (2.5) and (2.6), the function f is written as a sum over all the discrete phase space coordinates $(m, n) \in S$.

In *multi-resolution signal analysis* or *multi-frequency channel decomposition*, as discussed in [Mallat, 1989b; Mallat, 1989a; Grossmann and Morlet, 1984; Lee, 1996; Michaelis, 1997], one exploits the properties of the discrete wavelet transform to analyze signals in a scale-pyramid like fashion. For this, the phase space is usually sampled with a "wavelet-like" grid, where the support of ψ is essentially proportional to a_0^m (see Fig. 2.1):

$$S = \{(nb_0a_0^m, a_0^{-m}k_0) \mid m, n \in \mathbb{Z}\} \subset \mathbb{R} \setminus \{0\} \times \mathbb{R} \tag{2.8}$$

$$B = \{\psi_{m,n}^{a_0,b_0}(x) = a_0^{-m/2}\psi(a_0^{-m}x - nb_0) \mid m, n \in \mathbb{Z}\} \tag{2.9}$$

where $a_0 > 1$, $b_0 > 0$ and $k_0 = \int_{0 \leq \pm k < \infty} |\bar{\psi}(k)|^2 \frac{dk}{k}$. The choice of a_0 and b_0 is directly related to the choice of the mother wavelet ψ: For multi-resolution signal analysis, the dilation step a_0 and translation step width b_0 are usually chosen to be 2 and 1, respectively, while the wavelet ψ is often chosen such that ψ is well localized in both the spatial and the Fourier domain [Kronland-Martinet *et al.*, 1987; Meyer, 1992] and such that B constitutes an orthonormal basis. What is exploited here is essentially the fact that the support of $\psi_{m,n}$ is proportional to a_0^m. As a consequence, high-frequency wavelets $\psi_{m,n}$, with $m << 0$, are greatly concentrated and involve a very small time translation step $b_0a_0^m$ which is also proportional to a_0^m. This means that the wavelet transform is able to "zoom in" on the signal data by using more and more concentrated wavelets $\psi_{m,n}$.

In contrast to *first* choosing a_0 and b_0 and *then* the mother wavelet, as above, one might be interested in choosing the mother wavelet ψ *first* and *then* finding the parameters a_0 and b_0. This allows one to investigate, as done by [Lee, 1996], how the phase space should be sampled in order to achieve a frame.

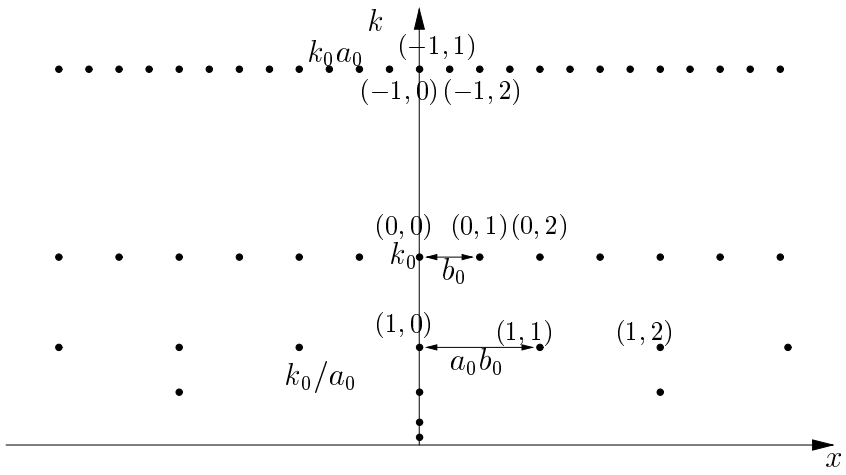


Figure 2.1. Phase space sampling scheme corresponding to the (discrete) wavelet transform. The constant k_0 is given by $k_0 = \int_0^\infty |\bar{\psi}(k)|^2 \frac{dk}{k}$; ψ was chosen to be even and we have chosen $a_0 = 2$ [Daubechies, 1990].

2.1.3 The 2-D Continuous Wavelet Transform

It is possible to extend the 1-D continuous wavelet transform to 2-D. For this, we need to introduce rotation θ in addition to dilation a and 2-D translation $\mathbf{b}$. For $\psi \in \mathbb{L}^2(\mathbb{R}^2)$, the admissibility condition (2.1) becomes [Daubechies, 1992]

$$0 < C_\psi = 4\pi^2 \int_0^\infty \int_0^{2\pi} \frac{|\bar{\psi}(\omega \cos\theta \ , \ \omega \sin\theta)|^2}{|\omega|} \, d\theta \, d\omega < \infty \ . \tag{2.10}$$

With

$$\psi_{a,\mathbf{b},\theta}(\mathbf{x}) = \frac{1}{a} \psi \left(\mathbf{R}_\theta \left(\frac{\mathbf{x} - \mathbf{b}}{a} \right) \right) \ ,$$

where $a > 0$, $\mathbf{b} \in \mathbb{R}^2$ and

$$\mathbf{R}_\theta = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \ ,$$

the resolution of identity (2.3) then becomes

$$f \ = \ \frac{1}{C_\psi} \int_0^\infty \int_0^\infty \int_0^{2\pi} (L_\psi f)(a, \mathbf{b}, \theta) \frac{1}{a^3} \psi_{a,\mathbf{b},\theta} \, d\theta \, da \, d\mathbf{b} \tag{2.11}$$

Note that the dilation parameter a is the same for both dimensions. This, however, can be relaxed for functions $f \in \mathbb{L}^2(\mathbb{R}^2)$ that are separable in every coordinate [Zhang and Benveniste, 1992]:

$$f(\mathbf{x}) = f_1(x_1) \times f_2(x_2) \ .$$

For such functions each component is handled separately in the integral, so that for any such function $f \in \mathbb{L}^2(\mathbb{R}^2)$ the continuous 2-D wavelet transform is given by

$$\begin{aligned}(Lf)(\mathbf{c},\mathbf{s},\theta) &= \int_{\mathbb{R}^2} f(\mathbf{x})\psi(\mathbf{SR}(\mathbf{x}-\mathbf{c}))d\mathbf{x} \\ &= \langle f, \psi_{\mathbf{c},\theta,\mathbf{s}}\rangle \\ &= \langle f, \psi_{\mathbf{n}}\rangle \ , \end{aligned} \tag{2.12}$$

with the rotation matrix $\mathbf{R}$, the dilation matrix $\mathbf{S}$, and the translation vector $\mathbf{c}$:

$$\begin{aligned}\mathbf{R} &= \begin{pmatrix}\cos\theta & -\sin\theta \\ \sin\theta & \cos\theta\end{pmatrix} \\ \mathbf{S} &= \text{diag}(s_x, s_y) \\ \mathbf{c} &= (c_x, c_y)^T \ .\end{aligned}$$

Here θ denotes the rotation angle of the wavelet $\psi(\mathbf{x})$, s_x, s_y the scalings in the x and y directions, and c_x, c_y the translations in the x and y directions. In this sense, the wavelets $\psi_{\mathbf{n}}$ are dilated, rotated and translated versions of the mother wavelet ψ. The five-dimensional parameter vector $\mathbf{n}$ is given by these parameters:

$$\mathbf{n} = (c_x, c_y, \theta, s_x, s_y) \ .$$

The function f can always be reconstructed by integration over all wavelet parameters:

$$\begin{aligned}f &= C_\psi^{-1} \int \left(Lf\right)_{\mathbf{n}} \psi_{\mathbf{n}} \, \frac{d\mathbf{n}}{|s_x s_y|} \\ &= C_\psi^{-1} \int \langle f, \psi_{\mathbf{n}}\rangle \psi_{\mathbf{n}} \, \frac{d\mathbf{n}}{|s_x s_y|} \ .\end{aligned}$$

2.1.4 The 2-D Discrete Wavelet Transform

A natural way to define the discrete wavelet transform is to discretize the phase space and to assign discrete values to the wavelet parameters as follows [Lee, 1996]: $s_x = (s_{x_0})^m$, $s_y =$

$(s_{y_0})^m, c_x = ns_0(s_{x_0})^m, c_y = ks_0(s_{y_0})^m, \theta = \theta_l = l\theta_0$, with $m, n, k, l \in \mathbb{Z}$. The discrete wavelet transform is then given by

$$(L^d_\psi f)(m, n, k, l) = \langle f, \psi_{mnkl} \rangle \ . \tag{2.13}$$

Equation (2.13) can be interpreted as an abstract representation of f by its wavelet coefficients. To represent f uniquely (if it is possible at all), huge numbers of wavelet coefficients are generally needed. How well f is represented by its coefficients $(L^d_\psi f)(m, n, k, l)$ and how many are needed depends on the chosen wavelet and on the values s_{x_0}, s_{y_0}, s_0 and θ_0.

The 2-D discrete wavelet transform is also used in the bunch graph approach [Wiskott *et al.*, 1997], where, however, only a few prominent feature points are represented by their wavelet coefficients. Of course, only a limited reconstruction of the image is possible in this case. The equation

$$f = \sum_{mnkl} w_{mnkl} \psi_{mnkl} \ . \tag{2.14}$$

allows two interpretations:

1. Given the wavelets ψ_{mnkl}, an image f can be *represented* by the set of weights w_{mnkl}. Understanding each wavelet ψ_{mnkl} as a feature of f, the weights w_{mnkl} give the "importance" of ψ_{mnkl} in the description of f.

2. The function f is *approximated* by a linear combination of weighted wavelets. Eq. (2.14) therefore defines a template for f, with approximation quality as an additional degree of freedom.

In [Wiskott *et al.*, 1997], the *representational* aspect of eq. (2.14) is emphasized in the sense that the goal was to represent individual properties of faces. In [Zhang and Benveniste, 1992] and [Szu *et al.*, 1992], the main interest was in function approximation, and eq. (2.14) is interpreted as an *approximation*.

2.1.5 Wavelet Networks

Wavelet Networks were first introduced by [Zhang and Benveniste, 1992] as a combination of feed-forward neural networks, namely the multi-layer sigmoid network and the wavelet decomposition. Multi-layer networks allow representation of non-linear functional mappings between the input and output variables. This is done by representing a multivariate non-linear

function in terms of a composition of non-linear functions of a single variable, called *activation functions* [Bishop, 1995]. Sigmoids $\sigma(x)$ are often applied as activation functions. The corresponding mapping functions then look like

$$g(\mathbf{x}) \quad = \quad \sum_{i=1}^{M} w_i \sigma(\mathbf{a}_i^T \mathbf{x} + b_i) + w_0 \ . \tag{2.15}$$

Here w_0 is called a bias and refers to a constant offset. The parameters $\mathbf{a}_i$ and b_i apply a linear projection to the input vector $\mathbf{x}$. These projections are then transformed by the non-linear activation functions σ which in turn are combined linearly to form the output g. It was shown [Poggio and Girosi, 1990] that finite sums of the form (2.15) exhibit the *universal approximation* property, i.e. they are dense in the space of continuous functions.

[Zhang and Benveniste, 1992] replace the sigmoid in (2.15) by an admissible wavelet ϕ (see Fig. 2.2), and argue that the resulting wavelet networks

- preserve the *universal approximation* property, i.e. provide the same approximation capability as feed-forward neural networks,
- provide an explicit link between the network coefficients w_i in (2.15) and the coefficients

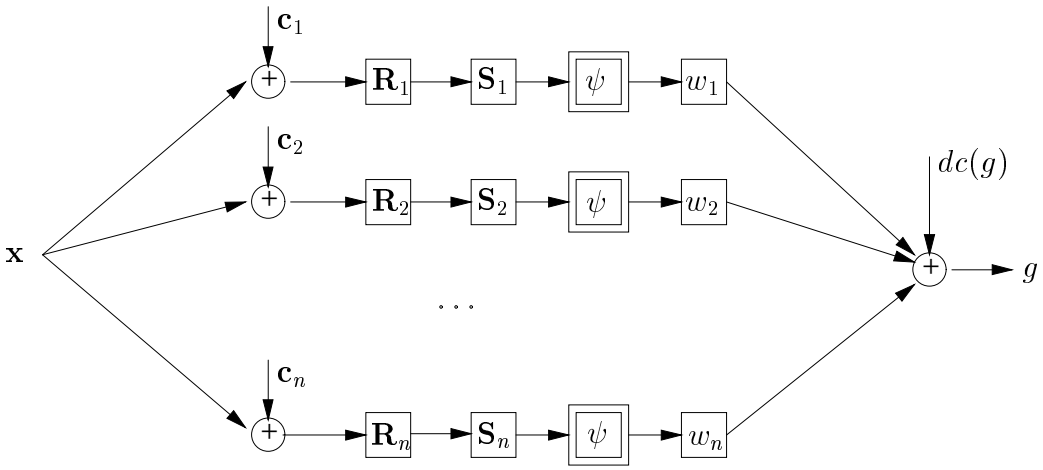


Figure 2.2. This figure shows the structure of a wavelet network. This structure establishes a one-to-one map with eq. (2.15); however, the function σ has been replaced by a 2-D admissible wavelet function ψ. The 1-D translation b has been replaced by the 2-D translation vector $\mathbf{c}$, and rotation and scaling matrices $\mathbf{R}$ and $\mathbf{S}$ are introduced. w_0 is the DC value of the function g that has to be added (if necessary).

of the wavelet transform and the reconstruction (2.5).

- achieve good approximation quality even with a reduced network size.

2.1.6 Gabor Filters

Complex Gabor functions were first introduced by Gabor [Gabor, 1946]. They are complex exponentials with a Gaussian envelope, or Gaussians which are modulated by complex harmonics. In one dimension, their impulse response is given by

$$g_{\sigma,\omega_0}(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2}\frac{x^2}{\sigma^2}\right) \exp\left(j\omega_0 x\right) . \tag{2.16}$$

In two dimensions the mathematical expression of the filter response looks like

$$g_{\boldsymbol{\sigma},\omega_0}(\mathbf{x}) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left(-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2}+\frac{y^2}{\sigma_y^2}\right)\right) \exp\left(j\omega_0(x+y)\right) . \tag{2.17}$$

In the above 2-D equation, the rotation and translation parameters are omitted. The parameters $\boldsymbol{\sigma}$ and θ are chosen beforehand as constants. Dilation, rotation and translation are done through the wavelet parameters in eqs. (2.10) - (2.12).

Eq. (2.17) can be split into an even part g^e and an odd part g^o:

$$g^e_{\boldsymbol{\sigma},\omega_0}(\mathbf{x}) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left(-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2}+\frac{y^2}{\sigma_y^2}\right)\right) \cos\left(\omega_0(x+y)\right) \tag{2.18}$$

$$g^o_{\boldsymbol{\sigma},\omega_0}(\mathbf{x}) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left(-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2}+\frac{y^2}{\sigma_y^2}\right)\right) \sin\left(\omega_0(x+y)\right) \tag{2.19}$$

In Fig. 2.3 plots of a 1-D and a 2-D odd Gabor function are shown.

Gabor functions offer the best localization in both frequency and image space, and they are known to be good feature detectors [du Buf, 1993; Manjunath and Chellappa, 1993; Mehrotra *et al.*, 1992; Michaelis, 1997]. In this thesis we will use odd Gabor functions only, as they have proven to give the best results for the purposes we will use them for. We will discuss this topic in more detail in Section 3.1.

2.2 Introduction to Gabor Wavelet Networks

In this section we propose, as a major contribution of this work, the GWN for image representation. The idea of the wavelet network is inspired by [Zhang and Benveniste, 1992] (see above).

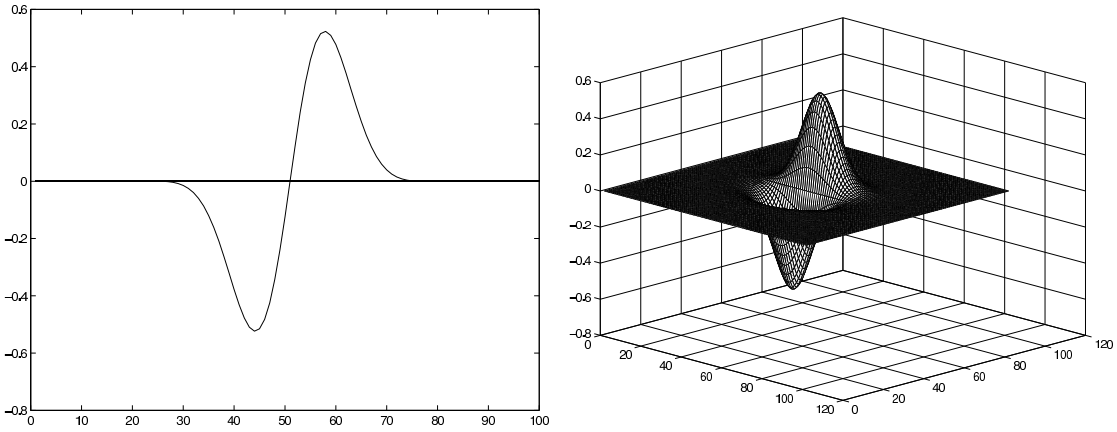


Figure 2.3. Both the odd 1-D (left) and the 2-D (right) Gabor function are shown. The frequency ω_0 is set to 1.

One degree of freedom of wavelet networks results from the choice of the mother wavelet. After several experiments [Pelc, 1997] we chose to use odd Gabor functions for several reasons: The use of Gabor functions in general is inspired by the fact that they provide the best possible tradeoff between spatial resolution and frequency resolution in both 1-D [Gabor, 1946] and 2-D [Daugman, 1985]. Furthermore, the use of Gabor filters in image analysis is biologically motivated, as they model the responses of the receptive fields of the orientation-selective simple cells in the human visual cortex [Daugman, 1985; Jones and Palmer, 1987]. In fact, it has been suggested [Daugman, 1988; Porat and Zeevi, 1988] that the receptive field responses of simple cells can be described by the family of 2-D Gabor wavelets. In addition, Gabor filters are recognized as good feature detectors [du Buf, 1993; Manjunath and Chellappa, 1993; Mehrotra *et al.*, 1992; Michaelis, 1997]. Especially for $\sigma\omega_0 < 2$, they are often used for edge detection [Michaelis, 1997]. Specific uses of the odd Gabor function have particular advantages, which will be discussed in Chapter 3.

An image representation using GWNs has the advantage of being sparser than the Gabor jet representation [Wiskott *et al.*, 1997], but it allows encoding of almost all the image information and leads to good reconstruction.

To define a GWN, we start out, generally speaking, by taking a family of N odd Gabor

wavelet functions $\Psi = \{\psi_{\mathbf{n}_1}, \ldots, \psi_{\mathbf{n}_N}\}$ of the form

$$\begin{aligned}\psi_{\mathbf{n}_i}(x,y) &= \frac{\omega_0}{2\pi\kappa}\exp\left(-\frac{\omega_0^2}{2\kappa^2}\left[\left(s_{x_i}\left((x-c_{x_i})\cos\theta_i+(y-c_{y_i})\sin\theta_i\right)\right)^2\right.\right.\\ &+ \left.\left.\left(s_{y_i}\left(-(x-c_{x_i})\sin\theta_i+(y-c_{y_i})\cos\theta_i\right)\right)^2\right]\right)\\ &\cdot \sin\left(\omega_0 s_{x_i}\left((x-c_{x_i})\cos\theta_i-(y-c_{y_i})\sin\theta_i\right)\right) \quad , \end{aligned} \tag{2.20}$$

with the parameter vector $\mathbf{n}_i = (c_{x_i}, c_{y_i}, \theta_i, s_{x_i}, s_{y_i})^T$. Here c_{x_i}, c_{y_i} denote the translation of the Gabor wavelet, s_{x_i}, s_{y_i} denote the dilations and θ_i denotes the orientation. The parameter ω_0 gives the radial frequency in radians per unit length, and κ is a constant that relates the standard deviation σ to the radial frequency ω_0: $\sigma = \frac{\kappa}{\omega_0}$. According to [Daugman, 1985; Lee, 1996], we define the following constraint:

Constraint 1

The half-amplitude bandwidth of the frequency response is 1 to 1.5 *octaves.*

This means that the relationship between σ and ω_0 is

$$\sigma = \frac{\kappa}{\omega_0} \text{ where } \kappa = \sqrt{2\ln 2}\left(\frac{2^\xi+1}{2^\xi-1}\right) \tag{2.21}$$

with ξ the bandwidth in octaves. For $\xi = 1$ octave, $\sigma \approx \pi/\omega_0$, and for $\xi = 1.5$ octaves, $\sigma \approx 2.5/\omega_0$. This constraint was also used in [Pelc, 1997].

We have set $\omega_0 = 1$, according to [Daugman, 1985; Michaelis, 1997], and $\kappa = \pi$, according to [Daugman, 1985]. With this, eq. (2.20) gives

$$\begin{aligned}\psi_{\mathbf{n}_i}(x,y) &= \frac{2}{\pi^3}\exp\left(-\frac{1}{2\pi^2}\left[\left(s_{x_i}\left((x-c_{x_i})\cos\theta_i+(y-c_{y_i})\sin\theta_i\right)\right)^2\right.\right.\\ &+ \left.\left.\left(s_{y_i}\left(-(x-c_{x_i})\sin\theta_i+(y-c_{y_i})\cos\theta_i\right)\right)^2\right]\right)\\ &\cdot \sin\left(s_{x_i}\left((x-c_{x_i})\cos\theta_i-(y-c_{y_i})\sin\theta_i\right)\right) \quad , \end{aligned} \tag{2.22}$$

The normalization factor is defined so that $\langle\psi,\psi\rangle = 1$, i.e. ψ is normalized with respect to the $\mathbb{L}^2(\mathbb{R}^2)$ norm.

The parameters $\mathbf{n}_i$ (translation, orientation and dilation) of the wavelets can be chosen arbitrarily at the beginning. According to [Zhang and Benveniste, 1992], any function $f \in \mathbb{L}^2(\mathbb{R}^2)$ can be represented by a wavelet network. We are therefore going to interpret the image f as a

function of the space $\mathbb{L}^2(\mathbb{R}^2)$ and assume further, without loss of generality, that f is DC-free. In order to find the GWN for image f we minimize the energy function

$$E = \min_{\mathbf{n}_i, w_i \text{ for all } i} \| f - \sum_i w_i \psi_{\mathbf{n}_i} \|_2^2 \tag{2.23}$$

with respect to the weights w_i and the wavelet parameters $\mathbf{n}_i$. Equation (2.23) says that the w_i and $\mathbf{n}_i$ are optimized (i.e. the translation, dilation and orientation of each wavelet are chosen) so that the image f is optimally approximated by the weighted sum of the Gabor wavelets $\psi_{\mathbf{n}_i}$.

To prevent the wavelets from degenerating during minimization, e.g. to prevent them from stretching out too much, the following important constraint is formulated according to the findings of [Daugman, 1985]:

Constraint 2
The aspect ratio $\frac{s_x}{s_y}$ *of the elliptical Gaussian envelope is at most* $2:1$.

We define a Gabor wavelet network as follows:

Definition 2
Let $\psi_{\mathbf{n}_i}$, $i = 1, \ldots, N$ *be a set of Gabor wavelets,* f *a DC-free image, and let* w_i *and* $\mathbf{n}_i$ *be chosen according to the energy function (2.23). The two vectors*

$$\begin{aligned} \Psi &= (\psi_{\mathbf{n}_1}, \ldots, \psi_{\mathbf{n}_N})^T \text{ and} \\ \mathbf{w} &= (w_1, \ldots, w_N)^T \end{aligned}$$

then define the Gabor Wavelet Network $(\Psi, \mathbf{w})$ *for image* f.

It should be mentioned that it was proposed earlier [Daubechies, 1990; Daugman, 1988; Lee, 1996] to use an energy function (2.23) in order to find the optimal set of weights w_i for a *fixed* set of non-orthogonal wavelets $\psi_{\mathbf{n}_i}$. We modify this approach by also finding the optimal parameters $\mathbf{n}_i$ for each wavelet $\psi_{\mathbf{n}_i}$. The parameters $\mathbf{n}_i$ are chosen from *continuous* phase space and the Gabor wavelets are positioned with sub-pixel accuracy. This is the main advantage over the discrete approach [Daubechies, 1990; Lee, 1996]. While in the case of a discrete phase space, local image structure has to be approximated by a combination of wavelets, a *single* wavelet can be chosen in the continuous case to precisely reflect local image structure. This assures that a maximum amount of image information is encoded. It also leads to an almost symbolic abstraction [Granlund, 1997] of the image data, as we will see later.

Using the optimal wavelets Ψ and weights $\mathbf{w}$ of the GWN of an image f, the GWN allows

Figure 2.4. The left image shows an original face image I, and the right image shows its reconstruction $\hat{I}$ using formula (2.24) with an optimal wavelet network Ψ of just $N = 52$ odd Gabor wavelets, distributed over the inner face region.

an accurate reconstruction $\hat{f}$ of the function f by a linear combination of the weighted wavelets:

$$\begin{aligned} \hat{f} &= \sum_{i=1}^{N} w_i \psi_{\mathbf{n}_i} \\ &= \Psi^T \mathbf{w} \quad . \end{aligned} \tag{2.24}$$

The structure of eq. (2.24) is shown graphically in Fig. 2.2. Of course, the quality of the image representation and the reconstruction depends on the number N of wavelets used. An example reconstruction can be seen in Fig. 2.4. $N = 52$ wavelets are distributed over the inner face region of the left image* I by the minimization formula (2.23). The reconstruction $\hat{I}$ using formula (2.24) is shown in the right image. Note that the Gabor wavelets are continuous functions that interpolate the discrete image they are trained on. This fact will be of great importance later when we need to deform $\hat{I}$ affinely.

*We will generally use the notation $f, g, \ldots$ to refer to band-limited, continuous 1-D or 2-D functions. The dimensionality should be clear from the context. We will use the notation $I, J, \ldots$ when we want to refer explicitly to discrete gray-value images as used in our experiments.

2.3 Optimization of Gabor Wavelet Networks

Minimizing eq. (2.23) is crucial, because finding a global minimum is an inefficient task. In order to find an optimal wavelet family Ψ for the GWN $(\Psi, \mathbf{w})$ for a discrete gray-value image I, we use the Levenberg-Marquardt (LM) method, which is the best known method for non-linear optimization. The LM method allows smooth variations between the inverse Hessian method and the steepest gradient descent method. Far from the minimum, the gradient descent method is used. As the minimum is approached, the Levenberg-Marquardt method smoothly switches to the inverse Hessian method [Press *et al.*, 1986]. The Levenberg-Marquardt method may get stuck in local minima, and a careful selection of the initial parameters is therefore important. This, however, also has the advantage that we can use prior knowledge about significant image features to allow task-oriented optimization.

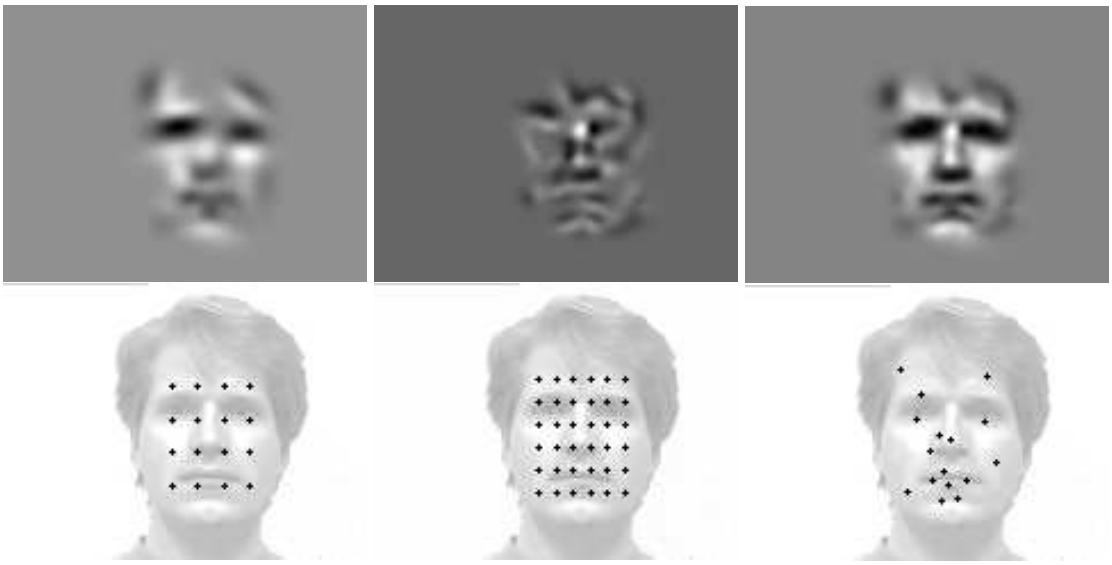
The initialization and optimization scheme we developed is similar to a Laplacian pyramid scheme. First we position 4×4 coarse wavelets equidistantly within the prominent image region (in the case of face representation this is the inner face region) (Fig. 2.5, bottom left). These 16 wavelets define the first pyramid layer. They are then optimized with respect to the energy function (2.23). The optimization result, $\hat{I}_{16}$, is shown in Fig. 2.5, top left. In a second step we calculate the difference between the original image and its reconstruction, $I - \hat{I}_{16}$, which is then approximated by 6×6 finer wavelets (Fig. 2.5, center, bottom). These wavelets form the second pyramid layer. The result is shown in the top center. Adding the two images together yields image $\hat{I}_{52}$ (Fig. 2.5, top right). The positions of the 16 first-layer wavelets after optimization are sketched in Fig. 2.5 bottom right. For comparison refer to the original image I in Fig. 2.4, left. This procedure can be repeated for further pyramid layers. It should be mentioned that at each indicated wavelet position in Fig. 2.5, just a single wavelet is located. The initial orientations are random and the initial scales are constant in each layer, and their values are chosen with respect to the distances to the neighboring wavelets. Intuitively, a coarse-to-fine strategy for optimization makes sense because the energy function (2.23) can be minimized efficiently by first using coarse and then fine wavelets.

In detail, a difference image D is defined as the component-wise (pixel-wise) difference between the original image I and its reconstruction $\hat{I}$:

$$D \;=\; I - \hat{I}\;. \tag{2.25}$$

At the beginning of the optimization, where no wavelets have yet been found, $\hat{I} = 0$ and $D = I$. Weight w_1 is then initially set to 1 and a Gabor wavelet $\psi_{\mathbf{n}_1}$ is selected that minimizes the energy

$$\|D\|_2^2 \;=\; \|I - w_1 \psi_{\mathbf{n}_1}\|_2^2 \tag{2.26}$$



4×4 initialization 6×6 initialization final result

Figure 2.5. The images demonstrate the idea of the Laplace-pyramid-like initialization and optimization scheme. The wavelet net is first initialized with the wavelets sketched in the bottom left image. The optimization result $\hat{I}_{16}$ is shown in the top left image. The difference between that image and the original image is then approximated by the wavelets that are initialized according to the bottom center image. The optimization result is shown in the top center image. Finally, the top right image $\hat{I}_{52}$ shows the sum of the top left and top center image. The bottom right image shows the final positions of the 16 wavelets of image $\hat{I}_{16}$ (left image).

best. In the next step, the weight w_1 is recalculated by orthogonally projecting the image I into the vector space $< \psi_{\mathbf{n}_1} >$ spanned by the single wavelet $\psi_{\mathbf{n}_1}$. In the next section we will go into greater detail about orthogonally projecting an image I into a vector space $< \psi_{\mathbf{n}_1}, \ldots, \psi_{\mathbf{n}_{n-1}} >$ spanned by a family of Gabor wavelets.

Assuming now that we already have a family of wavelets, then

$$D = I - \sum_{i=1}^{n-1} w_i \psi_{\mathbf{n}_i} \ . \tag{2.27}$$

The weights are found through orthogonal projection of the image I into the vector space $< \psi_{\mathbf{n}_1}, \ldots, \psi_{\mathbf{n}_{n-1}} >$. The difference image D is then in the complement of the span of these wavelets:

$$D \in (< \psi_{\mathbf{n}_1}, \ldots, \psi_{\mathbf{n}_{n-1}} >)^{\perp} \tag{2.28}$$

in the space $\mathbb{L}^2(\mathbb{R}^2)$. We then select a new wavelet $\psi_{\mathbf{n}_n}$ such that, with $w_n = 1$,

$$\|D - w_n \psi_{\mathbf{n}_n}\|_2^2 \tag{2.29}$$

is again minimized. This means in particular that

$$\langle D, \psi_{\mathbf{n}_n} \rangle = \langle I - \sum_{i=1}^{n-1} w_i \psi_{\mathbf{n}_i}, \psi_{\mathbf{n}_n} \rangle \neq 0 \ . \tag{2.30}$$

I is then again projected orthogonally into the new vector space $< \psi_{\mathbf{n}_1}, \ldots, \psi_{\mathbf{n}_n} >$ in order to calculate the weights $w_1, \ldots, w_n$. This may be repeated until N wavelets and weights have been found. How this projection is done will be described in the next section.

In the remainder of this section, we want to show that the resulting family of Gabor wavelets Ψ constitutes a basis, i.e. that all $\psi \in \Psi$ are linearly independent. This will be shown by induction. Clearly this holds for one wavelet. If we already have a family of wavelets $(\psi_{\mathbf{n}_1}, \ldots, \psi_{\mathbf{n}_{n-1}})$ that constitutes a basis, we have to show that the newly selected wavelet $\psi_{\mathbf{n}_n}$ is linearly independent of the others: $\psi_{\mathbf{n}_n} \notin < \psi_{\mathbf{n}_1}, \ldots, \psi_{\mathbf{n}_{n-1}} >$. Assuming that

$$\psi_{\mathbf{n}_n} \in < \psi_{\mathbf{n}_1}, \ldots, \psi_{\mathbf{n}_{n-1}} >$$

we have

$$< \psi_{\mathbf{n}_1}, \ldots, \psi_{\mathbf{n}_{n-1}} > = < \psi_{\mathbf{n}_1}, \ldots, \psi_{\mathbf{n}_n} >$$

and in particular

$$(< \psi_{\mathbf{n}_1}, \ldots, \psi_{\mathbf{n}_{n-1}} >)^{\perp} = (< \psi_{\mathbf{n}_1}, \ldots, \psi_{\mathbf{n}_n} >)^{\perp} \ .$$

This again means that

$$I - \sum_{i=1}^{n-1} w_i \psi_{\mathbf{n}_i} \in (< \psi_{\mathbf{n}_1}, \ldots, \psi_{\mathbf{n}_n} >)^{\perp} \ ,$$

which implies

$$\langle I - \sum_{i=1}^{n-1} w_i \psi_{\mathbf{n}_i}, \psi_{\mathbf{n}_n} \rangle = 0 \ .$$

This, however, contradicts the choice of $\psi_{\mathbf{n}_n}$ in the optimization step, where $\psi_{\mathbf{n}_n}$ was selected such that

$$\langle I - \sum_{i=1}^{n-1} w_i \psi_{\mathbf{n}_i}, \psi_{\mathbf{n}_n} \rangle \neq 0 \ .$$

Therefore the $\psi_{\mathbf{n}_i}$ are all linearly independent and the Gabor wavelet family $\Psi = \{\psi_{\mathbf{n}_1}, \ldots, \psi_{\mathbf{n}_n}\}$ constitutes a basis.

The above discussion has been in terms of discrete images I, but it evidently holds also for continuous functions $f \in \mathbb{L}^2(\mathbb{R}^2)$.

2.4 Direct Calculation of Weights

Gabor wavelet functions are non-orthogonal. For a given family Ψ of Gabor wavelets it is not possible to calculate a weight w_i directly by simple projection of the Gabor wavelet $\psi_{\mathbf{n}_i}$ onto the image. In this section we explain how simple computation of the weights is still possible.

In [Daubechies, 1990; Daugman, 1988] it was proposed to use eq. (2.23) to find the optimal weight w_i for each fixed wavelet through optimization. Because optimization is a slow process, however, we want to introduce two approaches to directly calculating the weights. The first approach is derived from wavelet theory and employs bi-orthogonal and dual wavelets. The second approach is derived from linear algebra. As we will see, both approaches are equivalent, but the approach(or better, the interpretation of the problem) based on the dual wavelets leads to a better and more stable solution.

As already mentioned, Gabor wavelets are non-orthogonal wavelets[†]. This problem can be solved by considering the bi-orthogonal family of wavelets $\tilde{\mathbf{\Phi}}$ [Chui, 1992; Feichtinger and Strohmer, 1998; Mallat, 1998] (see also eq. (2.7)):

Definition 3
Two families of wavelets, $\mathbf{\Phi} = \{\phi_i\}$ *and* $\tilde{\mathbf{\Phi}} = \{\tilde{\phi}_i\}$ *are called* bi-orthogonal *iff for all* i, j *they satisfy the bi-orthogonality condition:*

$$\langle \phi_i, \tilde{\phi}_j \rangle \;=\; \delta_{i,j} \; . \tag{2.31}$$

The wavelet $\tilde{\phi}$ *is called the* dual *wavelet of* ϕ*.*

Of course, when $\{\phi_i\}$ constitutes an orthogonal family, we have $\phi_i = \tilde{\phi}_i$ for all i.

If not stated otherwise, we will use the symbol $\mathbf{\Phi}$ when we want to refer to general wavelets, and will use the symbol Ψ when we want to refer explicitly to Gabor wavelets.

The use of bi-orthogonal wavelets allows direct calculation of weights: Let $f \in \mathbb{L}^2(\mathbb{R}^2)$, and let $\mathbf{\Phi} = \{\phi_i\}$ be a family of wavelets that constitutes a frame. Let $\tilde{\mathbf{\Phi}} = \{\tilde{\phi}_i\}$ be the family

[†] *Non-orthogonality of wavelets* is understood in the sense that the wavelet coefficients and the weights of the superposition are different in a non-orthogonal frame. Gabor wavelets can be considered to be approximately orthogonal only when their overlap is small. However, in this case no reconstruction is possible, so this case is of no interest to us.

of dual wavelets. Then there exist weights $\{w_i\}$ such that

$$f = \sum_i w_i \phi_i \ . \tag{2.32}$$

A weight w_k can then be calculated by using the dual wavelet $\tilde{\phi}_k$

$$\begin{aligned} \langle f, \tilde{\phi}_k \rangle &= \int f(x)\tilde{\phi}_k(x)dx \\ &= \int \left[\sum_i w_i \phi_i(x)\right] \tilde{\phi}_k(x)dx \\ &= \sum_i w_i \left[\int \phi_i(x)\tilde{\phi}_k(x)dx\right] \\ &= \sum_i w_i \delta_{i,k} \\ &= w_k \ . \end{aligned} \tag{2.33}$$

When the family of wavelets $\{\phi_i\}$ does not constitute a frame, (2.32) holds only approximately. A dual wavelet family $\{\tilde{\phi}_i\}$ constitutes an orthogonal projection of the function f onto the subspace $< \{\phi_i\} >$ which results in an optimal approximation of f by the $\{\phi_i\}$ in the mean square sense. Fig. 2.4 shows a geometrical interpretation of an orthogonal projection of a function f onto the wavelet family $\mathbf{\Phi} = \{\phi_0, \phi_1\}$. Applying the above discussion to our problem

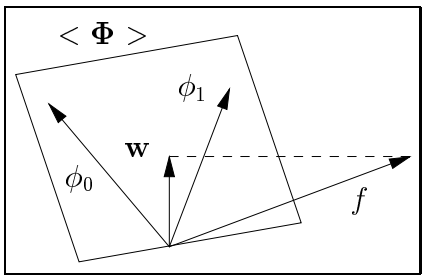


Figure 2.6. Geometrical interpretation of the least squares solution, illustrated for the case of a function f and two wavelets ϕ_0 and ϕ_1. The corresponding wavelet network output is represented as a linear combination of the two wavelets ϕ_0 and ϕ_1. The least-squares solution for $\mathbf{w}$ is given by the orthogonal projection of f onto $< \mathbf{\Phi} >$.

of finding the right weights w_i for a family of Gabor wavelets $\Psi = \{\psi_{\mathbf{n}_1}, \ldots, \psi_{\mathbf{n}_N}\}$ of some GWN, the w_i can be found by projecting the *dual* wavelets $\tilde{\psi}_{\mathbf{n}_i}$. The Gabor wavelet family $\{\tilde{\psi}_{\mathbf{n}_i}\}$ is the *dual* family to the Gabor wavelet family $\{\psi_{\mathbf{n}_i}\}$ iff it fulfills for each i, j the bi-orthogonality condition

$$\langle \psi_{\mathbf{n}_i}, \tilde{\psi}_{\mathbf{n}_j} \rangle = \delta_{i,j} \ . \tag{2.34}$$

With $\tilde{\Psi} = (\tilde{\psi}_{\mathbf{n}_1}, \ldots, \tilde{\psi}_{\mathbf{n}_N})^T$, we can write

$$\left[\langle \tilde{\Psi}, \Psi \rangle\right] = \mathbb{1} \ . \tag{2.35}$$

In other words, the family of dual wavelets $\tilde{\Psi}$ can be used to find the optimal set of weights $\mathbf{w}$:

$$\begin{aligned} w_i &= \langle \tilde{\psi}_{\mathbf{n}_i}, g \rangle \\ \mathbf{w} &= \tilde{\Psi} g \end{aligned} \tag{2.36}$$

Ψ and $\tilde{\Psi}$ are vectors of Gabor functions and their dual functions, respectively. The notation in eq. (2.36) refers to the continuous scalar products of each of the functions $\tilde{\psi}_{\mathbf{n}_i}$ with g.

In the following, the same symbols Ψ and $\tilde{\Psi}$ will refer to matrices. The functions $\tilde{\psi}_{\mathbf{n}_i}$ are assumed to be discretized and the i-th rows in the matrices Ψ and $\tilde{\Psi}$ contain the discrete values of $\psi_{\mathbf{n}_i}$ and $\tilde{\psi}_{\mathbf{n}_i}$. The product $\tilde{\Psi} g$ of the matrix $\tilde{\Psi}$ and the vector g is then just the discrete version of the scalar products in eq. (2.36). It will be clear from the context whether the continuous or the discrete case is being considered. In the notation of eq. (2.36), the discrete version of eq. (2.35) says that the matrix product $\tilde{\Psi}\Psi = \mathbb{1}$.

We find that

$$\tilde{\psi}_{\mathbf{n}_i} = \sum_{j=1}^{N} \left(\Psi_{i,j}\right)^{-1} \psi_{\mathbf{n}_j} \ , \tag{2.37}$$

where $\Psi_{i,j} = \langle \psi_{\mathbf{n}_i}, \psi_{\mathbf{n}_j} \rangle$ is the matrix of the pairwise scalar products. In order to show that the $\tilde{\psi}_{\mathbf{n}_i}$ in eq. (2.37) are indeed dual to be $\psi_{\mathbf{n}_i}$, we have to verify the bi-orthogonality condition (2.34):

$$\begin{aligned} \langle \psi_{\mathbf{n}_i}, \sum_{j=1}^{N} \left(\Psi_{k,j}\right)^{-1} \psi_{\mathbf{n}_j} \rangle &= \int \psi_{\mathbf{n}_i}(x) \left[\sum_{j=1}^{N} \left(\Psi_{k,j}\right)^{-1} \psi_{\mathbf{n}_j}(x) \right] dx \\ &= \sum_{j=1}^{N} \left(\Psi_{k,j}\right)^{-1} \left[\int \psi_{\mathbf{n}_i}(x) \psi_{\mathbf{n}_j}(x) dx \right] \\ &= \sum_{j=1}^{N} \left(\Psi_{k,j}\right)^{-1} \langle \psi_{\mathbf{n}_i}, \psi_{\mathbf{n}_j} \rangle \\ &= \sum_{j=1}^{N} \left(\Psi_{k,j}\right)^{-1} \left(\Psi_{j,i}\right) \\ &= \delta_{i,k} \ . \end{aligned} \tag{2.38}$$

In the second to last row, the i-th column of matrix $(\Psi_{i,j})$ is multiplied by the k-th row of its inverse, which evaluates to 1 if $i = k$, and to 0 otherwise. Equation (2.37) is not specific to Gabor wavelets, as one can see in the proof, but holds for *any* function family of finite dimensionality.

Equation (2.36) allows us to define the operator

$$\mathcal{T}_\Psi : \mathbb{L}^2(\mathbb{R}^2) \longmapsto \mathbb{L}^2(\mathbb{R}^2) \tag{2.39}$$

as follows: Given a family Ψ of wavelets of a GWN, the operator $\mathcal{T}_\Psi$ realizes an orthogonal projection of $\mathbb{L}^2(\mathbb{R}^2)$ onto $< \{\psi_{\mathbf{n}_i}\} > \subset \mathbb{L}^2(\mathbb{R}^2)$ (the closed linear span of $\{\psi_{\mathbf{n}_i}\}$):

$$\hat{g} = \mathcal{T}_\Psi(g) = \Psi\tilde{\Psi}g = \sum_{i=1}^{N} w_i \psi_{\mathbf{n}_i} = \Psi\mathbf{w}^t$$
$$\text{with } \mathbf{w}^t = \tilde{\Psi}g \ . \tag{2.40}$$

Equation (2.40) can be interpreted as follows: Given a function g, we search for the vector $\mathbf{w} \in \mathbb{R}^N$ such that $\hat{g} = \Psi\mathbf{w}^t$, which is optimally solved in a mean square sense, as explained above, by the dual $\tilde{\Psi}$: $\mathbf{w}^t = \tilde{\Psi}g$. In this sense, the function $\tilde{\Psi}$ maps a function g into the vector space $\mathbb{R}^n$. The re-mapping of $\mathbf{w}$ from the vector space $\mathbb{R}^N$ onto $< \{\psi_{\mathbf{n}_i}\} > \subset \mathbb{L}^2(\mathbb{R}^2)$ is established by eq. (2.24), i.e. $\hat{g} = \Psi\mathbf{w}^t$. These relations are sketched in Fig. 2.7.

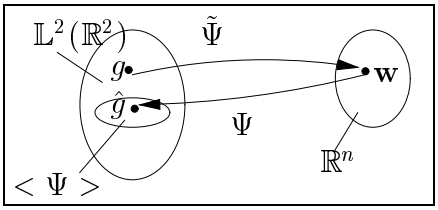


Figure 2.7. A function $g \in \mathbb{L}^2(\mathbb{R}^2)$ is mapped by the linear mapping $\tilde{\Psi}$ into the vector $\mathbf{w}$ of the vector space $\mathbb{R}^N$. The mapping of $\mathbf{w}$ into $< \{\psi_{\mathbf{n}_i}\} > \subset \mathbb{L}^2(\mathbb{R}^2)$ is achieved by the linear mapping Ψ. $\tilde{\Psi}$ can be identified with the pseudo-inverse of Ψ and the mapping of $\mathbb{L}^2(\mathbb{R}^2)$ onto $\mathbb{R}^N$, $\tilde{\Psi}g = \mathbf{w}$, is an orthogonal projection.

The above interpretation of eq. (2.40) suggests that one could also find the weight vector $\mathbf{w}$ by considering the pseudo-inverse of Ψ, as proposed in e.g.[Bishop, 1995]:

$$\mathbf{w}^t = \Psi^+ g \ . \tag{2.41}$$

The pseudo-inverse Ψ^+ is defined as

$$\Psi^+ = (\Psi^t\Psi)^{-1}\Psi^t \ . \tag{2.42}$$

A close look at this definition reveals a close relation to eq. (2.37): In fact, eq. (2.42) is nothing else than eq. (2.37) written in matrix notation, i.e. the discrete version of the continuous eq. (2.37).

It is interesting to mention that the two mappings $\mathbf{\Phi}$ and $\tilde{\mathbf{\Phi}}$ do not commute. This can also be seen in Fig. 2.7: $\mathbf{\Phi}\tilde{\mathbf{\Phi}}$ constitutes a mapping from $\mathbb{L}^2(\mathbb{R}^2)$ onto $< \{\psi_{\mathbf{n}_i}\} > \subset \mathbb{L}^2(\mathbb{R}^2)$, while $\tilde{\mathbf{\Phi}}\mathbf{\Phi} = \mathbb{I}$ is a mapping from $\mathbb{R}^N$ onto $\mathbb{R}^N$.

It was mentioned above that the interpretation through dual wavelets leads to a more stable solution. This can be seen by substituting eq. (2.37) into eq. (2.36). Each w_i is calculated by first determining the inner product between the function g and each of the wavelets $\psi_{\mathbf{n}_i}$: $\langle \psi_{\mathbf{n}_i}, g \rangle$. Then, in order to compute the weights, the vector of the inner products is multiplied by the matrix $(\Psi_{i,j})^{-1}$ (see eq. (2.37)). It is easy to show that the matrix $(\Psi_{i,j})^{-1}$ is, except for a factor, invariant with respect to the affine deformations of the GWN. It can therefore be computed off-line. Furthermore, to compute the weights, it is sufficient to calculate the inner product of each of the wavelets with the image, and the number of multiplications and additions needed for the product with the matrix is given by the number N of wavelets. On the other hand, using the interpretation based on the pseudo-inverse, the matrix Ψ^+ is multiplied by g. For discrete g, the matrix Ψ^+ has to have the same dimensionality as g. Also, Ψ^+ is not invariant with respect to the affine deformations of the GWN.

Clearly, both methods are equivalent; the difference is solely in the interpretation, which implies a different order of the computational steps that have to be carried out.

2.5 Distance Measures for Gabor Wavelet Networks

It is of interest to determine how similar two Gabor wavelet representations are. In this section we introduce and discuss various distance measurements:

1. A distance measurement between two specific GWNs $(\Psi_1, \mathbf{w}_1)$ and $(\Psi_2, \mathbf{w}_2)$. This allows us to compare two (possibly different) objects that are represented using different GWNs.

2. A distance measurement between two weight vectors $\mathbf{w}_1$ and $\mathbf{w}_2$ of a specific wavelet family Ψ, i.e., comparison of the two GWNs $(\Psi, \mathbf{w}_1)$ and $(\Psi, \mathbf{w}_2)$. This measurement allows us to compare two objects that are represented using the same wavelet family Ψ.

3. A distance measurement between two wavelet families Ψ_1 and Ψ_2. This measurement allows direct comparison of the two GWNs without considering the weight vectors.

These three distance measurements will be introduced in the following sections.

2.5.1 Direct Calculation of Distances between two Gabor Wavelet Networks

It was mentioned in Section 2.3 that optimization is a crucial problem. Finding a global optimum for the free wavelet parameters is very time-consuming, so that determining a local

minimum seems to be the only feasible solution. However, we have found in various experiments that the local minimum that is found using the Levenberg Marquardt method of Section 2.3 is extremely unrobust with respect to the initial values. A slight variation in the initial values may result in a completely different GWN. It is therefore reasonable to ask whether it is possible to compare two different GWNs in order to find out if they represent the same function f. This question can be reformulated using a basis transformation:

Given a function $f \in \mathbb{L}^2(\mathbb{R}^2)$, let f be represented by two different wavelet networks $(\mathbf{\Phi}^1, \mathbf{v})$ and $(\mathbf{\Phi}^2, \mathbf{w})$ with the wavelet families $\{\phi_i^1 | i = 1 \dots N\}$ and $\{\phi_i^2 | i = 1 \dots M\}$:

$$\begin{aligned} \hat{f}_1 &= \sum_{j=1}^{N} v_i \phi_i^1 \\ \hat{f}_2 &= \sum_{j=1}^{M} w_i \phi_i^2 \ . \end{aligned} \tag{2.43}$$

To compare the two wavelet networks, we have to transform the vector $\mathbf{v} \in \mathbb{R}^N$ of wavelet network $\mathbf{\Phi}^1$ into a vector $\mathbf{v}' \in \mathbb{R}^M$ that is given with respect to the wavelet network $\mathbf{\Phi}^2$. To do this, we use the technique of the dual wavelets: In order to represent $\hat{f}_1$ with the wavelets of $\mathbf{\Phi}^2$, we apply the dual wavelets $\tilde{\mathbf{\Phi}}^2$ of $\mathbf{\Phi}^2$ to $\hat{f}^1$ (see Fig. 2.7):

$$\mathbf{v}' = \tilde{\mathbf{\Phi}}^2 \hat{f}_1 = \tilde{\mathbf{\Phi}}^2 \mathbf{\Phi}^1 \mathbf{v} \ . \tag{2.44}$$

With this projection, $\mathbf{v}'$ now represents $\mathbf{v}$ with respect to the other wavelet network $\mathbf{\Phi}^2$. This procedure is sketched in Fig. 2.8. The same can be done for $\mathbf{w} \in \mathbb{R}^M$:

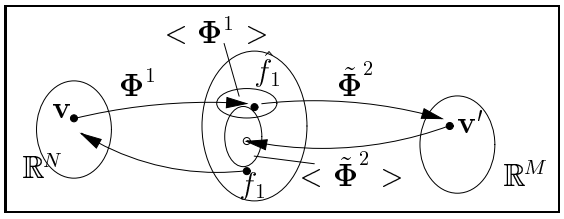


Figure 2.8. This figure sketches the basis transformation from one wavelet network onto another. A function $f_1 \in \mathbb{L}^2(\mathbb{R}^2)$ is projected into $\mathbb{R}^N$ and re-mapped into $\hat{f}_1$ in the subspace $< \mathbf{\Phi}^1 > \subset \mathbb{L}^2(\mathbb{R}^2)$. $\hat{f}_1$ is then mapped into $\mathbb{R}^M$.

$$\mathbf{w}' = \tilde{\mathbf{\Phi}}^1 \hat{f}_2 = \tilde{\mathbf{\Phi}}^1 \mathbf{\Phi}^2 \mathbf{w} \ . \tag{2.45}$$

The intermediate mapping from $\mathbf{v}$ to $\hat{f}_1$ in Fig. 2.8 is for visualization purposes, and can be omitted by understanding eq. (2.44) as

$$\mathbf{v}' = \tilde{\mathbf{\Phi}}^2 \hat{f}_1 = \left(\tilde{\mathbf{\Phi}}^2 \mathbf{\Phi}^1 \right) \mathbf{v} \ .$$

Using these two equations, we can compare $\mathbf{v}$ with $\mathbf{w}'$ and $\mathbf{w}$ with $\mathbf{v}'$:

$$\begin{aligned} d_{\mathbf{\Phi}^1}(\mathbf{v}, \mathbf{v}') &= \|\mathbf{v} - \mathbf{w}'\|_{\mathbf{\Phi}^1} \\ d_{\mathbf{\Phi}^2}(\mathbf{w}, \mathbf{w}') &= \|\mathbf{w} - \mathbf{v}'\|_{\mathbf{\Phi}^2} \ . \end{aligned}$$

These difference measures will be discussed in the next subsection. Now that we have two GWNs that are possibly optimized on different functions, these difference measures allow us to calculate the distance between the representations, and from this, the difference between the represented functions.

Clearly, the distance that can be calculated is *not* given by the difference between the original functions, but rather by the difference between the orthogonal projections of the functions onto the respective wavelet spaces. The above discussion therefore provides information only about the similarity of the two GWNs, but not about the similarity of the two possibly different functions.

The above two distance measures are specific for each wavelet family $\mathbf{\Phi}^i$. Calculating the difference between two GWNs $(\mathbf{\Phi}^1, \mathbf{v})$ and $(\mathbf{\Phi}^2, \mathbf{w})$ is therefore reduced to calculating the distances between corresponding vectors ($\mathbf{v}$ and $\mathbf{w}'$ or $\mathbf{w}$ and $\mathbf{v}'$) with respect to each wavelet space using the distance measures $d_{\mathbf{\Phi}^1}$ and $d_{\mathbf{\Phi}^2}$.

In the next subsection we will investigate the difference measure $d_{\mathbf{\Phi}}(\cdot\ ,\ \cdot)$.

2.5.2 Measuring Distances in Gabor Wavelet Space

In the previous section we used the notation $d_{\mathbf{\Phi}}(\mathbf{v}, \mathbf{w})$ in calculating the distance between two wavelet vectors $\mathbf{v}$ and $\mathbf{w}$ with respect to the wavelet basis $\mathbf{\Phi}$. However, it is not yet clear how this distance measure should be calculated. As done in [Wiskott *et al.*, 1997], one could calculate the Euclidean distance between these two weight vectors, which corresponds to the angle between the two vectors. But such a distance lacks any justification or geometrical interpretation, because it is not clear what the angle between the two vectors tells us about the difference between the images they represent.

In this section we therefore propose a different distance measurement and a different norm. They are derived from the Euclidean norm and Euclidean distance in the image space. The difference $d_{\mathbf{\Phi}}$ is defined to be the Euclidean distance between the two reconstructed images:

$$\begin{aligned} d_{\mathbf{\Phi}}(\mathbf{v}, \mathbf{w}) &= \|\mathbf{v} - \mathbf{w}\|_{\mathbf{\Phi}} \\ &:= \|\sum_{i=1}^{N} v_i \phi_i - \sum_{j=1}^{N} w_j \phi_j\|_2 \ . \end{aligned} \tag{2.46}$$

Various transformations lead to

$$\begin{aligned}
d_{\mathbf{\Phi}}(\mathbf{v},\mathbf{w}) &= \|\mathbf{v}-\mathbf{w}\|_{\mathbf{\Phi}} \\
&= \|\sum_{i=1}^{N} v_i\phi_i - \sum_{j=1}^{N} w_j\phi_j\|_2 \\
&= \left[\int \left(\sum_{i=1}^{N} v_i\phi_i(x) - \sum_{j=1}^{N} w_j\phi_j(x)\right)^2 dx\right]^{\frac{1}{2}} \\
&= \left[\int \left(\sum_{i=1}^{N} \delta_i\phi_i(x)\right)^2 dx\right]^{\frac{1}{2}} \quad \text{with } \delta_i = (v_i - w_i) \\
&= \left[\int \left(\sum_{i=1}^{N}\sum_{j=1}^{N} \delta_i\delta_j\phi_i(x)\phi_j(x)\right) dx\right]^{\frac{1}{2}} \\
&= \left[\sum_{i=1}^{N}\sum_{j=1}^{N} \delta_i\delta_j \int \phi_i(x)\phi_j(x)dx\right]^{\frac{1}{2}} \\
&= \left[\sum_{i,j} \delta_i\delta_j\langle\phi_i,\phi_j\rangle\right]^{\frac{1}{2}}. \qquad (2.47)
\end{aligned}$$

We therefore define the norm $\|\cdot\|_{\mathbf{\Phi}}$ as

$$\begin{aligned}
\|\mathbf{w}\|_{\mathbf{\Phi}} &:= \left[\sum_{i,j} w_i w_j\langle\phi_i,\phi_j\rangle\right]^{\frac{1}{2}} \\
&= \left(\mathbf{w}^t \left(\mathbf{\Phi}_{i,j}\right) \mathbf{w}\right)^{\frac{1}{2}} \qquad (2.48)
\end{aligned}$$

and the difference $d_{\mathbf{\Phi}}(\cdot,\cdot)$ as

$$\begin{aligned}
d_{\mathbf{\Phi}}(\mathbf{v},\mathbf{w}) &:= \left[\sum_{i,j} \delta_i\delta_j\langle\phi_i,\phi_j\rangle\right]^{\frac{1}{2}} \quad \text{where } \delta_i = (v_i - w_i) \\
&= \left(\boldsymbol{\delta}^t \left(\mathbf{\Phi}_{i,j}\right) \boldsymbol{\delta}\right)^{\frac{1}{2}} \text{with } \boldsymbol{\delta} = \left(\delta_1 \cdots \delta_N\right)^t. \qquad (2.49)
\end{aligned}$$

The products $\langle\phi_i,\phi_j\rangle$ have already appeared in eq. (2.37), where $\left(\Phi_{i,j}\right) = \langle\phi_i,\phi_j\rangle$ constitutes the matrix of the pairwise scalar products. This matrix is a measure of the "overlap" of the wavelets.

If the wavelets $\{\phi_i\}$ are orthogonal, the products are

$$\langle \phi_i, \phi_j \rangle \quad = \quad \begin{cases} 1, & \text{if } i = j \\ 0, & \text{if } i \neq j \ . \end{cases} \tag{2.50}$$

This means that $\| \cdot \|_{\mathbf{\Phi}} = \| \cdot \|_2$ holds for the Euclidean norm.

As mentioned above, the matrix $\mathbf{\Phi}$ is, up to a scalar factor, invariant with respect to affine transformations of the wavelet network. It can therefore be computed off-line beforehand.

2.5.3 Direct Comparison between two Gabor Wavelet Families

In the previous section we discussed how two wavelet networks can be compared. However, a generalization of the above results would allow the comparison to be independent of the weight vector. We will now discuss this.

A direct calculation of the distance between two families of wavelets, Ψ and $\mathbf{\Phi}$, is established by applying the above method to each of the wavelets $\phi_i \in \mathbf{\Phi}$:

$$\mathcal{T}_{\Psi}(\phi_j) \quad = \quad \sum_i \left[\langle \phi_j, \tilde{\psi}_i \rangle \right] \psi_i \ . \tag{2.51}$$

In Eq. (2.51) each wavelet ϕ_j is projected orthogonally onto the subspace $< \{\psi_j\} > \subset \mathbb{L}^2(\mathbb{R}^2)$. Each ϕ_i is represented as a linear combination of the wavelets ψ_i, so that

$$\sum_j \frac{\|\phi_j - \mathcal{T}_{\Psi}(\phi_j)\|_2}{\|\phi_j\|_2} \tag{2.52}$$

can be considered as a measure of how well the vector space $< \phi_j >$ can be approximated by the vector space $< \psi_j >$.

Similarly, the reverse combination

$$\sum_j \frac{\|\psi_j - \mathcal{T}_{\mathbf{\Phi}}(\psi_j)\|_2}{\|\psi_j\|_2} \tag{2.53}$$

measures how well the vector space $< \psi_j >$ can be approximated by the vector space $< \phi_j >$.

By combining eqs. (2.52) and (2.53), the distance between Ψ and $\mathbf{\Phi}$ can be determined by

$$d_{\Psi,\mathbf{\Phi}} = \sqrt{\left[\sum_j \frac{\|\phi_j - T_{\Psi}(\phi_j)\|_2}{\|\phi_j\|_2} \right]^2 + \left[\sum_j \frac{\|\psi_j - T_{\mathbf{\Phi}}(\psi_j)\|_2}{\|\psi_j\|_2} \right]^2} \ , \tag{2.54}$$

where $\| \cdot \|$ is the Euclidean norm. Using this distance measure, the distance between two object representations can be calculated very efficiently. Clearly, eq. (2.54) can also be computed by applying the distance measures d_{Ψ} and $d_{\mathbf{\Phi}}$.

2.6 Reparameterizing Gabor Wavelet Networks

We illustrated above that a GWN that is optimized on a particular object is very specific to that object. In order to ensure meaningful calculation of the weights and meaningful filtration of the image with the Gabor filter, the wavelets have to positioned precisely on the features they are supposed to represent. Compare, e.g., the two images in Fig. 2.9. We see that in the left

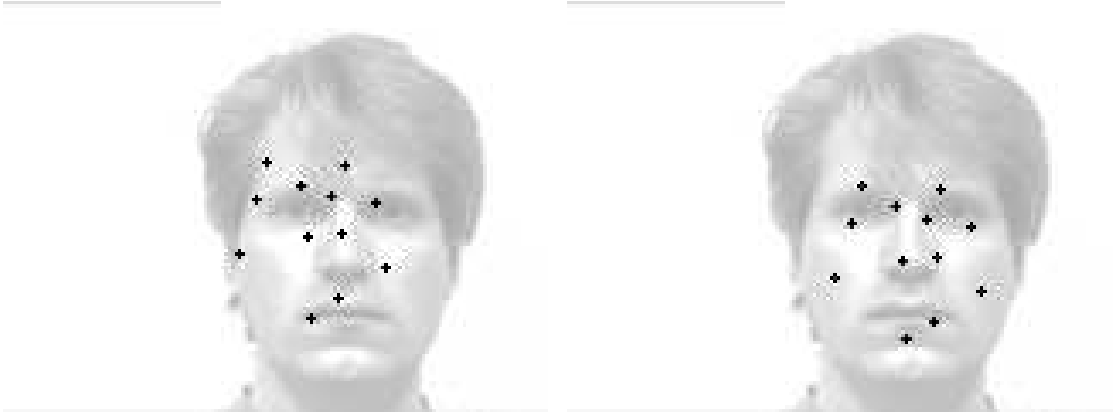


Figure 2.9. The left image shows a GWN that is positioned incorrectly on the facial image: features are not positioned on the features they should represent. The right image shows the correct positions.

image the wavelet positions are not correct. Computing the filter responses and the weights on the basis of these positions cannot lead to satisfactory results. On the other hand, calculating the filter responses and the weights on the basis of the correct positions[‡] ensures a correct relation between the filter parameters and the object that the filters are applied to. Consequently, the filter responses will be meaningful in the sense that their weights will be appropriately related to the object features. The task of finding the position, the scale and the orientation of a GWN in a new image is therefore very important, and will be dealt with in this section.

As another example, consider an image J that shows the person of Fig. 2.4 left, possibly distorted affinely. Given the corresponding GWN, we are interested (for example, in a tracking application) in finding the correct position, orientation and scale of the GWN so that the wavelets are positioned on the same facial features as in the original image. The parameters of the reparameterized GWN allow conclusions about the 3-D parameters of the tracked head.

[‡]The term "correct positions" refers to the positions originally taken by the wavelets after the optimization procedure.

Another example can be seen in Fig. 2.5, where the original positions of the wavelets are marked in the bottom right image, and in Fig. 2.11, where the wavelet positions of the *reparameterized* wavelet network are marked in new images.

Parameterization of a wavelet net is established by using a *superwavelet* [Szu *et al.*, 1992].

Definition 4
Let $(\Psi, \mathbf{w})$ *be a GWN with* $\Psi = (\psi_{\mathbf{n}_1}, \ldots, \psi_{\mathbf{n}_N})^T$, $\mathbf{w} = (w_1, \ldots, w_N)^T$. *A* superwavelet $\Psi_{\mathbf{n}}$ *is defined to be a linear combination of the wavelets* $\psi_{\mathbf{n}_i}$ *such that*

$$\Psi_{\mathbf{n}}(\mathbf{x}) \;=\; \sum_i w_i \psi_{\mathbf{n}_i}(\mathbf{SR}(\mathbf{x}-\mathbf{c}))\,, \tag{2.55}$$

where the parameters of the vector $\mathbf{n} = (c_x, c_y, \theta, s_x, s_y)$ *of the superwavelet* Ψ *are the dilation matrix* $\mathbf{S} = diag(s_x, s_y)$, *the rotation matrix* $\mathbf{R}$, *and the translation vector* $\mathbf{c} = (c_x, c_y)^T$.

A superwavelet $\Psi_{\mathbf{n}}$ is a wavelet [Szu *et al.*, 1992], and in particular a continuous function that has the wavelet parameters dilation, translation and rotation (see Section 2). Therefore, we can handle it in the same way as we handled each individual wavelet in the previous section. For a new image g we can arbitrarily deform the superwavelet by optimizing its parameters $\mathbf{n}$ with respect to the energy functional E:

$$E \;=\; \min_{\mathbf{n}} \|g - \Psi_{\mathbf{n}}\|_2^2 \tag{2.56}$$

Equation (2.56) defines the operator

$$\begin{aligned} \mathcal{P}_\Psi : \mathbb{L}^2(\mathbb{R}^2) \;&\longmapsto\; \mathbb{R}^5 \\ g \;&\longrightarrow\; \mathbf{n} = (c_x, c_y, \theta, s_x, s_y)\,, \end{aligned} \tag{2.57}$$

where $\mathbf{n}$ minimizes the energy functional E of eq. (2.56). In eqs. (2.56) and (2.57) Ψ is derived from the GWN of image f. For optimization of the superwavelet parameters, we can use the same optimization procedure that we used to find the GWNs. An example of the optimization process can be seen in Fig. 2.10: Sketched as white rectangular boxes are the initial values of $\mathbf{n}$, the values of $\mathbf{n}$ after 2 and 4 optimization cycles, and the final values of $\mathbf{n}$ after 8 cycles. The box indicates the image region in which the wavelets were initially homogeneously distributed, as shown in Fig. 2.5. Its center position marks the center position of the corresponding superwavelet. The superwavelet used in Fig. 2.10 is $\hat{I}_{16}$ of Fig. 2.5, i.e. it is derived from the person in Fig. 2.4. Another example can be seen in Fig. 2.11. The top images should be compared with the bottom right image in Fig. 2.5: It can be seen that the wavelets are positioned correctly on the correct facial features. The images at the bottom of fig. 2.11 show the reconstructions using the reparameterized GWNs.

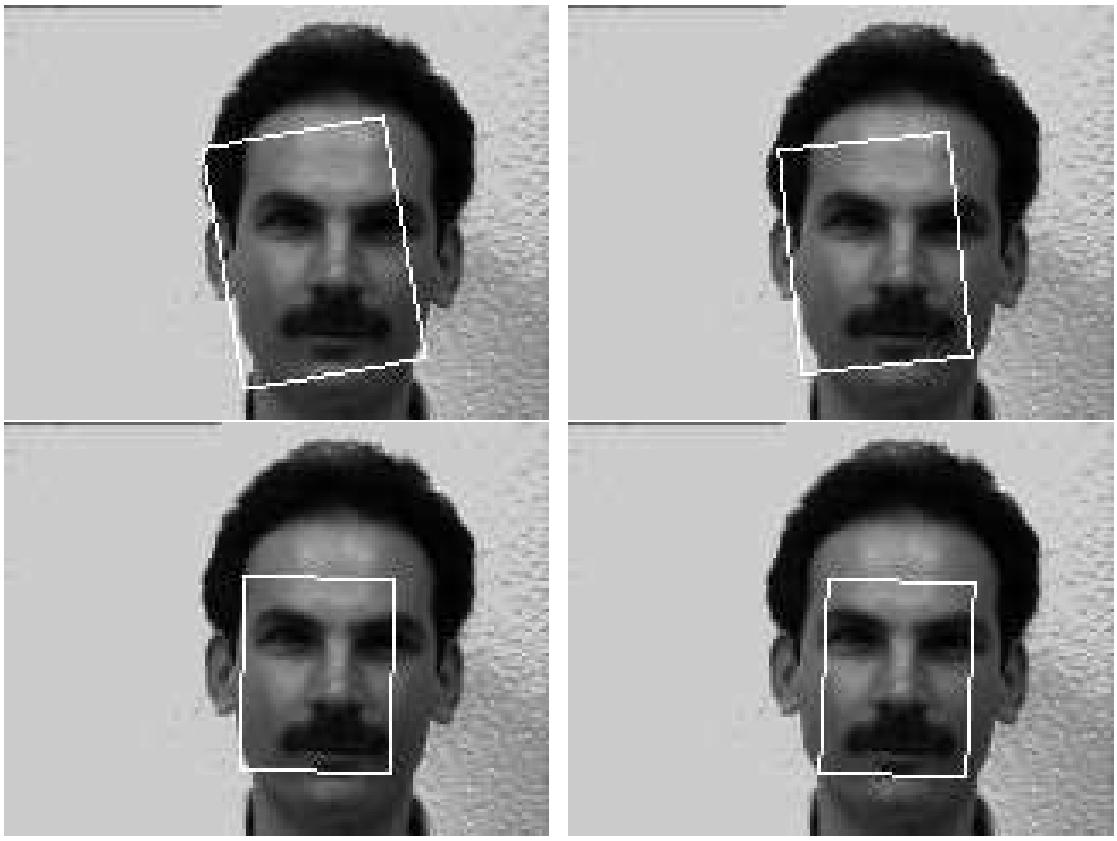


Figure 2.10. These images show the 1st, 2nd(top), 4th, and 8th (final) step (bottom) of the Levenberg-Marquardt method of optimizing the parameters of a superwavelet. In the top left image the initial values are shifted by 10 px. off the true position, rotated by 10° and scaled by 20%. The bottom right image shows the final result. $\hat{I}_{16}$ of Fig. 2.4 was used as the superwavelet.

The image distortion of a planar object that is viewed under orthographic projection is described by six parameters: translation c_x, c_y, rotation θ, dilation s_x, s_y, and shear s_{xy}. The degrees of freedom of a wavelet only allow translation, dilation and rotation. However, it is straightforward to include shear, and thus to allow any affine deformation of $\Psi_{\mathbf{n}}$. For this, we enhance the parameter vector $\mathbf{n}$ to a six-dimensional vector

$$\mathbf{n} = (c_x, c_y, \theta, s_x, s_y, s_{xy})^T$$

By rewriting the scaling matrix $\mathbf{S}$ as

$$\mathbf{S} = \begin{pmatrix} s_x & s_{xy} \\ 0 & s_y \end{pmatrix} \quad ,$$

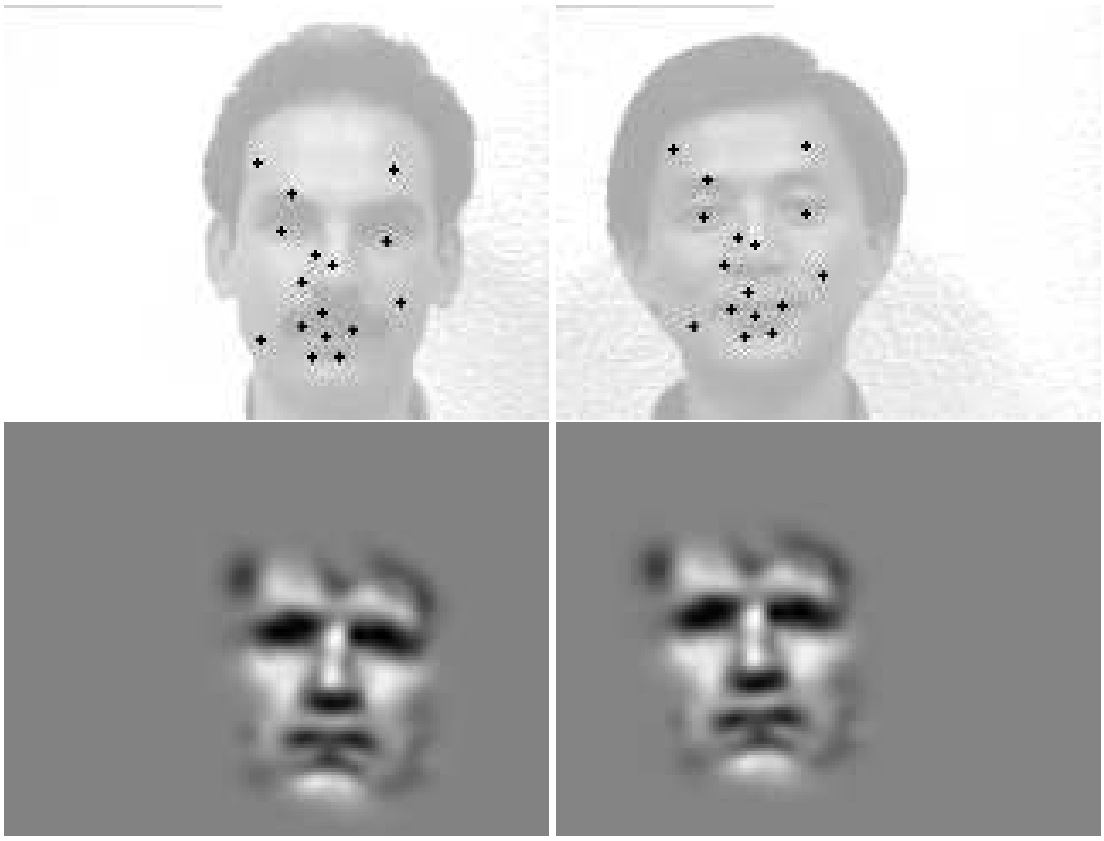


Figure 2.11. These images show the positions of each of the 16 wavelets after reparameterizing the wavelet net (top), and the corresponding reconstruction (bottom). The reconstructed faces have the same orientation, position and size that they were reparameterized on.

we are now able to deform the superwavelet $\Psi_{\mathbf{n}}$ affinely.

The reparametrization of the superwavelet can be understood as warping, where the original face, represented by the GWN $(\Psi, \mathbf{w})$, is warped into the new face. This idea is shown in Fig. 2.12.

The reparametrization (warping) works quite robustly. Using the superwavelet $\hat{I}_{16}$ or $\hat{I}_{52}$, we have found in several experiments that the initialization of $\mathbf{n}_0$ may vary from the correct parameters by approximately. ± 10 pixels in the x- and y- direction, by approximately 20% in scale, and by approximately $\pm 10^\circ$ in rotation (see Fig. 2.10). Of course, these are only approximate values since they depend on the number of wavelets used, on the template face, and on the scale of the wavelets. In our case, 10 pixels. corresponds to $\approx 1/3$ of the width of the white box in Fig. 2.10 that marks the inner face region.

Figure 2.12. These two images show the wavelet network $\hat{I}_{52}$, repositioned onto the two test images of Fig. 2.11. This demonstrates that the repositioning process can be understood as warping the superwavelet onto the new test faces.

2.7 The Relation between Bunch Graphs and GWNs

Another approach to object representation that is also based on Gabor wavelets is the well-known *elastic bunch graph* approach [Krüger *et al.*, 1996; Maurer and von der Malsburg, 1995; Wiskott *et al.*, 1997]. The underlying idea is that a face (or, more generally, an object) is represented by a set of specific, meaningful feature points. At each of these feature points in the image, 40 complex Gabor filters are applied. This gives 40 complex coefficients for each feature point, a so-called *jet*. A collection of such jets together with information about their relative locations constitutes a *bunch graph*. A single jet describes the gray value in a small local neighborhood around a feature point. The filter set is fixed and contains Gabor filters that are parameterized for eight different orientations and five different frequencies.

The elastic bunch graph approach is inspired by the discrete wavelet transform, where, in contrast to the continuous wavelet transform, the phase space is discretized. How to sample the phase space is a major problem in this context and has been widely studied [Daubechies, 1988; Daubechies, 1990; Daugman, 1988; Grossmann and Morlet, 1984; Lee, 1996; Mallat, 1989a; Mallat, 1989b]. In general, the discretization scheme depends on the selected wavelet function. Lee [Lee, 1996] studied how closely the phase space has to be sampled in order to achieve a lossless wavelet representation of an image when a non-orthonormal Gabor function is used as wavelet. He found that one needs at least eight equidistant orientation samples and five equidistant scale samples for each discrete position. We see that this justifies the choice of

40 Gabor filters in [Wiskott *et al.*, 1997]. However, we also see that an image representation using 40 wavelets per pixel is highly redundant and is only practical if it is reduced to a small set of feature points in the image. A bunch graph representation usually contains about 20 jets with 800 complex coefficients.

The reason for this highly redundant representation is that local image structure such as edges, lines or junctions needs to be *approximated* by a weighted sum of these 40 Gabor filters in the *discrete* phase space. Alternatively, one can model the local image structure directly by selecting the correct wavelet parameters in the *continuous* phase space. This is the underlying idea of GWNs. As shown above, 52 Gabor wavelets were sufficient for good representation of a facial image (compared, e.g., with the bunch graph approach, where a comparable representation needs many more wavelets).

2.8 Conclusion

In this chapter we have introduced GWNs. We have explained that these networks are optimized on the objects they are supposed to represent. Furthermore, we have observed, that two GWNs that are optimized on the same object usually appear to be different. We have therefore introduced distance measurements that allow us to calculate similarities between GWNs. We have also introduced a measure that allows us to calculate the similarity between two vectors of wavelet coefficients that are computed with respect to the same wavelet family. A further topic that was discussed is the reparametrization of a GWN: When we wish to calculate wavelet coefficients using the wavelet family of a certain GWN, on an object that is similar to the object on which the GWN was optimized, a good reparameterization of the GWN on the new object is very important.

Several important properties have not yet been discussed, such as the interpretation of the weights w_i, the role of the number of wavelets N in a GWN, and the relation between the wavelet parameters and their weights and filter responses. These topics will be discussed in the next chapter.

Chapter 3

Properties of Gabor Wavelet Networks

In the previous chapter we gave an extensive introduction to GWNs. In this chapter we will discuss properties of GWNs that have not yet been treated, including

- the relation between the parameterization of a wavelet in the image and its weight and filter response, as well as the interpretation of the weights with respect to the image,
- the role of the number of wavelets N of a GWN,
- how GWNs can be used for optimal filtering of an image.

These important properties of GWNs will be discussed in the following three sections. In the next three chapters the advantages of these three properties will be systematically investigated and exploited in real applications.

3.1 Feature Representation with Gabor Wavelets

Gabor wavelets are recognized to be good feature detectors [Manjunath and Chellappa, 1993; Mehrotra *et al.*, 1992], and especially for $\sigma\omega_0 < 2$, they are well-known filters for edge detection [Michaelis, 1997]. We would like to ask whether this property can be exploited for our needs and whether it has consequences and advantages for the representation of an object with these filters. Consider, e.g., the images in Fig. 3.1. One can see that the wavelets with the largest weights are positioned along the object edges, i.e. at positions where their filter responses are large. In this section we will investigate whether this observation can be generalized. In particular, we want to analyze how the final optimized parameters of wavelets and their weights are linked to their filter responses and how this is related to the property that Gabor functions are good edge detectors.

3.1.1 Relation between Filter Responses and Weights

The results of the optimization of a GWN depends partly on the mother wavelet function that is used and partly on the optimization procedure itself. Therefore, in order to understand what the optimization results express, we have to consider both the mother wavelet and the optimization procedure. In the following we will use the terminology of discrete images without loss of generality. If not stated otherwise, the discussion can be adapted for continuous functions as well.

Recall from Section 2.3 that the difference image $D = I - \hat{I}$ is given as the pixel-wise difference between a DC-free image I and its reconstruction $\hat{I}$. Assume now that all the weights w_i of the reconstruction are zero: $w_i = 0$, from which follows $\hat{I} = 0$, so that the difference image D equals the original image I. This is the case at the beginning of the optimization procedure. As optimization progresses, one wavelet after another is considered. Each wavelet is optimized so that it approximates a local region in I as well as possible. Afterwards, it is subtracted from I. The difference image D shows the parts of the image that remain to be approximated. The values of the image D are small in local regions that are already approximated.

The optimization procedure parameterizes a wavelet, so that the energy of the difference image, $\|D\|_2^2$, is minimized. This is the case if

1. the local structure in the difference function D that adds the largest portion to the energy $\|D\|_2^2$ is approximated,

2. the wavelet approximates the local structure in the difference image D optimally.

Referring to point 2, it can be shown that the inner product (correlation) of a Gabor wavelet with the difference image D at a position $\mathbf{y}$ is maximal iff the local structure there is approximated optimally by the wavelet $\psi_{\mathbf{n}}(\mathbf{x})$, i.e. the energy is minimized:

$$\begin{aligned} & \sum_{\mathbf{x}} \left(\psi_{\mathbf{n}}(\mathbf{x}) D(\mathbf{y}+\mathbf{x})\right) = \max \\ \text{iff} \quad & \sum_{\mathbf{x}} \left(\psi_{\mathbf{n}}(\mathbf{x}) - D(\mathbf{y}+\mathbf{x})\right)^2 = \min \ . \end{aligned}$$

This shows that the parameter vector $\mathbf{n}$ that leads to a maximal filter response in D is the same as the one that leads to an optimal approximation.

When the optimization starts, D and I are very similar, and the above holds for both. This is the reason why the first wavelets fit well to edges in the image I (see Fig. 3.1, bottom left). As the optimization proceeds, D and I become more and more different and the wavelets no longer fit the image edges of I (top right).

It is a property of the odd Gabor function as an edge detector that the filter has a clear response peak when it shows the precise location and orientation of an edge, and that the response decreases quickly with increasing distance from the edge [Canny, 1986]. Accordingly, the energy $\|D\|_2^2$ is minimized when the Gabor wavelet shows the precise location and orientation of the edge it has to approximate. The energy $\|D\|_2^2$ increases as the distance to the edge increases.

Concerning point 1, the filter response of an odd Gabor function is strong at "strong" edges. The stronger the edge, the larger must be the weight w_i which weights the wavelet $\psi_{\mathbf{n}}$ that is supposed to approximate the edge. The relation between the filter responses and the weights is given by the linear eq. (2.36): A maximal filter response leads (without loss of generality) to a maximal weight. Clearly, the optimization procedure is best able to minimize the energy $\|D\|_2^2$ when a strong edge is approximated by a wavelet. Therefore, the wavelets with the largest weights minimize the energy best. It is possible to define an *order of importance* for the reconstruction. According to the above discussion, one can use the weights, but they should be normalized with respect to the scale of the wavelets (see Section 3.2).

An example of this can be seen in Fig. 3.1. This figure shows the image of a toy wooden block (top left) on which a GWN was trained. The top right image shows the positions, scales and orientations of the wavelets as short black line segments. The first wavelets that are positioned in the image are, according to our discussion, positioned along the edges (see bottom left image). Since the edges are already approximated, the contributing gray values of the wooden block get smaller in the difference image. The newly appearing edges in the difference image again have to be approximated by a new set of wavelets. This can be seen clearly in the top right image, where many wavelets are positioned parallel to the edges. It can also be seen that the energy of the difference image, $\|D\|_2^2$, becomes smaller as the number of wavelets increases. This results in smaller filter responses.

By thresholding the weights, the more "important" wavelets can be selected, which leads to the bottom left image. Since large weights indicate that the corresponding wavelets represent edge segments (high filter responses), these wavelets encode local geometrical object information.

Because of the direct relation between the filter responses, the weights and the optimization results, different mother wavelet functions result in different wavelet networks. The choice of the odd Gabor function as the mother wavelet induces a model for the representation of local image primitives; here, edge segments locally model object edges. In fact, the odd Gabor wavelets introduce the only prior knowledge into this representation. The introduction of a model for local image primitives is the reason for the considerable data reduction that can be achieved with GWNs. In fact, representation of "subject01" (see Fig. 2.4) with 52 wavelets

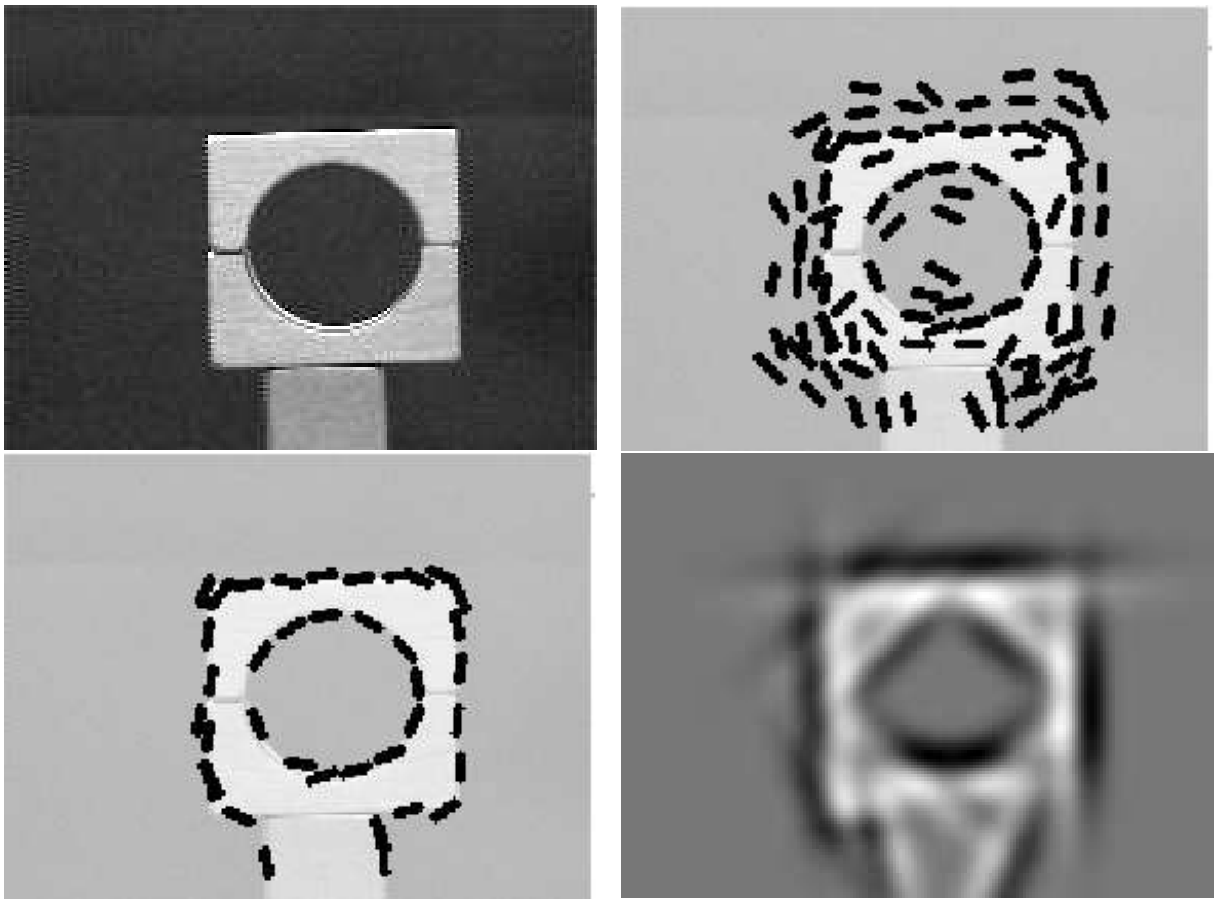


Figure 3.1. This figure shows images of a wooden toy block (top, left) on which a GWN was trained. The black line segments sketch the positions, sizes and orientations of all the wavelets of the GWN (right) and of some automatically selected wavelets (bottom left). The bottom right image shows the difference image D between the original image and the approximation by the wavelets in the bottom left image.

needs $52 \times 6 \times 4$ bytes $= 1248$ bytes. Since the original image has 78784 bytes, this corresponds to a data reduction of 98.4%

The sparseness is a property that will be exploited in Chapter 5 for the recognition of faces. We will also discuss this property in more detail there.

Other mother wavelet functions (models for local image primitives) have been tested, such as the Gaussian and its derivatives [Pelc, 1997]. These functions are often used as radial basis functions in RBF networks [Bishop, 1995]. It is interesting, however, that these models have proven to be much less effective.

Figure 3.2. These images show (from left to right) images $\hat{I}_{16}$, $\hat{I}_{52}$, $\hat{I}_{116}$ and $\hat{I}_{216}$, which represent image I with 16, 52, 116 and 216 Gabor wavelets, respectively.

3.2 Variation in Precision

One important property of GWNs is their ability to vary the precision with which an image can be represented: The more Gabor wavelets are used, the more precise the representation becomes. An example is shown in Fig. 3.2. There, a GWN $(\Psi, \mathbf{w})$ with $N = 216$ wavelets has been optimized on the rightmost image. Using

$$\hat{I} = \sum_{i=1}^{M} w_i \psi_{\mathbf{n}_i} \ ,$$

the precision of the representation varies with $M = 16, 52, 116, 216$. It can be seen that for $M = 116$ wavelets a good representation of the image can already be achieved. The order of the wavelets in this example corresponds to the order in which they were optimized. In Section 2.3, a pyramid scheme for the optimization of several wavelet pyramid layers was introduced. The example images in Fig. 3.2 correspond to these pyramid layers: The first image (from the left) shows only the first pyramid layer, the second image shows the first two pyramid layers, etc.

In Fig. 3.3 the wavelets are used according to the sizes of their weights in decreasing order. The weights are normalized with respect to the wavelet scale.

Fig. 3.4 shows a graph that quantitatively represents the information in Fig. 3.2 and Fig. 3.3: It can be seen that the energy decreases much faster when the wavelets are chosen according to their normalized weights.

For any $\epsilon > 0$ one can find an N such that

$$\| f - \sum_{1}^{N} \psi_i w_i \|_2^2 < \epsilon$$

Figure 3.3. These images show (from left to right) the reconstructions of Fig. 3.2 with 16, 52, 84, 116 and 180 wavelets. The wavelets are chosen according to the sizes of their weights, starting with the largest one.

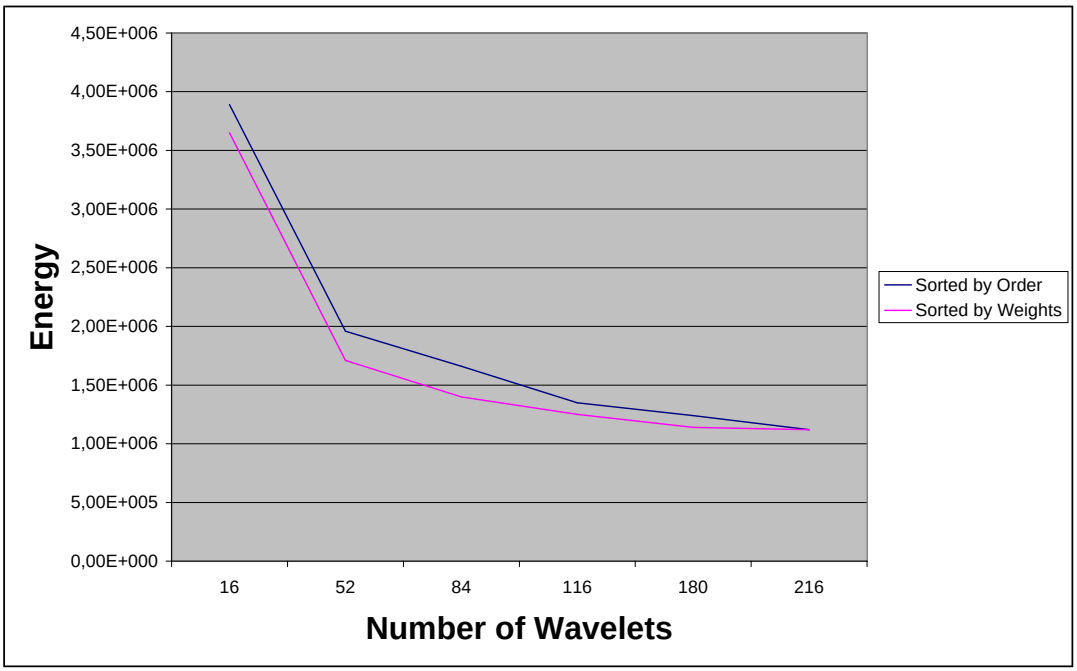


Figure 3.4. In this graph, the decrease in energy is plotted as the number of wavelets is increased in the order in which they were optimized (top) or in order of the sizes of their weights (bottom).

is satisfied. This property is a major property of the discrete wavelet transform [Mallat, 1989b; Mallat, 1989a], exploited especially for multi-resolution analysis. GWNs inherit this property from the discrete wavelet transform.

In the same manner, assuming that for a certain $\epsilon > 0$ a GWN $(\Psi, \mathbf{w})$ of a sufficiently large size N is given, i.e. $\dim(\Psi) = N$, we can find M wavelets, $1 \leq M \leq N$, such that

$$\|f - \sum_{1}^{M} \psi_i w_i\|_2^2 < \epsilon_0 \text{ for any } \epsilon_0 > \epsilon.$$

Given a GWN, we can decide how much information we want to use by varying M. This property of variable precision will be discussed in greater detail in the next chapter, where we will introduce the term *progressive attention* to refer to this property.

3.3 Gabor Wavelet Networks for Optimized Image Filtering.

The weights of GWNs are linearly related to the filter responses of the wavelets. This means that the weights can be computed solely from the filter responses. Furthermore, this means that in the filter responses, all the data that is needed to represent and reconstruct the image is already encoded.

The fact that the filter responses already contain all the image information is due to the filter scheme, given by the GWN, that is being used. This can be explained as follows: The wavelet family Ψ of a GWN defines the basis for the sub-space $< \Psi >$. An image I of the image space can be approximately represented by a vector from this vector space. How lossy the mapping is from the image space into the sub-space depends on the basis Ψ. Given an optimized GWN, the loss is minimized for a certain image. The corresponding vector of the vector space is calculated through the filter responses of the wavelet functions. It can clearly happen that if another wavelet basis is used, the loss for that image is very high.

Consider, e.g., a family of Gabor wavelets (which does not necessarily define a basis) that contains four differently oriented filters at each position of a homogeneous 4×4 grid. Mapping the well known "subject01" into the sub-space $< \Psi >$ through eq. (2.39) allows us to visually verify the loss of image data that occurs (see Fig. 3.5).

The mapping through eq. (2.39) is an orthogonal projection. It leads to the optimal vector for the given wavelet family. Therefore, the mapping is as good as it can get. The quality of the mapping is limited by the given wavelet family Ψ, and consequently by the filtering scheme.

The same experiment can be repeated with up to 8 different orientations and with a grid of up to 8×8 homogeneously distributed positions. The results can be seen in Fig. 3.5. One can see that the loss of image data is very high, taking into account that 64, 128, 256 and 512 (!) filter responses were used. This loss appears especially severe when one compares these images with the images in Fig. 3.2, where only 16, 52, 116 and 216 Gabor wavelets were used.

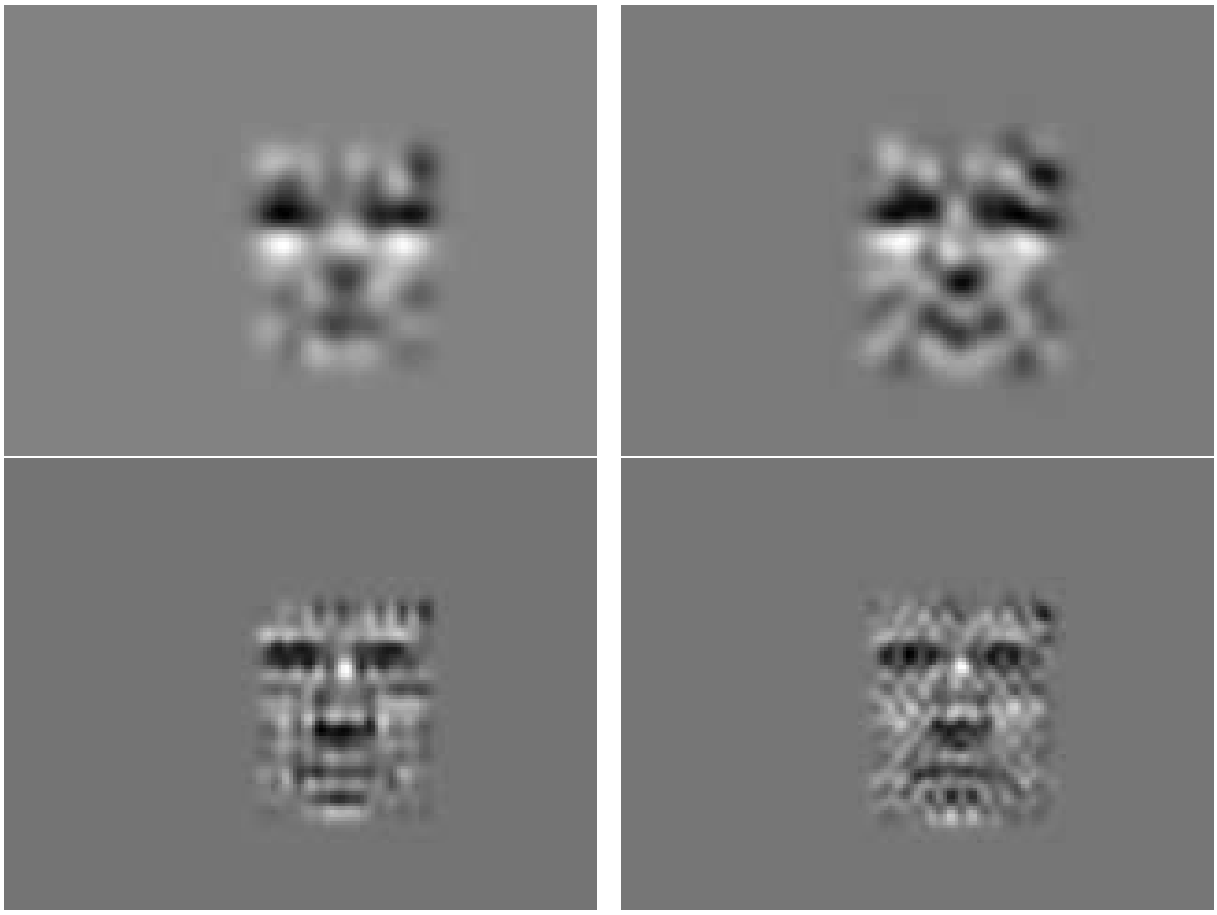


Figure 3.5. These images show, qualitatively, what image information is contained in a set of Gabor filter responses, when the filtering is done with (from left, top to right, bottom) 4×4 homogeneously distributed Gabor filters with 4 and 8 orientations, or with 8×8 homogeneously distributed filters with 4 and 8 orientations.

These experiments show very clearly that the amount of data that can be extracted from an image through filtering depends heavily on the filtering scheme. In particular, an optimized filtering scheme is able to encode more image data than a non-optimized filtering scheme. Furthermore, our experiments show that a GWN offers an optimized filtering scheme that allows a maximal amount of information to be extracted from the image.

The linear relation between the optimal weights and the filter responses and the optimized filtering property of GWNs will be discussed and exploited more precisely in Chapter 6.

3.4 Conclusions and Comments

In this chapter we have discussed three important and fundamental properties of GWNs. First, we discussed the relation between the parameters of a wavelet, its weight, and its filter response. Furthermore, we argued that Gabor functions are able to model local features in an image. In fact, the correlation between the filter and the difference image is maximized where the energy of the difference image is minimized. This shows that there is a precise relation between the original image and the parameters of the optimized wavelets.

Gabor functions are good edge detectors that show strong maxima when they are correctly parameterized; their filter response decreases quickly as the parameterization changes. This means that the ability to model local image structure also decreases quickly when the parameterization is different from the optimal one. Therefore, a GWN that is optimized for one image is not likely to be good for another image. In Chapter 5 we will investigate how individual this representation really is. There we will study, in a face recognition experiment, whether each representation has enough individuality to be able to distinguish between various persons.

Second, we have discussed the possibility of varying the precision of a GWN by changing the number of wavelets used. The variation can be done, e.g., with respect to a given task, which allows control of evaluation speed, representation precision, etc. In Chapter 4, we will exploit this variability for tracking. There we will investigate how this variability allows us to control the tracking speed as well as the tracking precision, which degrades when the number of wavelets is decreased.

Third, the linear relationship between weight and filter response is most important in order to understand that the wavelets of the network, when used as filters, provide a “handle” on the image data. This means that for the same task, the set of optimized filters may be much smaller than the set of non-optimized filters. This increases efficiency. In Chapter 6, we will exploit the optimized filtering scheme for a gaze detection application. We will compare a non-optimized filtering scheme with an optimized one and investigate how performance, stability and computation speed increase.

Chapter 4

Progressive Attention for Real-Time Tracking

The fundamental idea of active vision systems [Aloimonos, 1993; Sommer, 1995] is that they are autonomous systems that take part in their environment. This means that they have to keep track of surrounding events while remaining focused on achieving their task. This implies two things:

1. selective perception, in order to
   (a) achieve a given task,
   (b) keep track of possible distractors that might disturb the vision system in achieving the task.
2. taking actions that are dependent on the task and on the perceived visual information.

According to [Aloimonos, 1994], *perception* has to be related to *action*: An *active vision system* is an active observer which has control over the image acquisition process and which perceives (image) information that is relevant to what it intends to do [Aloimonos, 1994]. *Perception* here means the information acquisition and selection process and the control strategies that are applied to it [Bajcsy, 1992]. *Action* is anything that changes the state of the system or the environment. Both perception and action are dynamic processes that depend on the current state of the data interpretation and the goal or task of the vision system.

Consider the following example: A robot that is supposed to follow another robot through a group of people has to “concentrate” on the leading robot, while it has to “keep an eye” on people that may get in its way. The robot that is following has to “concentrate” its attention on the leading robot:

- It has to recognize the leading robot so that it does not follow someone/something else by mistake;
- It has to determine the position and heading direction of the leading robot precisely enough so that it can follow on a direct path.

In other words, precise information about the leading robot is needed.

At the same time, our robot has to attend to other persons (i.e. distractors) that are about to move in its way by detecting their approximate positions and possibly their approximate heading directions so it can navigate around them. Depending on the distance and motion of each destructor relative to the intended path of our robot, more approximate information may be sufficient or more precise information may be needed.

In other words: "Concentration" on the leading robot is necessary, but a "quick glance" at the distractors is sufficient. The amount of information that needs to be extracted from the camera images should correlate with the needed degree of precision.

Active vision systems [Aloimonos, 1994] have, among others, the following properties:

- purposive: they use resources purposively to solve a problem.
- selective: they use a minimal amount of information, i.e. they separate relevant from irrelevant information and use only the information that is relevant to solve the problem.

This was previously pointed out by [Bajcsy, 1992], who mentioned that the problem of active sensing "can be stated as a problem of control strategies applied to the data acquisition process which depends on the current state of the data interpretation and the goal or task of the process".

It is agreed that (image) information representation is of major importance in active vision systems [Aloimonos, 1993; Aloimonos, 1994; Bajcsy, 1992; Brown, 1994; Jain, 1994]. The Marr paradigm [Marr, 1982] uses general representations that would allow it to solve *any* problem. The Marr paradigm implies a bottom-up representation: first the information, then the algorithms and solutions. This means that the image acquisition is independent of the algorithms, and the algorithms are not able to acquire more information later. Selective sensing, as proposed e.g. by Sandini and Brown [Aloimonos, 1993; Brown, 1994], implies, on the other hand, a top-down approach, in which information selection is purposive: first the solutions, then the information. Systems may retrieve information in a single, general purpose form, and leave it to cognitive modules to transform the information according to their needs. In [Aloimonos, 1994] it was pointed out, however, that visual systems should directly produce forms of information that suit specific cognitive processes. This conforms exactly with the selectivity property, and it is also important in order to assure high computation speed.

Returning to our example, we may ask whether our robot needs the same type of representation of the leading robot (which it is interested in) and of the distractors (which it is not particularly interested in) in order to define its next action. As the active vision paradigm suggests, the perception should be related to the action. When our robot has to concentrate on the leading robot, a precise representation should be employed, because the related action is to recognize the leading robot and to follow it on a direct path. For the surrounding distractors, only an approximate representation is needed, because the only task is to avoid them.

Two important questions arise here:

1. A fundamental problem is to determine what image representation an active vision system should use. This problem has to be solved by the programmer nowadays, using prior knowledge about the set of tasks the system will have to perform. In our example, the task of the system is to track and calculate the precise and approximate states of the leading robot and the distractors.

2. Another fundamental question is what information from the image should be used. This depends on the task the system has to carry out and on the state of the system and the environment. The question is how information should be represented so as to allow the robot to relate the representation to the task and to decide *how much* and *what* information should be used.

The ability of the system to relate action to perception, i.e.

- to decide *what* image information,
- *how much* image information is needed, and
- *how precise* this image information has to be,

will be called *progressive attention*. This term was adapted from [Zabrodsky and Peleg, 1990], who used it in the context of image coding.

In this chapter we will use the properties of the active vision paradigm and the progressive attention scheme as guidelines in constructing a system that is able to track efficiently and with variable precision. The tracking system is designed as a perception-action cycle. It relies on its internal state, which reflects its present situation in the scene, and on the images that are recorded by a camera. Tracking is considered here as a low-level task of a higher-level vision system. The higher-level system is assumed to define the level of precision for the tracking task.

4.1 Related Work

Progressive attention is related to *incremental focus of attention (IFA)* for tracking [Toyama and Hager, 1996] and to the attentive processing strategy (GAZE) for face feature detection [Herpers *et al.*, 1995]. Both of these were inspired by [Tsotsos, 1990] and relate features to scales by using a coarse-to-fine image resolution strategy. *Progressive attention*, on the other hand, should not relate features to scales but to the object that the features describe. In this sense, as stated above, the object is considered as a collection of image features, and the more information about the object is needed to estimate its state, the more features are extracted from the image. To realize *progressive attention* we will use GWNs for object representation.

4.2 Foundations and Definitions

The paradigmatic starting point of the tracking algorithm presented in this chapter is the *perception-action cycle* (PAC). Intuitively, this cycle is a fusion of perception and action; no precise definition has yet been given for it. In our definition we will follow [Sommer, 1997]:

Definition 5
A vision system is realized as a perception-action cycle *when it is able to fulfill its task based solely on a sequence of live camera images and the task it is constructed for.*

In order to define *active vision-based object tracking*, we will follow the definition of the perception action cycle.

Definition 6
Given a visually perceivable target object together with its initial state s_0, the **active vision-based object tracking task** *is to estimate the state s_t of the object at each time step t, given a live image I_t of the object and the previous state s_{t-1}.*

A *target* is a visually observable, not necessarily physical or rigid entity. A *state* s of the target object is given as a finite vector that quantifies certain qualities of the target object, such as the object's projected position in the image, its position in space, its projected size in the image, its orientation in space, and its projected orientation in the image. Shape parameters, velocity etc. may also be included [Toyama, 1997]. One may differentiate between the projected true state of the object in the 3-D scene, which is referred to as the *true state* $\hat{s}$, and the observed/estimated state of the object, which is called by the *observed state* s. The true state should ideally equal the observed state.

The definition of *active vision-based object tracking* implies several things:

- When tracking starts, the true initial state $s_0 = \hat{s}_0$ is given.
- Using the live image I_t and the previous observed state s_{t-1} implies that the state s_{t-1} is "updated" to the new observed state s_t so that the expected squared error between $\hat{s}_t$ and s_t is a minimum:

$$E[(s_t - \hat{s}_t)^T (s_t - \hat{s}_t)] = \text{minimum} \ .$$

- The difference between two successive live images I_{t-1} and I_t, and therefore the difference between two successive states s_{t-1} and s_t, depends on the tracking speed. A higher tracking speed results in smaller state differences. Successive states are therefore more more accurately recovered, the closer they are in time. [Brown and Terzopoulos, 1994; Leondes, 1966].

The last point is clear from a statistical point of view: the correlation between two successive true states $\hat{s}_{t-1}$ and $\hat{s}_t$ decreases with increasing separation in time. This fact is extensively discussed (among others) in [Brown and Terzopoulos, 1994; Leondes, 1966]; further discussion is beyond the scope of this thesis.

4.3 Tracking with Gabor Wavelet Networks

In this section we will give details about the tracking system. This system is strictly appearance-based and is realized as described in Definition 6.

Recall for a moment Section 2.6: There, a GWN $(\Psi, \mathbf{w})$ was interpreted as a superwavelet $\Psi_{\mathbf{n}}$:

$$\Psi_{\mathbf{n}}(\mathbf{x}) = \sum_{i=1}^{N} w_i \psi_{\mathbf{n}_i}(\mathbf{SR}(\mathbf{x} - \mathbf{c})) \ , \tag{4.1}$$

where N is the number of wavelets used and $\mathbf{n}$ is the parameter vector of the superwavelet.

The introduction of the term *superwavelet* had the advantage that the GWN $(\Psi, \mathbf{w})$ could be understood as a single wavelet and could consequently be deformed accordingly by optimization of the superwavelet parameters $\mathbf{n}$:

$$E \ = \ \min_{\mathbf{n}} \|g - \Psi_{\mathbf{n}}\|_2^2 \ . \tag{4.2}$$

The operator cP as introduced in in Section 2.6 was defined as

$$\begin{aligned} \mathcal{P}_f : \mathbb{L}^2(\mathbb{R}^2) \ &\longmapsto \ \mathbb{R}^5 \\ g \ &\longrightarrow \ \mathbf{n} = (c_x, c_y, \theta, s_x, s_y) \ , \end{aligned} \tag{4.3}$$

and computed the vector $\mathbf{n}$ that minimizes the energy functional E of the above equation between the input image g and the superwavelet $\Psi_{\mathbf{n}}$.

This technique can be enhanced for gray-value image sequences J_t. In this case, (4.2) can be rewritten as

$$E = \min_{\mathbf{n}_t} \|J_t - \Psi_{\mathbf{n}_t}\|_2^2 \, . \tag{4.4}$$

In other words,

$$\mathbf{n}_t = \mathcal{P}(J_t) \, , \tag{4.5}$$

so that for frame J_t at time step t the superwavelet $\Psi_{\mathbf{n}_t}$ is optimized with respect to the energy functional (4.4). As explained above, good initialization is needed, and the better the initialization, the faster the convergence. In this tracking approach, $\mathbf{n}_{t-1}$ is taken to be the initial value for $\mathbf{n}_t$. Therefore, as argued above, $\mathbf{n}_{t-1}$ can be considered as good initial values if the temporal sampling rate is high enough.

In other words, the superwavelet $\Psi_{\mathbf{n}}$ is used as a template, and the minimization procedure finds the best "fit" of this template to the input image. In eq. (4.1), the pair $(\Psi, \mathbf{w})$ was interpreted as a superwavelet. The number N of wavelets is given here by the size of the wavelet family Ψ. However, one can replace the N in eq. (4.1) by any M with $1 \leq M \leq N$. As shown in Section 3.2, this allows variation in the precision of the template that is used for tracking. Decreasing M results into a speedup of the minimization process, as well as in an error surface that has fewer, broader, and more shallow minima. We realized the progressive attention scheme by controlling the number M of wavelets used.

The progressive attention principle assumes that for each number of filters M, the M most important wavelets are selected from the set of N wavelets in the wavelet family. The order of importance clearly depends on the given task. In this chapter, the task is appearance-based visual tracking on the basis of a given template. The order of importance is therefore given in this case by the ability of the wavelets to minimize the energy function (4.4). This was already explained in Section 3.2. We have found that the wavelets that minimize the energy function (4.4) best also minimize the energy function (2.23) best. We therefore define their order of appearance by their ability to minimize the energy function (2.23).

It should be pointed out that the minimization in eq. (4.4) is able to converge stably only when all face features are visible. Otherwise, background may easily cause failure of the minimization process.

4.4 Experimental Results

In this section we will present and discuss experiments on affine face tracking with GWNs. This will accomplish three things:

1. It will show that the proposed tracking method works in principle.

2. It will allow us to discuss the progressive attention principle, and will show how tracking precision varies with a change in the number M of filters.

3. It will allow us to discuss how stable our proposed tracking is when we use the active vision approach. As explained above, the active vision approach implies the ability to use, at each time step t, a novel camera image I_t and a present state s_{t-1} to estimate the new state s_t. A stable algorithm has to be able to cope with large state differences.

4.4.1 Testing Tracking on Various Image Sequences

In this subsection we test the proposed approach on various test image sequences. All these test sequences show a person in motion. The face of the person is always visible and always more or less frontal to the camera so that the facial features are always visible. The experiments were carried out on off-line standard encoder test sequences such as "salesman", "claire" or "miss_america", and also on on-line image sequences. To record the on-line image sequences, our active face tracker [Krüger *et al.*, 1999] was used.

For tracking in the off-line sequences, the GWN $\hat{I}_{16}$ of Fig. 2.4 was used as a superwavelet. This GWN contains 16 Gabor wavelets.

For the on-line tests we used networks with 14 wavelets that were trained on a face image of the tracked person.

Example frames from the tracking results on the salesman sequence are shown in Fig. 4.1. The white boxes in the images denote the detected position, orientation and scale. Ideally, the white box should always frame the inner region of the face. It can be seen in some examples in Fig. 4.1 that the white box is too large, which indicates "incorrect" estimated state parameters. It can also be seen that "incorrect" estimated states did not cause the tracking to fail; the tracking was successful throughout the entire test sequence. "Incorrect" state estimation occurs here because the set of wavelets used is small and because the wavelets were optimized on a different face. The term "incorrect" is used here in quotation marks; "imprecise" is the term that should have been used instead. What is observed here is the principle of *progressive attention*: The

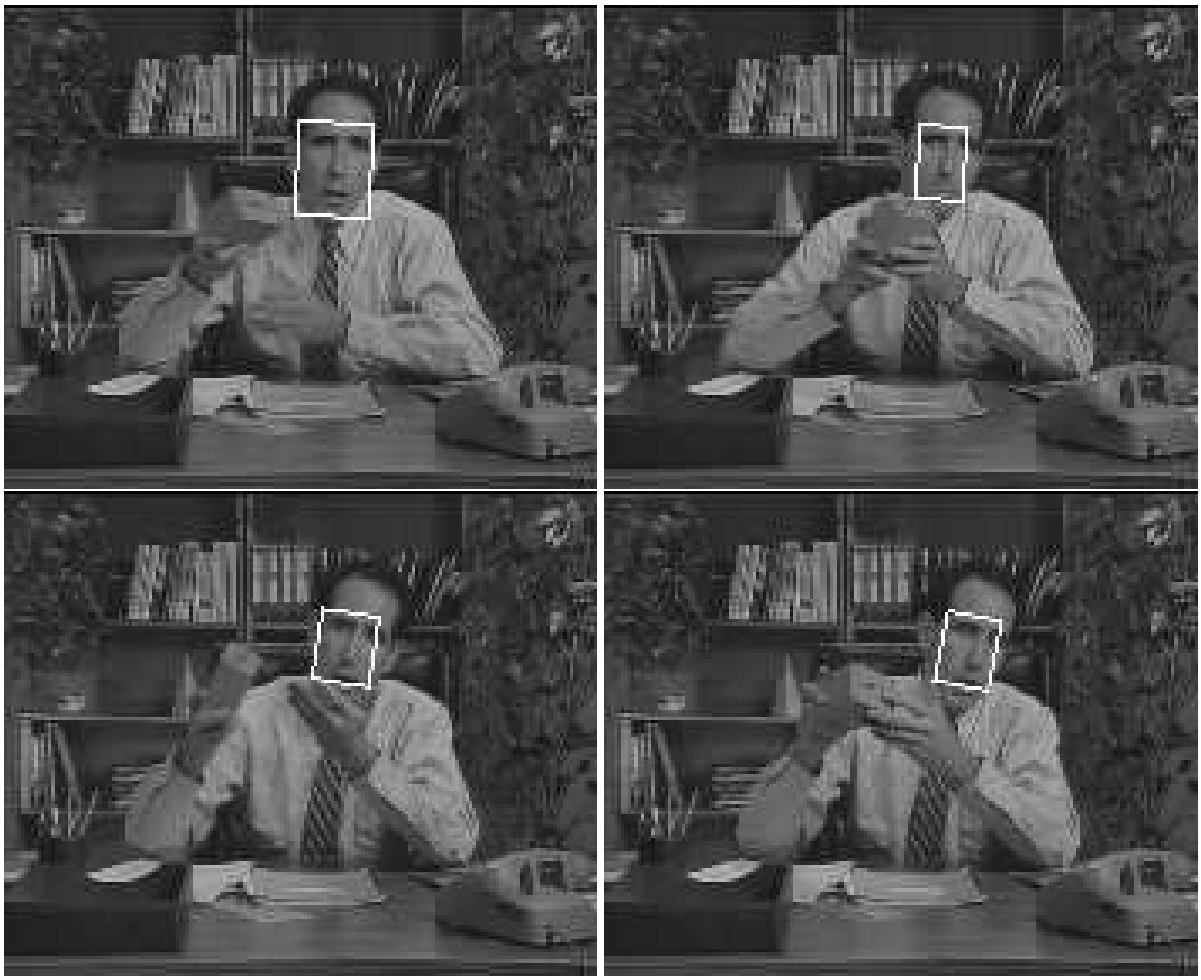


Figure 4.1. These images show (top left to bottom right) frame 11, frame 50, frame 120 and frame 137 of the salesman sequence.

algorithm converged toward the correct minimum, but stopped too early, as the minimum was too shallow.

The experimental results on the other off-line test sequences are similar and are omitted here. The images in Fig. 4.2 show tracking results on an on-line sequence. It can be seen that the white box, again marking the inner face region, is positioned very precisely in this example. The reason is that the GWN was trained on the face of the tracked person. In this example a GWN with 14 wavelets was used.

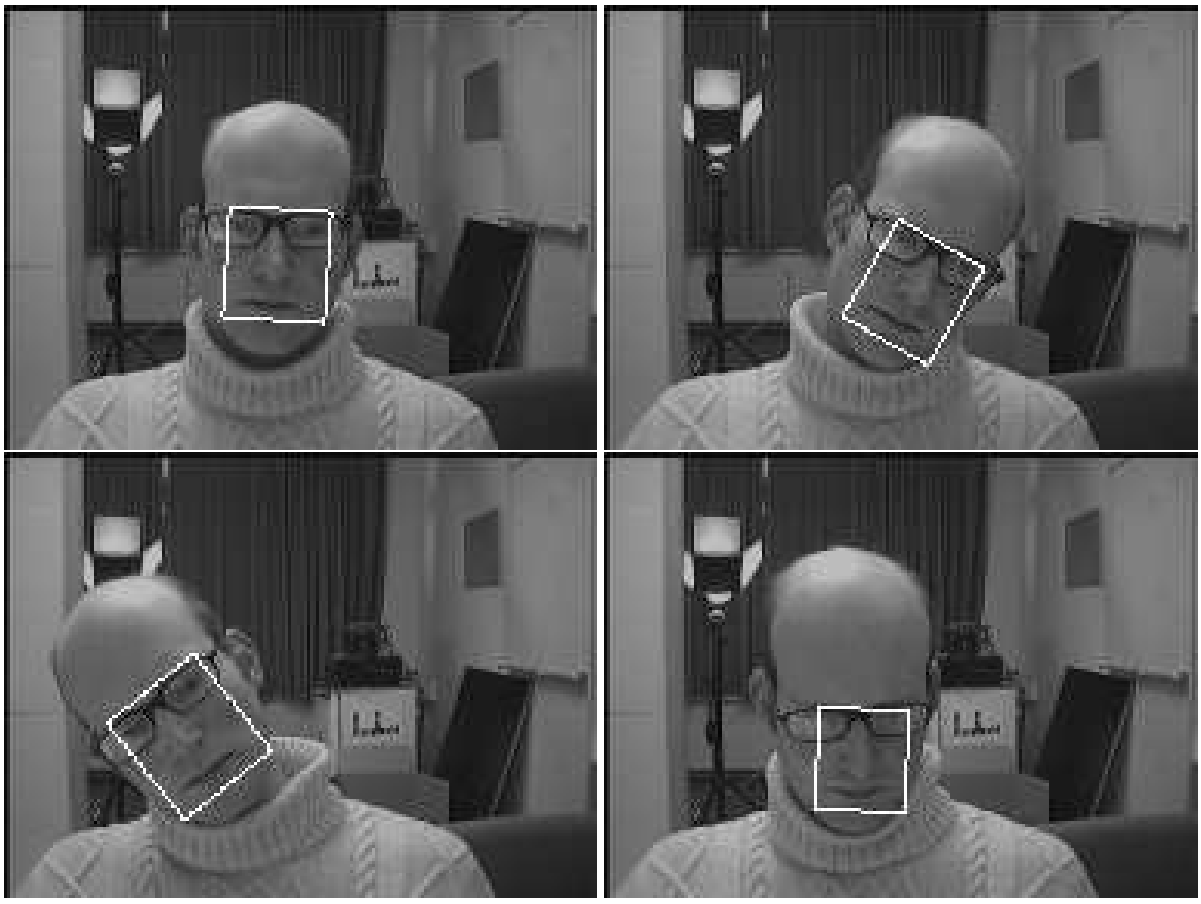


Figure 4.2. These images show snapshots of an on-line experiment.

4.4.2 Evaluation of Tracking Precision

In this subsection we discuss how the tracking precision depends on the number M of wavelets used, so that we can quantify the progressive attention principle. For this purpose we recorded an image sequence of a person who is sitting in front of a computer. The video camera that is used for recording the images is positioned on the computer monitor. Sample images are shown in Fig. 4.3. The image sequence has a length of 18 seconds, i.e., 450 images. A GWN with 116 Gabor wavelets was trained on the face of that person. In order to investigate the progressive attention principle of our tracking approach, GWNs were used that contained only the largest 8, 9, 12, 14, 24 and 33 wavelets, sorted according to decreasing normalized weight. Remaps of these GWNs are shown in Fig. 4.4. In the experiments presented here, we wanted to find out how precisely the parameters of the superwavelet can be found when the number of wavelets in the superwavelet is varied. To do this we used these six GWNs as superwavelets

Figure 4.3. These figures show sample images from the test sequence used in this subsection. This sequence was used to investigate the progressive attention principle of our tracking approach. Shown (left to right, top to bottom) are images 10, 64, 175, 219, 254, 307, 335, 356 and 382.

Figure 4.4. The figures show remaps of the GWNs used in this experiment. These GWNs contain, from the left, 116, 33, 24, 14, 12, 9 and 8 wavelets.

and showed how precisely the superwavelet parameters can be found with each of the six small GWNs. For clarity we reduce our presentation to the estimation of the superwavelet parameters x-position, y- position, and orientation θ.

First, we used the large GWN with all 116 Gabor wavelets to estimate a "ground truth", i.e. the best possible parameter estimation. For the estimation of this "ground truth" full-resolution images were used. The estimation is consequently relatively slow and runs at approximately 1 Hz. The "ground truth" will be denoted in the graphs in Figures 4.5 through 4.10 by the dashed line. In these figures it can be seen that the x-position of the face in the sequence images varied from ≈ 100 to ≈ 300 (in pixel coordinates), the y-position varied from ≈ 135 to ≈ 160, and the angle θ varied from $\approx -20^\circ$ to $\approx 30^\circ$. In all the graphs, the x-axis indicates the frame number and the y-axis indicates the estimation results for the x-, y-, or θ-parameter. The frame numbers indicated in the caption of Fig. 4.3 are related to the frame numbering of the x-axis. An upright head is indicated by $\theta = 0$. The head is positioned initially (frame number 1) at image position $x = 233$ and $y = 137$.

We used six GWNs with varying numbers of Gabor wavelets to estimate the superwavelet parameters x-position, y-position and angle θ. The estimation results are shown in the graphs in Figures 4.5 through 4.10 as a solid line. In the graphs in Figures 4.5 and 4.6, the estimates of the x-position are shown for the six GWNs. In the graphs in Figures 4.7 and 4.8, the estimates of the y-position are shown for the six GWNs. Finally, the graphs in Figures 4.9 and 4.10 show the six estimates of the angle θ. In all the graphs it can clearly be seen that the more wavelets are used the less noisy the estimated parameters are. For the large GWN with 33 Gabor wavelets, the estimation results are close to the "ground truth". In the top graph in Figures 4.5, 4.7,4.9, tracking results with just 8 Gabor wavelets are shown. At approximately frame 330, the tracking failed, as can be clearly noticed in the graphs.

4.4.3 Robustness of the Tracking Approach with Respect to Object Speed

In order to calculate the robustness of the approach with respect to speed variations, we calculated the visible speed of the head from the estimated "ground truth" from the previous subsection. The displacement of the tracked object between two successive frames (speed) is given here as the sum of squared differences (SSD) between the estimated head positions in the two frames: If par(t, x) and par(t, y) denote the estimated x- and y-position parameters for frame t then the SSD(t) for frame t, is given by

$$\text{SSD}(t) = \sqrt{(\text{par}(t,x) - \text{par}(t-1,x))^2 + (\text{par}(t,y) - \text{par}(t-1,y))^2} \ . \tag{4.6}$$

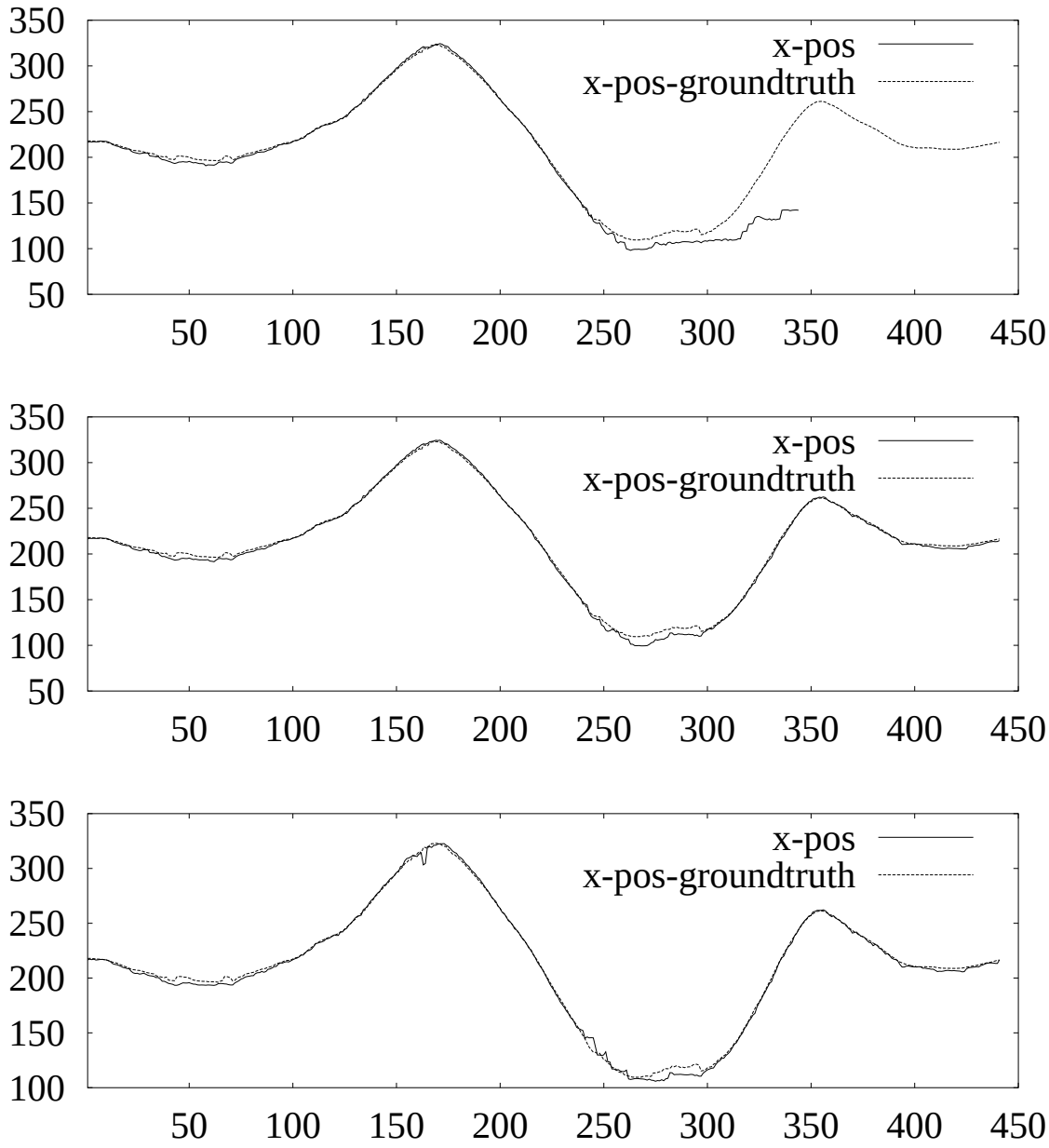


Figure 4.5. These figures show the change in the x direction. The solid line is the ground truth. The dotted lines are the estimated results with 8 (top), 9 (center), and 12 (bottom) wavelets. The x-axis indicates the frame number, the y-axis the estimated x coordinate.

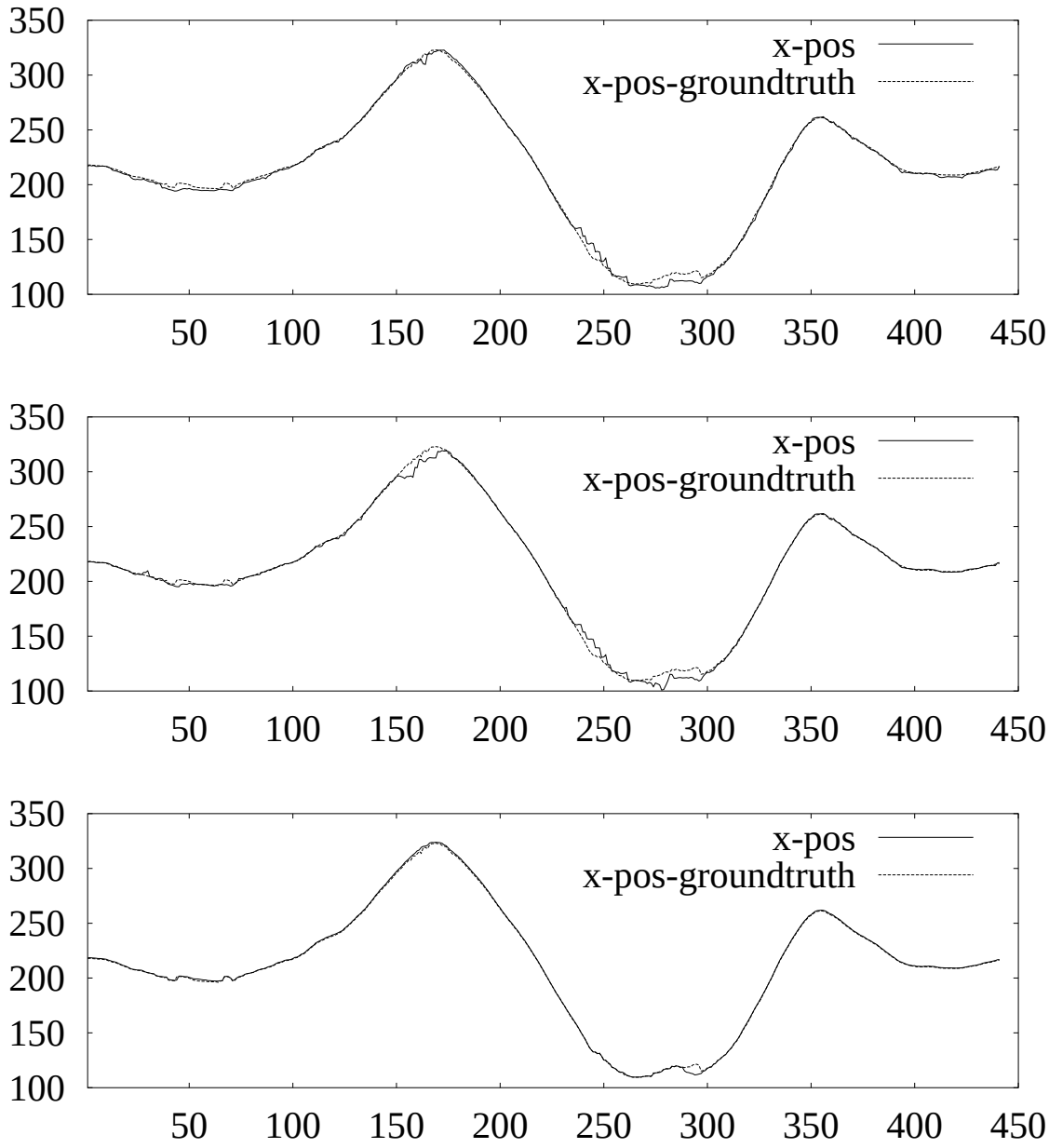


Figure 4.6. These figures show the change in the x direction. The solid line is the ground truth. The dotted lines are the estimated results with 14 (top), 24 (center), and 33 (bottom) wavelets. The x-axis indicates the frame number, the y-axis the estimated x coordinate.

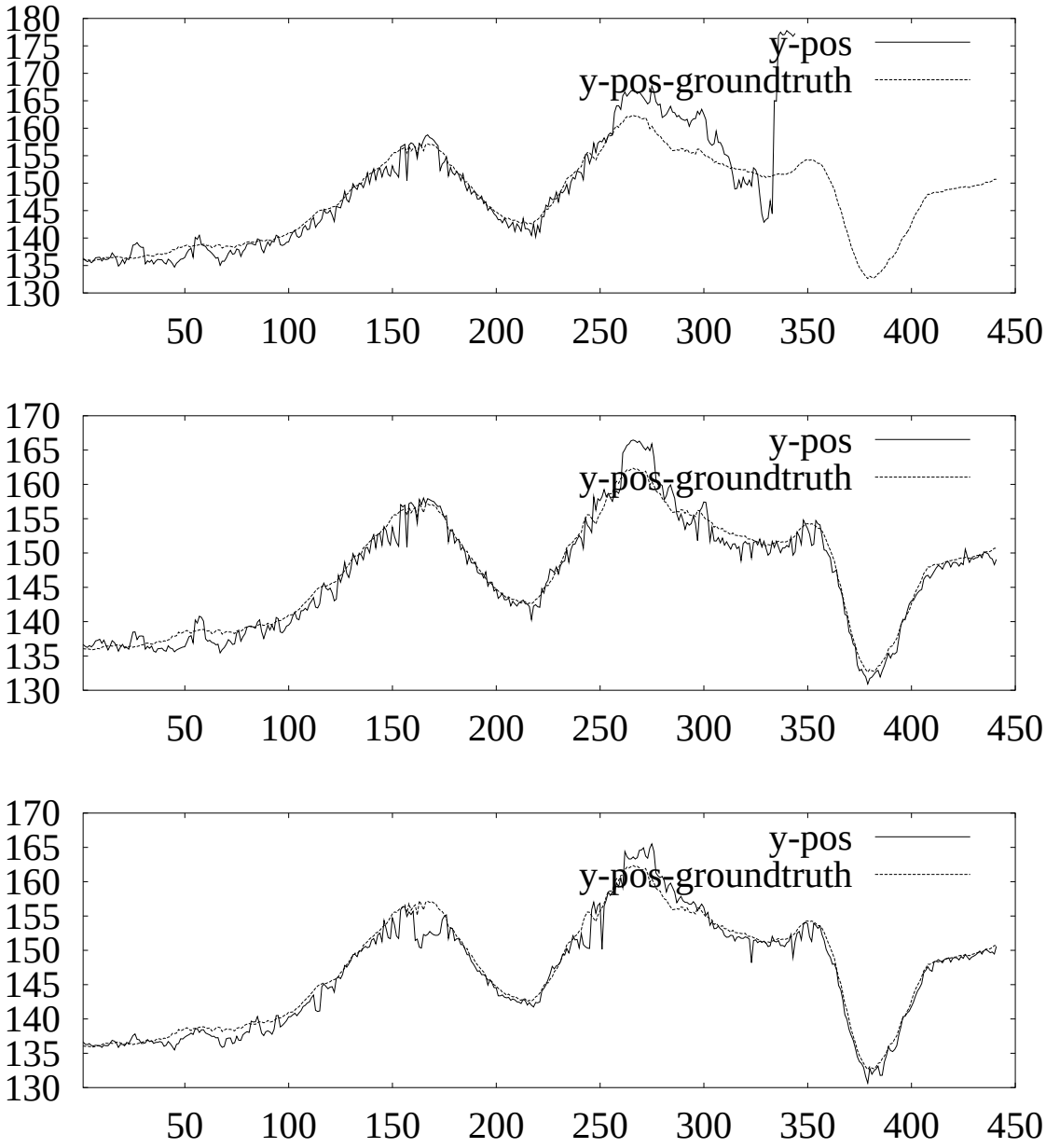


Figure 4.7. These figures show the change in the y direction. The solid line is the ground truth. The dotted lines are the estimated results with 8 (top), 9 (center), and 12 (bottom) wavelets. The x-axis indicates the frame number, the y-axis the estimated y coordinate.

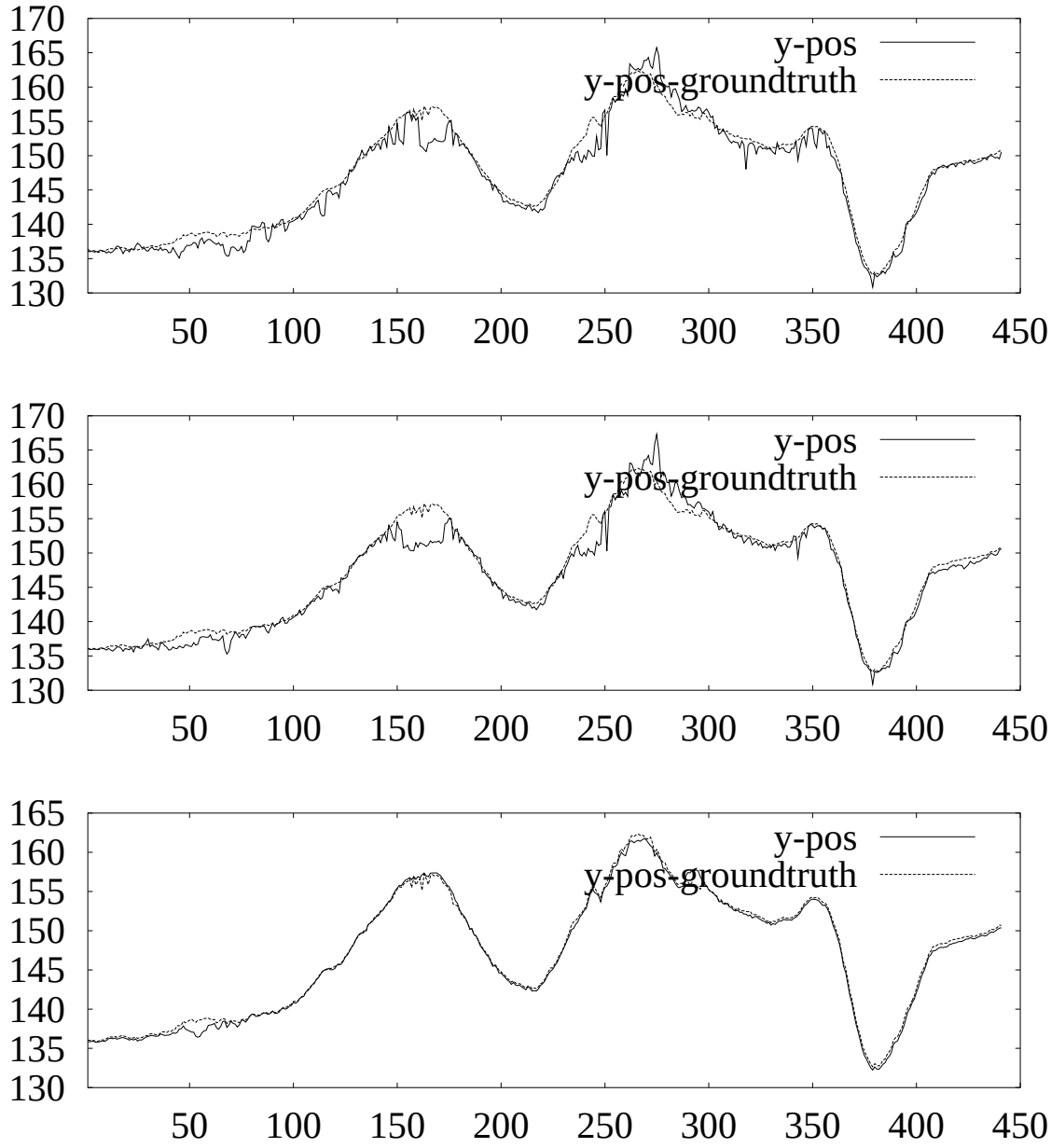


Figure 4.8. These figures show the change in the y direction. The dashed line is the ground truth. The solid lines are the estimated results with 14 (top), 24 (center), and 33 (bottom) wavelets. The x-axis indicates the frame number, the y-axis the estimated y coordinate.

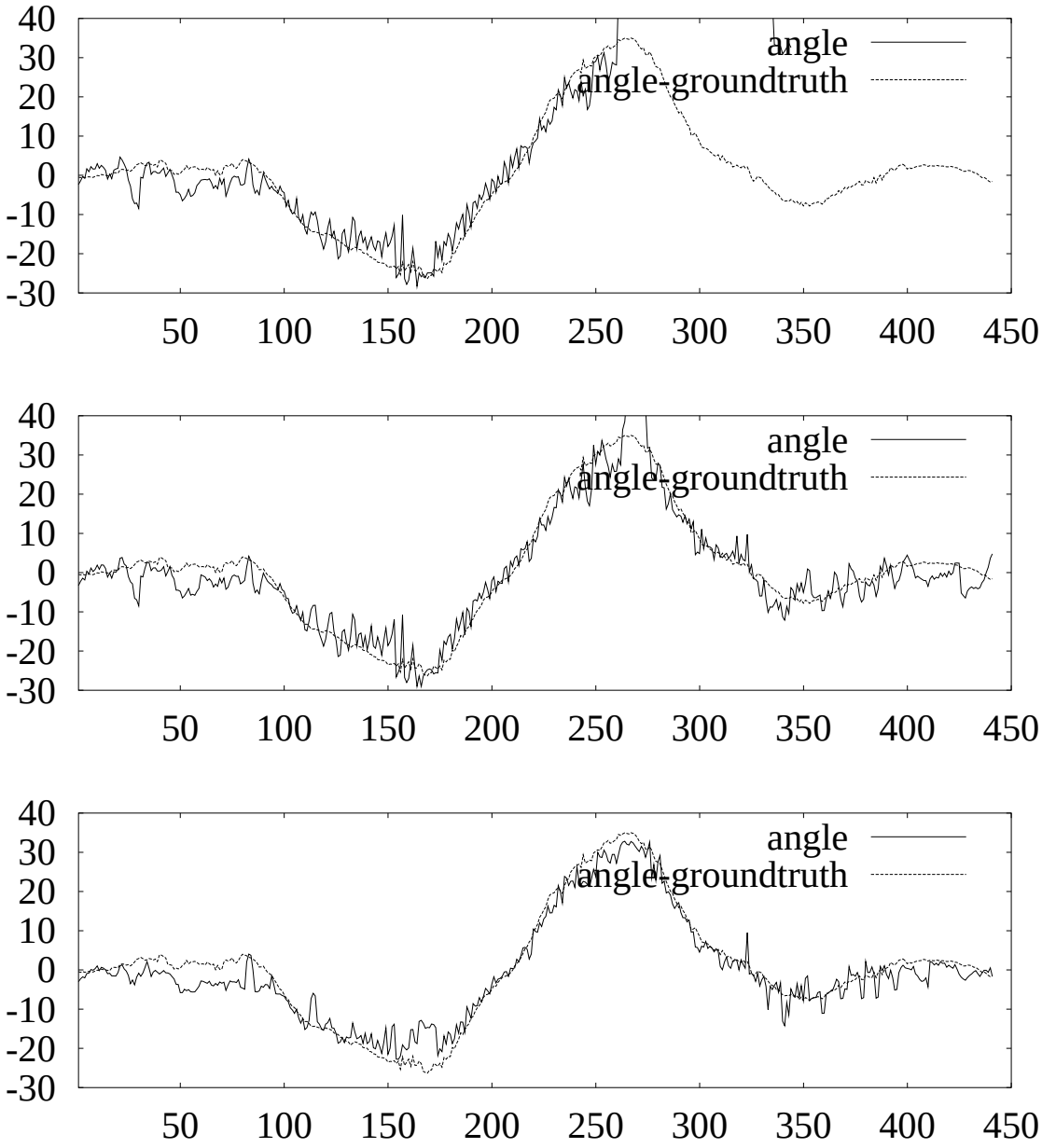


Figure 4.9. These figures show the change in the θ direction. The dashed line is the ground truth. The solid lines are the estimated results with 8 (top), 9 (center), and 12 (bottom) wavelets. The x-axis indicates the frame number, the y-axis the estimated angle θ.

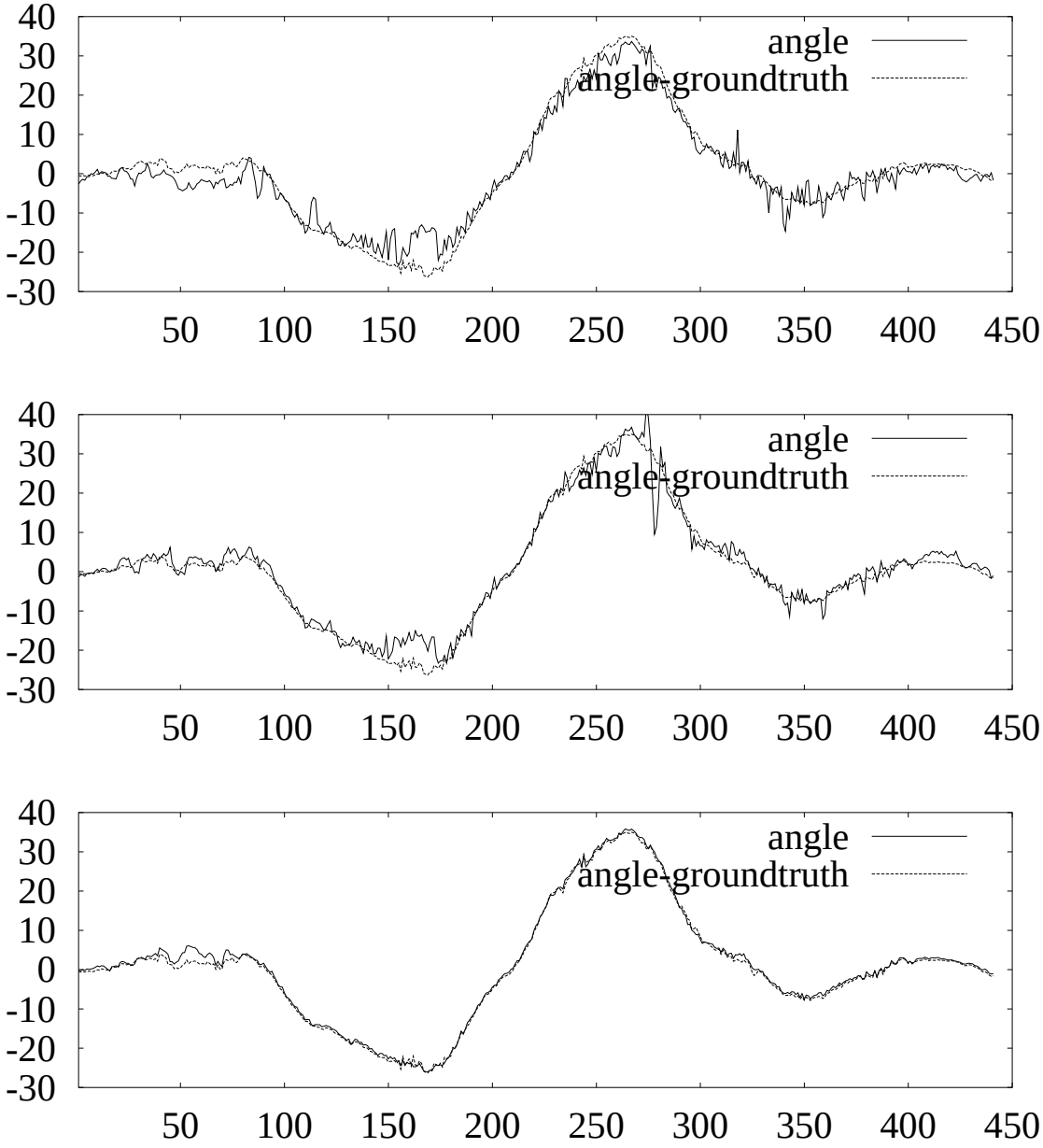


Figure 4.10. These figures show the change in the θ direction. The dashed line is the ground truth. The solid lines are the estimated results with 14 (top), 24 (center), and 33 (bottom) wavelets. The x-axis indicates the frame number, the y-axis the estimated angle θ.

The object displacements for each frame of our test sequence are plotted in the top graph in Fig. 4.11. In the plot one can see how much time the tracker needed to compute each new state. As mentioned above, we use the same Levenberg-Marquardt (LM) method for the computation of each state s_t as we used for finding optimal wavelets. The number of evaluation steps of the LM method depends on the distances of the initial value from the local minimum. The bottom graph indicates that the LM method needed only two cycles most of the time. Of course, a higher number of cycles indicates a slower tracking speed. The bottom graph in Table 4.11 shows the number of cycles for GWN I_{14}, but the results were similar for all other tested GWNs. In order to relate the number of cycles to a "real" speed, given in milliseconds, the reader should refer to Fig. 4.12. In this graph, the approximate computation speed per LM cycle, given in milliseconds, is plotted with respect to a variable number of Gabor wavelets. The speed was computed on a 450 MHz Linux Pentium.

Clearly, this plot should increase monotonically. Internal micro-processor architecture, cache, and compiler optimization, however, resulted in a not strictly monotonic curve.

4.5 Discussion and Conclusions

In this chapter we have shown how GWNs can be used for affine real-time face tracking. For this we exploited several principal advantages of GWNs:

1. GWNs have the advantage that they can be arbitrarily translated, rotated, scaled and sheared. This is because GWNs are given by a discrete linear combination of continuous Gabor wavelets. By following the active vision principle, for tracking at each time step we used solely the actual state of the system and a novel live image to compute the new system state. We have shown using various test image sequences that this approach works satisfactorily.

2. By following the progressive attention principle we varied the number of wavelets that were used to describe the face. When fewer wavelets are used, the tracking becomes imprecise; when more are used, the tracking becomes more and more precise.

3. Finally, we have discussed how the tracking speed changes with the number of wavelets used. We have argued that the evaluation time increases with the number of wavelets that have to be computed. Also, we have investigated how many gradient decent cycles the tracker needed for the sample image sequence.

The results with respect to this last point are difficult to generalize because

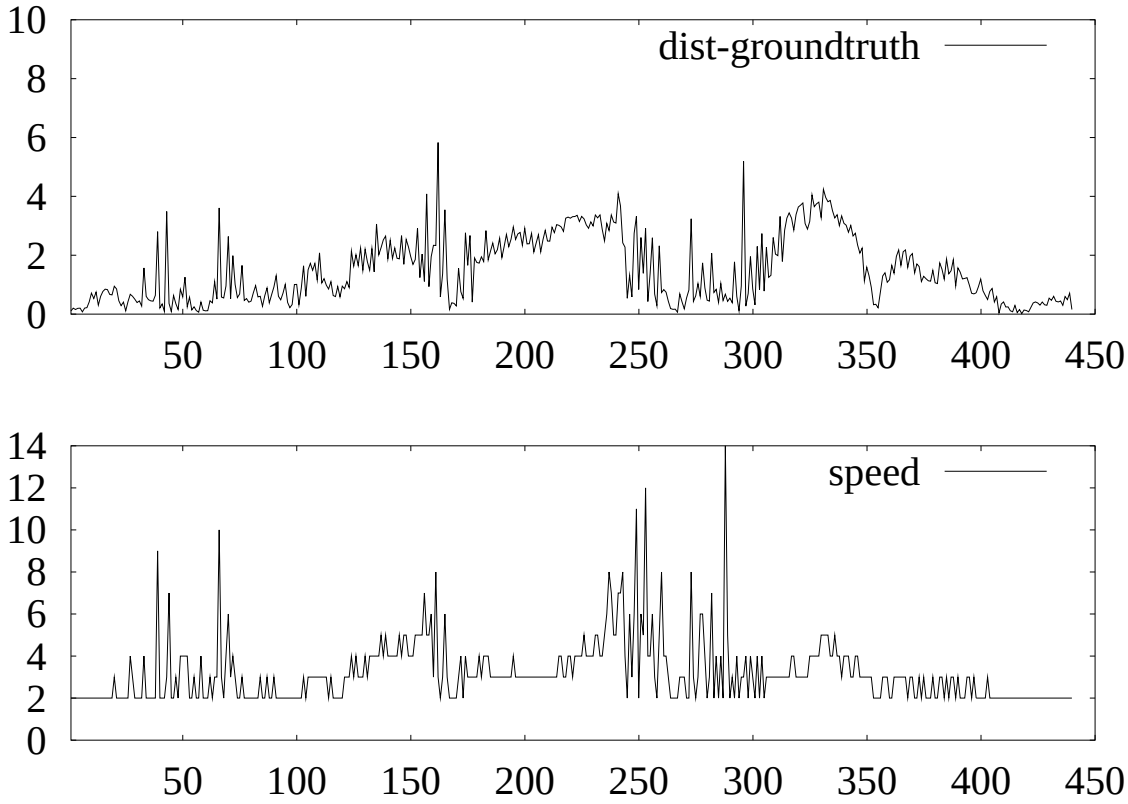


Figure 4.11. The top figure indicates the speed of the head as estimated by the GWN with 133 Gabor wavelets. The x-axis indicates the frame number, the y-axis the sum of squared difference (SSD) in position between two successive frames. Higher values indicate higher differences. The bottom graph shows the speed for detection of state s_t from state s_{t-1} and the novel image I_t. Lower values indicate higher speed. The graphs were computed with $\hat{I}_{14}$, but they look similar for all other $\hat{I}$.

1. exact timing results depend heavily on the underlying hardware and operating system;

2. we have presented computational results only for one image sequence. Even though all the other test sequences we have used led to similar results, it is still not possible to draw any generally valid conclusion.

The bottom graph in Fig. 4.11 shows that the tracker usually needed between two and four cycles for the computation of each new parameter vector $\mathbf{n}_t$. For a wavelet network with 14 wavelets this resulted into a speed of between 10 and 20 frames per second on a 450 MHz Linux

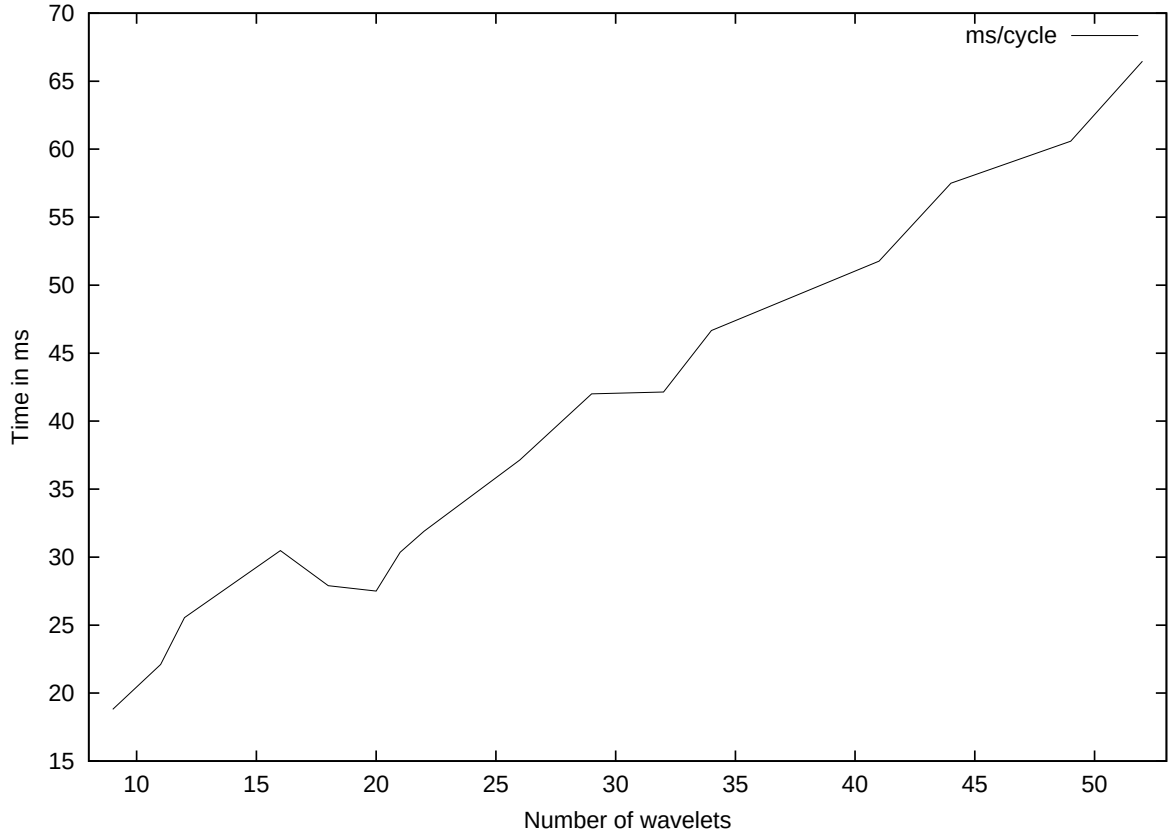


Figure 4.12. This plot indicates the speed of a single LM cycle with respect to a variable number of Gabor wavelets. The y-axis indicates the speed in ms and the x-axis indicates the number of Gabor wavelets used. Speed was computed on a 450 MHz Linux Pentium.

Pentium. For some frames the speed was much slower. In these situations, either the number of wavelets in the network could have been reduced, resulting in a speedup, in order to assure a frame evaluation in real time; or the tracking would have failed. At frame-number ≈ 280, even with as few as 10 wavelets, the computation time would have exceeded 280 ms, which would have caused the tracking system to fail. How the number of cycles can be controlled better, in order to keep the probability of tracking failure low, could be subject of further research.

Chapter 5

Image Coding for Automatic Face Recognition

As we have already explained in the introduction and in Chapter 4, how image and object information should be encoded is a major question. This question is the subject of extensive research in computer vision and robotics, and is still notoriously difficult to answer. How effectively the image data is exploited by the representation in order to fulfill the given task depends on the encoding scheme. Also, the image representation determines the distance measurements and the efficiency of successive processing steps. In other words, the image representation provides a "handle" on the image information: The relevant image information is contained somewhere in the image, and it is the image representation that allows the relevant information to be selectively extracted from the image. The term *relevant* depends on the specific task.

In this section, we will do two things:

1. We will show how Gabor Wavelet Networks can be used to distinguish between different objects. In this connection, we will take the problem of *automatic recognition of faces* as a challenge. Below we will present an introduction to the terminology used and major problems involved in face recognition.

2. We will use this application to illustrate

   (a) how image data is represented,

   (b) how unique the representation is to each represented object, and

   (c) how and to what extent generalization can be achieved.

Automatic recognition of faces is a challenge because the variety of possible images of a single person is huge:

- A face may be imaged from different viewpoints,
- it may be illuminated by different light sources in different directions,
- it may appear differently because of beards, glasses and hairstyles, and
- most importantly, facial appearance varies considerably because of facial expressions and age.

Given a facial image, an automatic face recognizer either has to output the correct identity of the individual in the image or reject the person as "unknown to the system". A face recognizer has a stored set of face images of different individuals that defines its *knowledge*. Each stored individual is encoded according to a predefined encoding scheme. When a new face image is input to the recognizer, the image is encoded and compared to each of the stored individuals. The recognizer then identifies or rejects, based only on its knowledge, the person in the image. The ability of the recognizer to cope with all possible face variations while avoiding mis-identifications clearly depends on the image representation and encoding scheme. Different face recognition approaches [Brunelli and Poggio, 1993; Cootes *et al.*, 1998; Edelman *et al.*, 1992; Moghaddam and Pentland, 1997; Turk and Pentland, 1991; Wiskott *et al.*, 1997; Belhumeur *et al.*, 1997; Zhao *et al.*, 1998] differ mainly in the image representation that they use. Examples of these are various versions of principal component analysis (the Karhunen-Loeve transform) [Turk and Pentland, 1991; Moghaddam and Pentland, 1997; Edelman *et al.*, 1992; Cootes *et al.*, 1998; Belhumeur *et al.*, 1997; Zhao *et al.*, 1998], the Gabor jet representation [Wiskott *et al.*, 1997], or a feature-based representation [Brunelli and Poggio, 1993]. In this chapter we will discuss the capabilities of GWNs as an image representation and coding scheme. In Chapter 2 we gave an extensive introduction to GWNs: We gave the relevant definitions and distance measurements, and we gave an extensive discussion of the advantages of the GWN representation over other object representations. These advantages will now be studied for face recognition purposes, and it will be verified step by step that GWNs can satisfy the invariance requirements stated above.

We will begin in Section 5.1 with an introduction, including foundations, preliminaries, and important terms. We will then give an overview of related work in Section 5.2. In Section 5.3 we will investigate evaluation topics.

Existing face recognition systems [Brunelli and Poggio, 1993; Cootes *et al.*, 1998; Edelman *et al.*, 1992; Moghaddam and Pentland, 1997; Turk and Pentland, 1991; Wiskott *et al.*, 1997;

Belhumeur *et al.*, 1997] propose different solutions to the problems of viewpoint, illumination, expression, etc., and a natural concern is the overall performance of these systems. Although each researcher has reported recognition results for his system, the results depend heavily on the chosen test set of face images, and cannot be regarded as a basis for comparison of the approaches. Indeed, the selection of the particular collection of faces on which to carry out tests is probably the greatest source of variability, yet it is the least relevant one. The FERET database [Phillips *et al.*, 1998] may eventually provide a standard, but is only now becoming widely available, and it does not claim to test recognition over a comprehensive set of transformations.

In order to avoid these difficulties we will use the Yale Face Database as a fixed face image database for our discussion. Other databases, e.g., the Manchester face database, have also been used in our experiments and have confirmed the results that will be presented. The images will not be preprocessed, unless stated otherwise. This will allow us to do several things:

1. It will allow us to investigate how images are encoded and what image data is encoded. This will allow us to avoid erroneous conclusions about the properties of the representation.

2. It will allow us to directly compare the various image representation approaches and encoding schemes, as results obtained on this database using other approaches are known [Belhumeur *et al.*, 1997].

5.1 Foundations and Preliminaries

In this section we will give an introduction to the foundations, terminology and methods of machine-based face recognition. The terminology sometimes varies between authors. We will use the terminology introduced in [Phillips *et al.*, 1998].

In machine-based face recognition, we basically deal with two sets:

1. A *gallery set* $\mathcal{G}$ is a set of known individuals. More formally, a *gallery* $\mathcal{G}$ is a set of image sets. Each image set in $\mathcal{G}$ is associated with a specific individual and consists of all possible face images of that individual, including all the variations mentioned above (pose, illumination, expression, etc.).

   This definition implies that each set in $\mathcal{G}$ is very large. In practice, a small set of representative images of each individual is used, and the missing examples have to be interpolated. How well the missing examples can be interpolated depends on the choice of the image representation.

2. A *probe set* $\mathcal{U}$ is a set of images of unknown persons. The *probe images* in the probe set are presented to the recognition system and are to be identified or rejected.

We say that "a probe is in the gallery" when there exists a set in the gallery in which the probe image is contained. The person in the probe image is then identified to be the individual associated with that particular gallery set.

We mentioned above that ideally the gallery sets should be very large. In order to cope with this potentially large set size, image representations are used that are invariant with respect to the possible image variations. If we could imagine an image representation that was *perfectly* invariant with respect to these variations, each set could consist of a single image, encoded using that representation. In this sense, each image set in the gallery is mapped by the representation into a single, specially coded image.

During the "optimization phase" of a face recognition system, the gallery images are coded using the system's image representation. As just mentioned, in the ideal case the representation would project every image of a given individual onto a single image. Therefore, instead of needing to encode every image in each set, a single image from each set may be sufficient. This means that for each individual who is known to the system, only a single image would need to be supplied.

In practice, however, the assumption of perfect invariance does not hold. Some representations have invariance properties with respect to some types of variation, but do not show any invariance to other types of variation. Therefore, a larger set of images per person may be needed. We will give some examples of this in the next section when discussing related work.

During the "matching phase", a probe image is input to the recognizer. The recognizer encodes the probe image using the same representation as was used for the gallery images. A distance measure, specific to the image representation, can then be calculated to determine the differences between the probe image and the gallery images. These differences are used to decide on identification or rejection.

Three different types of automatic face recognition problems exist:

- the closed-universe recognition problem,
- the verification problem, and
- the open-universe recognition problem.

In a *closed universe*, every probe is contained in the gallery, i.e. every probe person is known to the system: $\mathcal{U} \cap \left(\bigcup \mathcal{G}\right) = \mathcal{U}$. The closed-universe recognition problem is the problem of identifying the individual in the probe image, assuming that the probe is in the gallery.

For effective performance evaluation of closed-universe recognition, the question is not always whether the top match is the correct match. Instead, the question of whether the correct match is one of the top N matches provides an indication of how many images have to be examined in order to get the desired recognition performance. The quality of the representation and recognition results can be measured by considering the top N matches. The trade-off between the size of N and the fraction of the times that the correct face is included in the top N matches measures the performance of the system.

The *verification problem* is usually considered as an *open-universe* problem. In an *open universe*, some probes may not be included in the gallery, i.e. some probe persons may not be known to the system: $\mathcal{U} \cap \bigcup \mathcal{G} \neq \mathcal{U}$. For verification, a probe image and an identity are given as inputs to the system. The identity is assumed to refer to a gallery face. The system now has to verify whether the identified individual is the same as the one in the probe image.

The degree of similarity between the gallery image and the probe image is used to decide on recognition or rejection. A threshold for the similarity measure is usually given in advance by hand, or can be learned during a training phase of the system. A higher threshold makes the system more conservative. In this context, one speaks of *false negatives* and *false positives*. *False negatives* are probe faces that do indeed correspond to the correct identity, but are still rejected because of a threshold that is too high. *False positives*, on the other hand, are probe faces that are accepted as having the supposed identity even though they do not. This occurs when the threshold is too low. Clearly, one wants to minimize false positives as well as false negatives. In real verification systems acceptance bounds are usually set conservatively, and a user may be asked to alter his pose or expression if the verification fails. If the user is indeed the person he claims to be, the verification system will eventually decide correctly, possibly after several (false) rejections.

Another problem model is the *open-universe problem*, which is defined with respect to an *open universe*. As in the closed-universe problem, a probe image is input to the recognition system. Here, however, the probe person need not be known to the system. An open-universe recognition problem can be solved by solving a *verification problem*. The system tests, for each face in the gallery, the hypothesis that the probe face is the same as the hypothesized gallery face. Again, careful selection of the similarity threshold is important in keeping the false positive and false negative rates low.

5.2 Principles of Automatic Face Recognition and Related Work

Various approaches to automatic face recognition exist. The currently best-known approaches are based on principal component analysis (PCA) and bunch graphs. Other approaches, such as ones in which recognition is based on the geometry of local face features [Brunelli and Poggio, 1993], are recognized to be less capable. In the following two subsections we will describe the two best-known approaches.

5.2.1 Principal Component Analysis

One of the best-known image representation approaches used for object and face recognition in computer vision is principal component analysis (PCA) [Jolliffe, 1986; Turk and Pentland, 1991; Sirovitch and Kirby, 1987; Murase and Nayar, 1995]. PCA is also known as the *Karhunen Loeve Transform* [Loeve, 1955; Kirby and Sirovich, 1990]. PCA has been particularly successful in face recognition and has received considerable attention in this context.

Formally, PCA is defined as follows: Let $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ be a set of n-dimensional gallery images. We want to define a linear, orthonormal mapping $W \in \mathbb{R}^{n \times m}$ from the n-dimensional image space into an m-dimensional feature space, with $m < n$. Using W, a new feature vector $\mathbf{y}_k \in \mathbb{R}$ can be calculated for each image $\mathbf{x}_k$:

$$\mathbf{y}_k = W^T \mathbf{x}_k \ , \quad k = 1, \ldots, N \ . \tag{5.1}$$

The covariance (scatter) matrix C is defined as

$$C = \sum_{k=1}^{N} \left(\mathbf{x}_k - \mu\right)\left(\mathbf{x}_k - \mu\right)^T \ , \tag{5.2}$$

where N is the number of gallery images and $\mu \in \mathbb{R}^n$ is their mean image. From eq. (5.1) one sees that the covariance (scatter) matrix of the new feature vectors $\{\mathbf{y}_1, \ldots, \mathbf{y}_N\}$ is $W^T C W$. In PCA the linear mapping W_{PCA} is chosen such that the covariance matrix of the feature vector is a diagonal matrix and that the determinant of $W^T C W$ is maximized:

$$\begin{aligned} W_{PCA} &= \arg\max_W \left| W^T C W \right| \\ &= \left[\mathbf{w}_1, \ldots, \mathbf{w}_m\right] \ . \end{aligned} \tag{5.3}$$

The set $\{\mathbf{w}_1, \ldots, \mathbf{w}_m\}$ is the set of n-dimensional eigenvectors of the covariance matrix C that correspond to the m largest eigenvalues. The eigenvectors have the same dimensionality as the sample images; they are therefore often referred to as eigenpictures [Sirovitch and Kirby, 1987] or eigenfaces [Turk and Pentland, 1991].

The use of PCA for automatic face recognition has been very successful. However, its theoretical foundations are not really clear:

1. PCA is a linear transform. Using PCA for face recognition assumes that the space of face images is a linear space, i.e. it is assumed that the variations in facial images caused by different expressions and different individuals form a linear space. This assumption is empirically justified, at least to some extent, by the success of PCA [Craw *et al.*, 1999].

2. PCA is known to be variant with respect to affine deformations. Consequently, the gallery has to be normalized such that the face features are co-located in a common coordinate system in order to be comparable. Still, it is a major drawback that eigenfaces not only encode inter-class variations that are useful for recognition, but also intra-class variations (e.g. expression), which include information that is unwanted for recognition. How to separate inter-class from intra-class information is not clear.

3. Variations between images are often due to illumination changes [Moses *et al.*, 1994]. The matrix W_{PCA} then contains eigenfaces that are due to lighting variations. A consequence is that the points in the projected space are not well clustered, or even worse, the classes are smeared. An often proposed method of reducing variations due to lighting is to discard the three most significant principal components [Moses *et al.*, 1994]. But the hope that these eigenvectors capture solely variations due to lighting is unlikely to be fulfilled; other important information, that is vital for discrimination, may be lost also.

As a consequence, PCA gives its best results when the gallery images and probe images are aligned in a one common coordinate system, when the images do not show facial expressions, and when the lighting is controlled. PCA results degrade severely when the images are not aligned, and it degrades moderately with expression and illumination variations [Phillips *et al.*, 1997; Phillips *et al.*, 1998].

5.2.2 Elastic Bunch Graph Matching

Elastic bunch graph matching is based on Gabor wavelets [Daugman, 1988; Wiskott *et al.*, 1997; Krüger *et al.*, 1996; Maurer and von der Malsburg, 1995]. The underlying idea is that a face (or, more generally, an object) is represented by a set of specific, meaningful feature points. Each of these feature points is described by a *jet* (see Section 2.7), which is a set of filter responses of 40 complex Gabor filters that are applied at that point. Thus a jet describes a local neighborhood of gray-values around a feature point. The filter set is fixed and usually (but this

may vary) contains Gabor filters that have eight different orientations and five different central frequencies.

For gallery images, relevant feature points can be selected by hand; for probe images, the corresponding feature points have to be found automatically. The highly redundant representation makes the search for these feature point inefficient [Wiskott *et al.*, 1997]. On the other hand, no normalization with respect to the facial features is needed, either for the encoding of the gallery or for the matching process. Furthermore, the use of Gabor functions ensures that the representation is somewhat invariant with respect to illumination changes. The similarity function [Wiskott *et al.*, 1997]

$$\mathcal{S}_a(\mathcal{J}, \mathcal{J}') = \frac{\sum_j a_j a'_j}{\sqrt{\sum_i a_j^2 \sum_j a'^2_j}} \tag{5.4}$$

is defined to be the normalized magnitude of the filter responses. In the above equation, $\mathcal{J}$ and $\mathcal{J}'$ refer to different jets, and a_j and a'_j to their magnitudes.

5.3 Representing Faces with GWNs for Automatic Recognition

In the following sections we will discuss various aspects of automatic face recognition with GWNs. First of all, we will explain in detail how the recognition should be done. Therefore, in this section we will give some details about the general procedure for automatic face recognition with GWNs. Then, in the following sections, we will systematically evaluate the properties and dependencies of GWNs with respect to

1. facial expression and other variations such as glasses,
2. illumination variations, and
3. affine deformations.

The idea underlying matching with GWNs is that a GWN $(\Psi, \mathbf{w})$ that is optimized for a particular person f appears to be very specific to that individual. A different image of the same person can be represented by a GWN $(\Psi, \mathbf{w}')$, in which the family of wavelets Ψ is the same, but the weight vector $\mathbf{w}'$ is recalculated. But for any other individual g, it appears that the GWN that was optimized for f is not a good representation. When trying to reconstruct g using the wavelet family Ψ, a new weight vector $\mathbf{w}''$ can be found, but the reconstruction according to eq. (2.24) is far from being acceptable. An example is shown in Fig. 5.1. The left image shows the reconstruction of the face image f on which the GWN $(\Psi, \mathbf{w})$ was optimized. The

center image shows the same individual with a different facial expression, represented with the wavelet family Ψ and the newly calculated weight vector $\mathbf{w}'$. The right image shows another face image g represented with the GWN $(\Psi, \mathbf{w}'')$. All weight vectors $\mathbf{w}'$ and $\mathbf{w}''$ were chosen optimally according to Section 2.4. These images show that for the new image g, no vector $\mathbf{w}''$

Figure 5.1. The left image shows the original face, represented with an optimized GWN. The center image shows the same person, but with a "smile" expression. The right image shows a different individual, represented with the same GWN used for the first two images. We see that the new individual cannot be represented well by a GWN that was optimized for the first individual.

can be found that gives a reconstruction as good as those shown for the original individual f (left image) or for the original individual with a different facial expression ("happy") (middle image). This shows that the wavelet family Ψ, since it is optimized for an individual f, is very specific to f. Therefore, when we say below that a GWN $(\Psi, \mathbf{w})$ is specific to a person, we mean that the wavelet family Ψ is very specific to that particular person, and we ignore the specificity of $\mathbf{w}$.

Since, as seen in Fig. 5.1, GWNs are very specific to the persons they are optimized on, it seems reasonable that to find out whether the probe image g shows the person in gallery image f, we can apply the GWN of the face in f to the face in g; the quality of the reconstruction will determine whether or not the two images show the same person.

In summary, our matching strategy consists of three steps:

1. Encoding each of the gallery images with a GWN,
2. encoding a probe image with each GWN in the gallery, and
3. successive comparison of the probe image with each of the gallery images.

5.3.1 Encoding of the Gallery

In the first step, the gallery images are encoded. Above, we introduced the term *gallery* in the strict sense, as a set of image sets each of which contains all possible face images of a specific individual. Let $\mathcal{G} = \{\mathcal{F}_i | i = 1 \dots n\}$. Each set $\mathcal{F}_i$ contains all possible images of individual $\mathcal{F}_i$. We would like to consider the mean face of $\mathcal{F}_i$ as a representative of that set. It has been found [Cootes *et al.*, 1998; Moghaddam and Pentland, 1997; Vetter and Blanz, 1998] that the mean face does not show any expression or illumination effects. However, since we cannot calculate the mean face, we simply take the image of person $\mathcal{F}_i$ that shows a "normal" facial expression, i.e. no expression, and normalize the lighting conditions. We therefore rewrite the above definition of $\mathcal{G}$ as containing the set of mean faces: $\mathcal{G} = \{f_1 \dots f_n\}$, where f_i now refers to the mean face of a specific person.

For each image $f_i \in \mathcal{G} = \{f_1, \dots, f_m\}$, a GWN$(\Psi, \mathbf{w})_{f_i} = (\Psi_i, \mathbf{w}_i)$ is optimized and stored. Each individual is thus represented by a specific GWN in the gallery $\mathcal{G}'$: $\mathcal{G}' = \{(\Psi_1, \mathbf{w}_1)$ $\dots (\Psi_m, \mathbf{w}_m)\}$. Each GWN $(\Psi_i, \mathbf{w}_i)$ is considered to be the representation of the individual f_i. The GWNs are optimized as explained in Chapter 2.

5.3.2 Encoding the Probe Image

In order to recognize the person in the probe image g, that image needs to be encoded using each GWN in the gallery. If there is a GWN that allows a good representation of the probe g, that GWN identifies the person in g. If there is no such GWN, the probe g is rejected as unknown.

In order to represent a probe g using a Gallery GWN $(\Psi_i, \mathbf{w}_i)$, the operators $\mathcal{P}$ and $\mathcal{T}$ are employed; both were introduced in Chapter 2. The operator $\mathcal{P}$ is used to reparameterize (warp) the given GWN $(\Psi, \mathbf{w})$ onto the face image in the probe g. The operator $\mathcal{T}$ is used to calculate the optimal set of weights $\mathbf{w}'$ for the particular family of wavelets Ψ.

First, the operator $\mathcal{P}$ is applied, in order to reparameterize (warp) the GWN $(\Psi_i, \mathbf{w}_i)$ to fit the face in probe p:

$$\mathbf{n} = \mathcal{P}_{\Psi_i}(g) \ , \tag{5.5}$$

where $\mathbf{n}$ is the new affine parameter vector of the Gabor superwavelet $\Psi_{i_\mathbf{n}}$. To fit the net to the unknown face, the GWN is deformed affinely. Other variations are not considered.

We have argued above that before recalculating the weights with the operator $\mathcal{T}$, a good reparameterization of the GWN is of vital importance. If the probe image contains the face of the individual f_i, we can be sure that the reparameterization with the network $(\Psi_i, \mathbf{w}_i)$ will

be successful. In our experiments, the optimal set of parameters was found in 100% of these cases. In situations where the images are different, it may happen occasionally that the operator $\mathcal{P}$ converges to the wrong minimum. However, in our experiments convergence was correct in 85% of the cases. The optimality of the reparameterization values was judged by examining images like those in Fig. 5.5, top row, which shows different individuals with superimposed marked positions of the first 16 wavelets of the reparameterized GWN.

Second, the operator $\mathcal{T}$ is applied, in order to calculate the new weight vector $\mathbf{w}'$ that is optimal (with respect to the reparameterized Gabor superwavelet $\Psi_{i_\mathbf{n}}$) for the probe image g:

$$\hat{g} = \mathcal{T}^{\mathbf{n}}_{\Psi_i}(g) \ , \tag{5.6}$$

where $\hat{g}$ denotes the reconstruction of g with respect to the reparameterized superwavelet $\Psi_\mathbf{n}$ of f_i and the optimal weights $\mathbf{w}'$.

In summary, the steps are:

1. optimal reparameterization of the GWN using the positioning operator $\mathcal{P}_\Psi$,
2. calculation of the optimal weights for the optimally reparameterized GWN by using the projection operator $\mathcal{T}_\Psi$.

This can be written concisely as

$$\hat{g} = \mathcal{T}_\Psi^{\mathcal{P}(g)}(g) \ . \tag{5.7}$$

The time required to evaluate operator $\mathcal{T}_\Psi^{\mathcal{P}(g)}$ is the sum of the evaluation times of the operators $\mathcal{P}$ and $\mathcal{T}$. The operator $\mathcal{P}$ was already shown to be usable for tracking, i.e. its evaluation time is less than one second. The operator $\mathcal{T}$ requires a single matrix multiplication, which lies in the range of milliseconds.

5.3.3 Comparing the Gallery Image with the Probe Image

Let $(\Psi, \mathbf{w})$ be the GWN of image f. The composite operator $\mathcal{T}_\Psi^\mathcal{P}$ of eq.(5.7) leads to an image $\hat{g}$ that is very similar to g iff g is well characterized by that GWN. This means that (5.7) is approximately the identity iff $g \approx f$ or $g = \mathbf{0}^*$ (images are assumed to be DC-free). Assuming, without loss of generality, that $g \neq \mathbf{0}$, we can write the following: If $(\Psi, \mathbf{w})$ is the GWN of image f, then

$$\mathcal{T}_\Psi^{\mathcal{P}(g)}(g) = \begin{cases} \hat{g} \approx g & \text{iff } \ g \approx f \\ \hat{g} \neq g & \text{iff } \ g \neq f \ . \end{cases} \ . \tag{5.8}$$

Using eq. (5.8) it is straightforward to define two *similarity measures*:

*This is the trivial case where all weights are zero.

1. *Euclidean distance*

$$d^2_\Psi(f,g) = \|\mathcal{T}_\Psi^{\mathcal{P}(g)}(f) - \mathcal{T}_\Psi^{\mathcal{P}(g)}(g)\|_2 \ . \tag{5.9}$$

2. *normalized cross correlation*

$$d^c_\Psi(f,g) = \frac{\overline{\mathcal{T}_\Psi^{\mathcal{P}(g)}(f) \cdot \mathcal{T}_\Psi^{\mathcal{P}(g)}(g)} - \overline{\mathcal{T}_\Psi^{\mathcal{P}(g)}(f)}\ \overline{\mathcal{T}_\Psi^{\mathcal{P}(g)}(g)}}{\sqrt{\mathrm{VAR}\big(\mathcal{T}_\Psi^{\mathcal{P}(g)}(f)\big)\ \mathrm{VAR}\big(\mathcal{T}_\Psi^{\mathcal{P}(g)}(g)\big)}} \ , \tag{5.10}$$

where $\cdot$ denotes the pixelwise product, $\bar{\cdot}$ denotes the mean and $\mathrm{VAR}(\cdot)$ the variance.

Let us take a closer look at the two distances. The distance measure $d^2_\Psi(f,g)$ is defined to be the sum of the pixelwise squared differences between the two images. The first image is the wavelet representation of the gallery image f that has been warped onto the probe image g (see Fig. 2.11), with the *original weight vector* $\mathbf{w}$ of gallery image f:

$$\hat{f} = \mathcal{T}_\Psi^{\mathcal{P}(g)}(f) \ . \tag{5.11}$$

The image f needs to be warped onto the image g so that the face features are aligned in a common coordinate system. This is important later for the pixelwise comparison.

The second image is the wavelet representation of the probe image g with respect to the GWN of the gallery image f and with the new weight vector $\mathbf{w}'$. This image is given by

$$\hat{g} = \mathcal{T}_\Psi^{\mathcal{P}(g)}(g) \ . \tag{5.12}$$

The distance $d^2_\Psi(\cdot,\cdot)$ is then defined as

$$\begin{aligned} d^2_\Psi(f,g) &= \|\hat{f} - \hat{g}\|_2 \\ &= \|\mathcal{T}_\Psi^{\mathcal{P}(g)}(f) - \mathcal{T}_\Psi^{\mathcal{P}(g)}(g)\|_2 \ . \end{aligned} \tag{5.13}$$

Clearly, the more similar the two images $\hat{f}$ and $\hat{g}$ are, the smaller is the distance d^2_Ψ.

In Fig. 5.2, examples are shown: the image $\hat{I} = \mathcal{T}_\Psi^{\mathcal{P}(J)}(I)$ of eq. (5.11) (left) and the image $\hat{J} = \mathcal{T}_\Psi^{\mathcal{P}(J)}(J)$ of eq. (5.12) (right). Ψ is the GWN optimized for image I. The distance d^2_Ψ is the sum of the squared differences between the two images. In the ideal case where the probe image J is the same as the gallery image I, the d^2 distance is zero. If the probe image J is derived from the same person as the gallery image I, in general the d^2 measure is small.

The distance measure $d^c_\Psi(f,g)$ is defined to be the *normalized cross correlation* between the two images $\hat{f}$ and $\hat{g}$, normalized with respect to the means and the variances of both images:

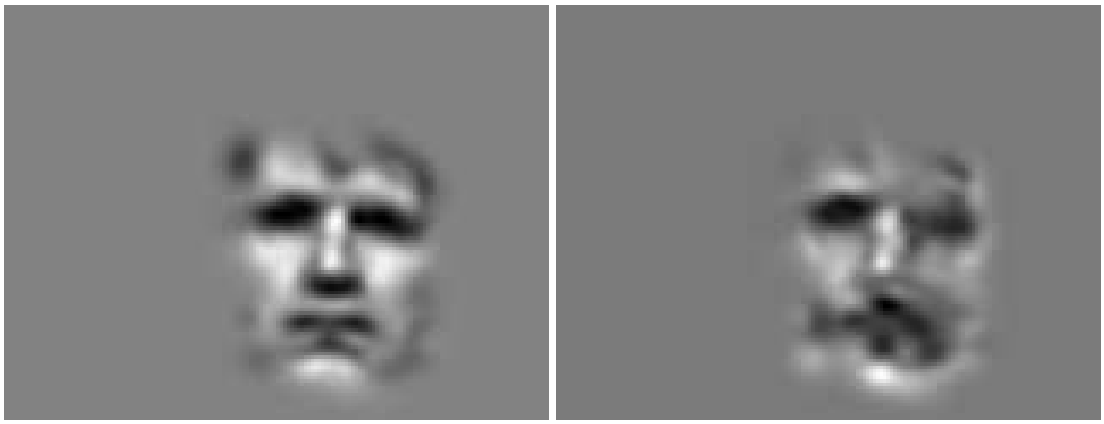


Figure 5.2. These two images show the original image I, warped onto the image J, $\mathcal{T}_\Psi^{\mathcal{P}(J)}(I)$ (left), and the result of the operator applying $\mathcal{T}_\Psi^{\mathcal{P}(J)}(J)$ to the image J. The distance $d^2_\Psi(I,J)$ between images I and J is defined to be the sum of squared differences between these two images and the distance $d^c_\Psi(I,J)$ is defined to be their normalized cross correlation. The GWN Ψ is optimized on image I.

The mean is discarded and the variance is normalized to 1. The normalized cross correlation has the property

$$-1 \leq d^c_\Psi(f,g) \leq 1$$

for all images f,g. The closer $|d^c_\Psi(f,g)|$ is to 1, the more similar the images are.

The two distance measures d^2 and d^c in the above equations calculate the differences based on the pixel values of the remap. This is inefficient, as the GWN results in a data reduction that allows us to represent each image by a small set of weights $\mathbf{w}$. It is possible to calculate the difference measures based solely on the weight vectors $\mathbf{v}$ and $\mathbf{w}$. As mentioned above, the operator $\mathcal{P}$ has the effect that the two images f and g are aligned. This means that both images can be represented using the *same* wavelet family, only the weights are different. Starting from the GWN $(\Psi, \mathbf{v})$ of image f, we end up with the two GWNs $(\Psi, \mathbf{v})$, and $(\Psi, \mathbf{w})$ that represent f and g; a new weight vector $\mathbf{w}$ is derived with the operator $\mathcal{T}$. This means that eq. (5.13) can be rewritten:

$$\begin{aligned} d^2_\Psi(f,g) &= \|\hat{f} - \hat{g}\|_2 \\ &= \|\sum_{i=1}^{N} v_i\psi_i - \sum_{j=1}^{N} w_j\psi_j\|_2 \ . \end{aligned} \tag{5.14}$$

Comparing this with $d_{\mathbf{\Phi}}$ in eq. (2.49), one sees that d^2_Ψ can be simplified for the specific wavelet family Ψ so that d^2_Ψ can be calculated directly with the weights $\mathbf{w}$ and $\mathbf{v}$, using d_Ψ:

$$d^2_\Psi(f,g) = (\mathbf{v}-\mathbf{w})^t \left(\Psi_{i,j}\right) (\mathbf{v}-\mathbf{w}) \,, \tag{5.15}$$

with $\left(\Psi_{i,j}\right) = \langle \psi_i, \psi_j \rangle$, as above. This allows us to calculate the distance d^2 solely from the weights, and it avoids the need for explicit reconstruction and a pixelwise comparison.

Like d^2 , the measure d^c can be calculated on the basis of the weights only:

$$d^c_\Psi(f,g) = \frac{\mathbf{v}^t \left(\Psi_{i,j}\right) \mathbf{w}}{\sqrt{\mathbf{v}^t \left(\Psi_{i,j}\right) \mathbf{v}}\sqrt{\mathbf{w}^t \left(\Psi_{i,j}\right) \mathbf{w}}} \tag{5.16}$$

As we will see in the next section, these two distance measures allow face recognition rates of up to 96%.

5.4 Recognizing Faces Independently of Expression Variations

In the previous section we presented a general procedure for recognizing faces with GWNs As mentioned there, each GWN is used as a representation of the set of all possible face images of a specific individual. The ability of the GWN to represent all the images in that particular set, in addition to itsinability to represent any image in another set, makes it feasible to use the GWN for unique representation of image sets, and so for recognition of faces. In this sense, the GWN is taken to represent an invariance property of the images within a specific set, such that this invariance is not a property of any other image set. If GWNs are taken to uniquely identify images of a certain person, we have to verify the invariance of the GWN with respect to (see also the beginning of Chapter 5):

- facial expression,
- illumination, and
- pose.

Pose variations can be compensated if the variation in appearance can be modeled by an affine deformation, as discussed above. Illumination variations will be discussed in the next section. In this section will verify the invariance of the GWNs with respect to facial expressions. The following section will give an overview of different approaches to image coding that have been successfully used for face recognition and will discuss their invariance properties with respect to facial expressions. We will conclude this section with experimental results.

5.4.1 Background and Related Work

Two general approaches exist for face expression invariant face recognition. We call them here explicit and implicit approaches.

Explicit approaches try to explicitly model e.g. facial expressions or illumination effects. In [Edwards *et al.*, 1998; Vetter and Blanz, 1998] the system tries to synthesize the probe face, and the synthesis parameters allow it to identify the individual as well as the expression. In [Edwards *et al.*, 1998] PCA and special eigenfaces are used to model the texture and geometry of faces. PCA is used to find valid instances of the synthesis parameters (eigenvectors) and to separate the parameters for identification and for the various expressions. Using this approach, [Edwards *et al.*, 1998] were able to synthesize almost any face and any expression. In [Vetter and Blanz, 1998] no such general representation is used. Instead, a separate model is used for each person from which every expression of that person can be synthesized. In contrast to this approach, which is a 2-D-approach, [Vetter and Blanz, 1998] use full 3-D information which is then back-projected onto 2-D. In both approaches, the sum-of-squared difference (SSD) is used as a criterion for the quality of the synthesis.

Implicit approaches ignore expression variations to some extent. For example, most common eigenface approaches [Moghaddam and Pentland, 1997; Craw *et al.*, 1999] ignore what [Edwards *et al.*, 1998] try to explicitly model. The resulting variations in the eigenvalue parameters are compensated using a statistical model of eigenvalue variations in parameter space. Since PCA is a "global" representation, this is relatively robust because local variations are canceled out. In [Wiskott *et al.*, 1997], expressions are automatically compensated by the variability of the jets. The graph itself is rather static and is only allowed to deform affinely.

Expression-invariant face recognition with GWNs can be considered to be an implicit approach. We assume that variations in face appearance caused by different facial expressions are only of a small scale. Large-scale information, which includes geometric properties and holistic face information, is assumed to remain mostly unchanged.

Many publications deal with facial expressions in general. However, only the few cited above attempt to recognize individuals. Many other approaches attempt to recognize the expression, independent of the individual. Here, the most commonly used approach is to track muscle actions over time (Facial Action Coding System (FACS)) [Ishikawa *et al.*, 1998; Hong *et al.*, 1998; Lien *et al.*, 1998].

In most experiments, standard facial expressions are usually considered, including

- *normal*: a normal facial expression, i.e. no particular expression

- *happy*: a happy facial expression, ranging from a smile (closed mouth) to a laugh, (open mouth). Most variations are in the mouth region; minor variations are found around the eyes
- *sad*: a sad facial expression. Again, most variations are around the mouth, and there are minor changes around the eyes.
- *surprised*: a surprised facial expression. There are major changes around the mouth and eyes; the eyes are wide open and the eyebrows are lifted.
- *sleepy*: a sleepy facial expression. The eyes are shut, i.e. there are minor local changes in the eye region; the face looks very much like the normal expression.
- *wink*: a wink facial expression. One eye is closed, the other open. Depending on how easily the individual is able to wink, there are more or less strong local variations around the closed eye.

Examples of the various facial expression are shown in Fig. 5.3. These images are derived from the Yale Face Database.

Figure 5.3. The Yale database contains images showing six different facial expressions of each individual in the database: *normal, happy, sad, surprised, sleepy* **and** *wink.*

Facial expressions like *surprised, happy* and *sad* show very strong variations in face appearance. For face recognition approaches, which deal with expressions implicitly, these expressions are difficult to compensate. Approaches that are able to synthesize expressions seem to have fewer recognition problems. However, no precise experimental results have yet been published. For expression recognition, on the other hand, these expressions are clearly the easiest to identify.

5.4.2 Experiments

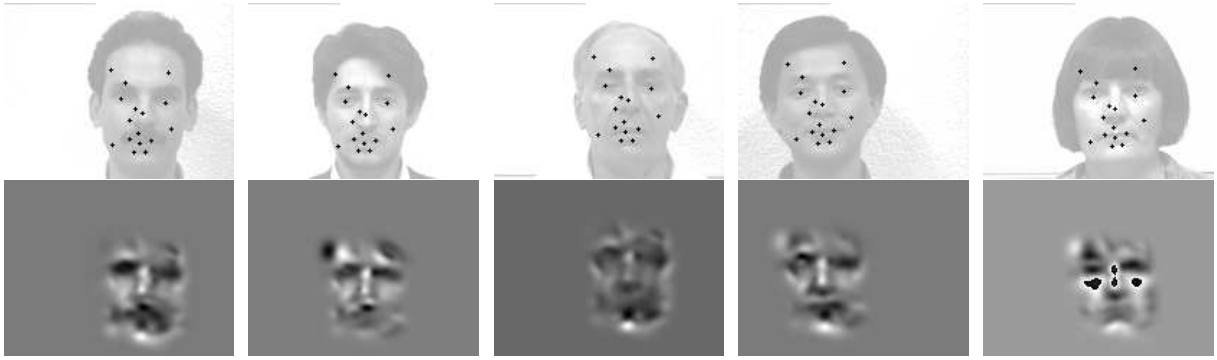
Experiments were carried out on the images in the Yale Face Database. The database consists of 15 different subjects whose faces show the six different expressions *normal, happy,*

sad, *surprised*, *sleepy* and *wink*, and also contains images showing the subject with and without glasses. The Manchester Database contains various different expressions, but they were not systematically organized. It was our goal to recognize each subject independently of the facial expression or the eye wear. To achieve this we optimized a GWN for each individual, where the optimization was done on the image showing the normal expression. Here the normal expression is considered to be a mean expression. For more details about the optimization procedure see Section 5.3.1.

Optimizing a GWN for each individual leads to a gallery of 15 GWNs, one for each individual, and 15 operators (see eq. (5.7)) $\mathcal{T}_{\Psi_i}^{\mathcal{P}(\cdot)}(\cdot)$, one for each individual's normal image I_i and GWN $(\Psi_i.\mathbf{w}_i)$, $i = 1, \ldots, 15$. In Fig. 5.4 and 5.5, example results of applying the operator $\mathcal{T}_{\Psi_1}^{\mathcal{P}(J)}(J)$ to the different facial expressions of individual I_1 (bottom rows), and to various other individuals I_j, $j \neq i$ (top rows), are shown. The GWN used here is the one optimized on the image with the *normal* expression of "subject01", which is shown in Fig. 5.4, top left. The optimized GWN is shown in the left bottom image in Fig. 5.4.

Figure 5.4. Various images of "subject01" (top) and the results of applying the operator $\mathcal{T}_{\Psi}^{\mathcal{P}(J)}(J)$. (bottom). To calculate these examples, the GWN of Fig. 2.4, $\hat{I}_{52}$, with 52 wavelets, was applied. The "normal" image was taken to be the image I on which the GWN Ψ was optimized.

The images in the top row of Fig. 5.5 show the superimposed positions of the wavelets in the GWN after the reparameterization of the GWN of "subject01". By looking at the examples in Figs. 5.4 and 5.5, we can intuitively compare the results of applying operator $\mathcal{T}_{\Psi_1}^{\mathcal{P}(J)}(J)$ to different probe images J, with the optimal results when it is applied to the gallery image, i.e. the image that the GWN was optimized on (bottom left, Fig. 5.4). This is what is done by the



subject02 subject03 subject05 subject06 subject11

Figure 5.5. Various subjects in the database (top) and the results of applying the operator $\mathcal{T}_\Psi^{\mathcal{P}(J)}(J)$(bottom). To calculate these examples, the GWN of Fig. 2.4, $\hat{I}_{52}$ with 52 wavelets, was applied. The "normal" image in Fig. 5.4 was taken to be the image I on which the GWN Ψ was optimized.

two distance measures

$$\begin{aligned} d^2_\Psi(I,J) &= \|\mathcal{T}_\Psi^{\mathcal{P}(J)}(I) - \mathcal{T}_\Psi^{\mathcal{P}(J)}(J)\|_2 \text{ and} \\ d^c_\Psi(I,J) &= \frac{\overline{\mathcal{T}_\Psi^{\mathcal{P}(J)}(I)\cdot\mathcal{T}_\Psi^{\mathcal{P}(J)}(J)} - \overline{\mathcal{T}_\Psi^{\mathcal{P}(J)}(I)}\;\overline{\mathcal{T}_\Psi^{\mathcal{P}(J)}(J)}}{\sqrt{\mathrm{VAR}\big(\mathcal{T}_\Psi^{\mathcal{P}(J)}(I)\big)\;\mathrm{VAR}\big(\mathcal{T}_\Psi^{\mathcal{P}(J)}(J)\big)}} \end{aligned} \tag{5.17}$$

that were introduced in Section 5.3.3. In eq. (5.17) the wavelet family Ψ is understood to have been optimized on image I. Image J is assumed here to be the probe image.

5.4.3 Experimental Results

Tables 5.6 and 5.7 show the experimental results, where the similarity was computed between subject01 with a normal expression and

- images of subject01 with different facial expressions,
- images of subject02 – subject15, all showing normal expressions.

The similarity measurements used were d^2 and d^c. The two tables can be generalized to the other subjects. A clear difference can be seen for d^c between the probe images of the original subject and the probe images of other subjects. The difference in d^2 seems to be less drastic (note that the scalings of the axes of Tables 5.6 and 5.7 are different), but still confirms our expectation. We obtained a recognition rate of

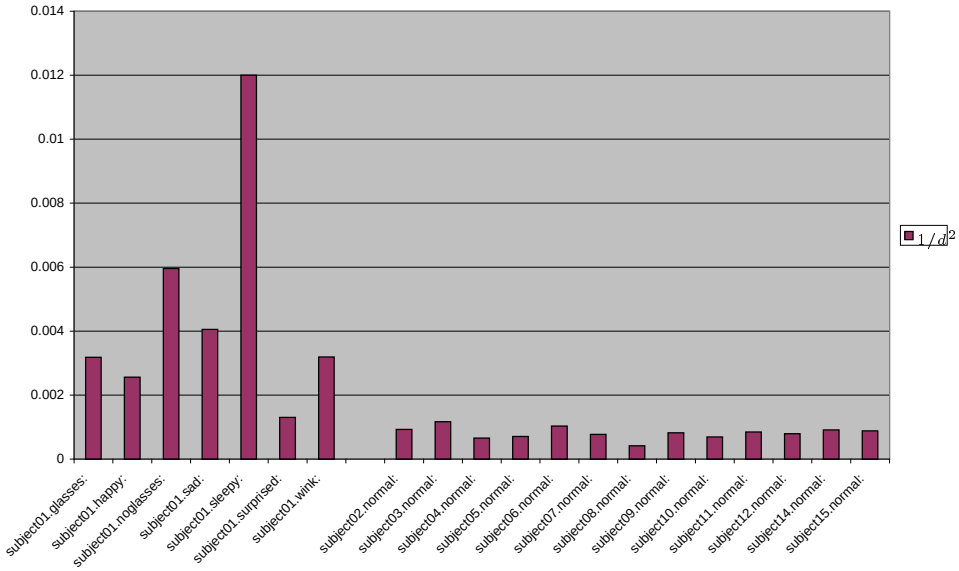


Figure 5.6. This table shows the similarity measurements $1/d^2$ of the images of the various subjects in the face database to the reference image in Fig. 5.4, left. Higher values indicate higher similarity between the images. One sees that the values in the left part of the table (same subject) indicate much higher similarity than the values in the right part of the table (different subjects).

- 96% with the d^c measure
- 94.7% with the d^2 measure.

Using the distance measure introduced by [Wiskott *et al.*, 1997] (see eq. (5.4)), the recognition results degraded to 89.3%.

Recognizing the "surprised" expression failed on five individuals. Leaving out the "surprised" expression, the recognition rates increased to

- 97.6% with the d^c measure
- 96.9% with the d^2 measure.

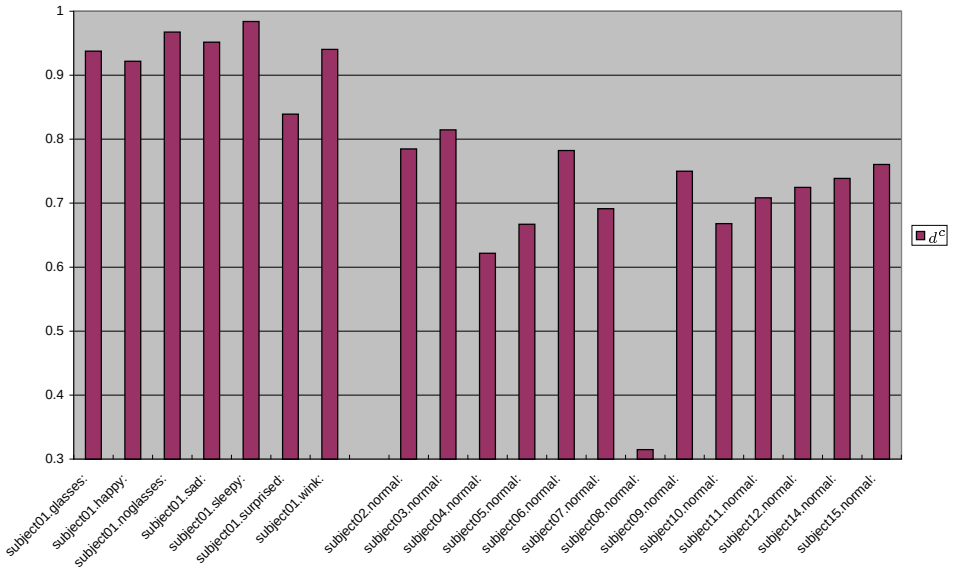


Figure 5.7. This table shows the correlation measurements d^c of the images of the various subjects in the face database to the reference image in Fig. 5.4, left. Higher values indicate a higher similarity between the images. One sees that the values in the left part of the table (same subject) indicate much higher similarity than the values in the right part of the table (different subjects).

5.4.4 Analysis of and Comments on the Experimental Results

A GWN that is optimized on a normal facial expression image of a specific individual represents a collection of local facial features. Because of the sparseness of this representation, different individuals cannot be represented well with a single GWN. A GWN encodes the overall geometry of the face that the network is optimized on, and it is assumed that this overall geometry remains in principle the same in different images.

The optimization of a GWN on an image f ensures that the positions, sizes and orientations of each of the image features of f are encoded very precisely. (We mentioned this in Section 3.1). Each Gabor wavelet encodes a specific feature, and the entire family of Gabor wavelets encodes the overall layout of the image features. Since the local image features and the global layout are very specific to the image on which the net was optimized, the "fit" of the GWN to

other face images is very bad, because the features are located differently and the face has a different overall layout. Assuming, however, that the overall geometry of a specific person's face stays the same in all images, and that different expressions result only in minor local changes, the GWN allows good reconstruction of all images of that individual's face, independent of his/her expression. In fact, it is reasonable to ask to what degree facial expressions can be compensated. It seems that most variations in facial appearance under varying expressions occur locally and at large scales. The GWNs used in our experiments contain mainly wavelets of these large scales, as can be observed by comparing the original image and its remap (see Fig. 2.4). Furthermore, the GWN representation is not a strictly local representation. Rather it cancels out local variations by considering entire neighborhoods of pixels that are located within the support of each filter. This is also the reason why images in which the probe person wears glasses are all well recognized.

On the other hand, the fit considerably degrades when a GWN is warped onto other individuals. This clear difference in "fit" can be seen in Fig. 5.1, where the ability to represent a different individual clearly degrades.

The assumption that only minor local changes occur under variations of facial expression is violated for the "surprised" expression. This can be seen in the "surprised" image of each person. The result is that the ability of a GWN to represent an individual with the "surprised" expression degrades considerably, as can be seen from the similarity measurements in Tables 5.6 and 5.7.

Variations in the overall geometry of a face that are due to affine deformations are compensated automatically during the positioning process under the operator $\mathcal{P}$.

5.5 Recognizing Faces Independently of Illumination Changes

In the previous section we discussed the invariance properties of the GWN representation with respect to facial expressions. In this section we will examine the invariance properties of GWNs with respect to illumination changes. When faces are illuminated by different light sources, or from different directions, the faces can appear dramatically different. Stable recognition in spite of severe lighting variations is still an open problem, and also depends on research areas like shape from shading and photometric stereo [Hayakawa, 1994; Horn, 1986; Belhumeur *et al.*, 1997].

As in the previous section, different approaches exist that make it possible to cope with illumination variations: *Explicit approaches* try to model the illumination situation [Belhumeur *et al.*, 1997; Sung and Poggio, 1994; Shashua, 1992; Nayar and Murase, 1996; A.S.Georghiades

et al., 1999]. They are theoretically well-founded; however, they are unable to model *every* lighting condition with every possible number of light sources. *Implicit approaches* ignore illumination variations. No model is needed in this case and only minor assumptions have to be made. This section will give an overview of different approaches that have been successfully used for face recognition and will discuss their invariance properties with respect to illumination variations. We will conclude this section with experimental results.

5.5.1 Background and Related Work

Eigenface methods are expected to suffer severely under lighting variations because illumination variations cannot be modeled by the set of eigenfaces [Belhumeur *et al.*, 1997; Craw *et al.*, 1999]. Furthermore, it is well known that the training of the eigenfaces on a set of gallery images suffers from variable illumination during image capture [Belhumeur *et al.*, 1997; Craw *et al.*, 1999]. When eigenfaces are trained on the basis of images that were captured under varying illumination, the eigenfaces will retain these variations. As a consequence, points in the projected space will not be well clustered; instead, classes will be smeared out. This phenomenon was extensively discussed in [Moses *et al.*, 1994]. In order to cope with illumination variations during training of the eigenfaces, [Moses *et al.*, 1994] proposed discarding the three most significant eigenfaces, as they appear to contain most of the illumination variations. The hope was that by discarding the most significant eigenfaces, the clustering in feature space would be better. However, it is unlikely, that the three most significant eigenvectors solely capture lighting variations. More likely, important information for discrimination and classification will also be lost.

In order to cope with illumination variations in some other way, various suggestions have been made. In [Sung and Poggio, 1994], for example, a best-fit brightness plane is first subtracted in order to reduce the strength of heavy shadows caused by extreme lighting angles. A best-fit brightness plane provides a linear approximation to the gray value variation in the image. This is especially useful in situations where the illumination is not frontal. The subtraction of the best-fit brightness plane is followed by histogram equalization.

In [Shashua, 1992; Nayar and Murase, 1996], a linear subspace method is proposed that should allow recognition under arbitrary lighting conditions. The linear subspace method exploits a well-known method from photometric stereo. It can be observed that images of a Lambertian surface without shadows lie in a 3-D linear subspace. More precisely, let p be a point on a Lambertian surface that is illuminated by an infinitesimal light source at infinity. Furthermore, let $\mathbf{s} \in \mathbb{R}^3$ denote the product of the light source intensity with the light source

orientation. Then the resulting intensity at point p, as viewed by the camera, is given by

$$E(p) = a(p)\mathbf{n}(p)^T\mathbf{s} \ .$$

Here $\mathbf{n}(s)$ is the inward normal vector to the surface at the point p, and $a(p)$ is the albedo of the surface at p. This means that the albedo and the surface normal can be recovered, given three images of a Lambertian surface from the same viewing direction but illuminated from three linearly independent light source directions. In other words, the image of a surface that is illuminated from an arbitrary direction can be recovered by a linear combination of these three images [Shashua, 1992]. Assuming that faces are Lambertian and that the positions of the light sources in the original images are precisely known, then in principle, the linear subspace method allows us to cope with arbitrary illumination situations with a single light source. However, self-shadowing, specularities, and facial expressions cannot be handled by the linear subspace method. A further drawback is that the linear subspace method needs to store at least three images for each person (same viewing direction, three linearly independent illumination directions) in order to approximate the linear subspace. Also, this method can easily deal with situations where a face is illuminated by only a single light source, but in situations where it is illuminated by a set of light sources, these approaches fail. In fact, the impossibility of finding an illumination model that allows modeling of any situation is the major drawback of the explicit approach.

An alternative to modeling illumination explicitly with best-fit brightness planes or linear subspace methods is to use an implicit approach that filters or "ignores" the illumination variations. As an example, [Zeng and Sommer, 1996] discusses the effects of illumination variations in the frequency domain. Homomorphic filtering is used as a preprocessing step to a PCA-based recognition approach in order to improve recognition rates. As a further example, the bunch graph approach [Wiskott *et al.*, 1997] employs Gabor wavelets for the representation of local gray-value variations. Gabor wavelet functions, as used in [Wiskott *et al.*, 1997], are known to have a vanishing DC component. This means that illumination variations are not perceived if they are homogeneous within the support of the filter. The term "homogeneous variation" means that the illumination may vary with a constant offset c for all pixels within the support of each Gabor filter. This results in a change in the mean value (the mean calculated within the support of the filter) by the offset c while the filter response of the DC-free filter stays the same. In other words, the bunch graph approach, which discards the mean values and relies only on the filter responses, can be viewed as being robust to homogeneous illumination variations.

The support of the Gabor filters is relatively small, so that illumination homogeneity within

the filter support is a relatively weak assumption. This means that specularities and minor self-shadowing can well be ignored by the bunch graph representation as long as the supports of the affected Gabor filters are small relative to the affected image area.

The filter responses depend on the local contrast in the image. This contrast may change under illumination variations. The similarity function of [Wiskott *et al.*, 1997], which is given as the normalized cross correlation between two jets, normalizes local gray-value variance (contrast):

$$\mathcal{S}_c(\mathcal{J},\mathcal{J}') = \frac{\sum_j a_j {a'}_j}{\sqrt{\sum_i a_j^2 \sum_j {a'}_j^2}} ,$$

where the a_i refer to the responses of the Gabor wavelets. "Contrast variation" is understood here as a constant factor c applied to the image. Again, because the filter support is local, the factor c has to be constant only within the support of each jet.

GWNs also use DC-free Gabor wavelet functions, and therefore have basically the same properties with respect to illumination changes as the bunch graph approach. Furthermore, the distance measure $d^c(\cdot,\cdot)$, which is based on the normalized cross correlation between the weights of the gallery image f and the weights of the new probe image g, normalizes the gray-value variance (contrast) (see 5.16):

$$d^c_{\Psi}(f,g) = \frac{\mathbf{v}^t \left(\Psi_{i,j}\right) \mathbf{w}}{\sqrt{\mathbf{v}^t \left(\Psi_{i,j}\right) \mathbf{v}}\sqrt{\mathbf{w}^t \left(\Psi_{i,j}\right) \mathbf{w}}} . \tag{5.18}$$

We will see in the next subsection that face recognition is indeed robust to illumination variations, as expected.

5.5.2 Experiments

In this section we present the results of experiments on the invariance of the GWN object representation with respect to illumination changes.

No image database is available that would allow systematic evaluation. Therefore, in this section we will use synthesized images. The images we used for testing are derived from the Yale Database. They are the images showing the normal facial expression. To these images, a brightness plane with variable orientation was added [Sung and Poggio, 1994]. The brightness plane is defined as

$$h_{A,B}(x,y) = Ax + By . \tag{5.19}$$

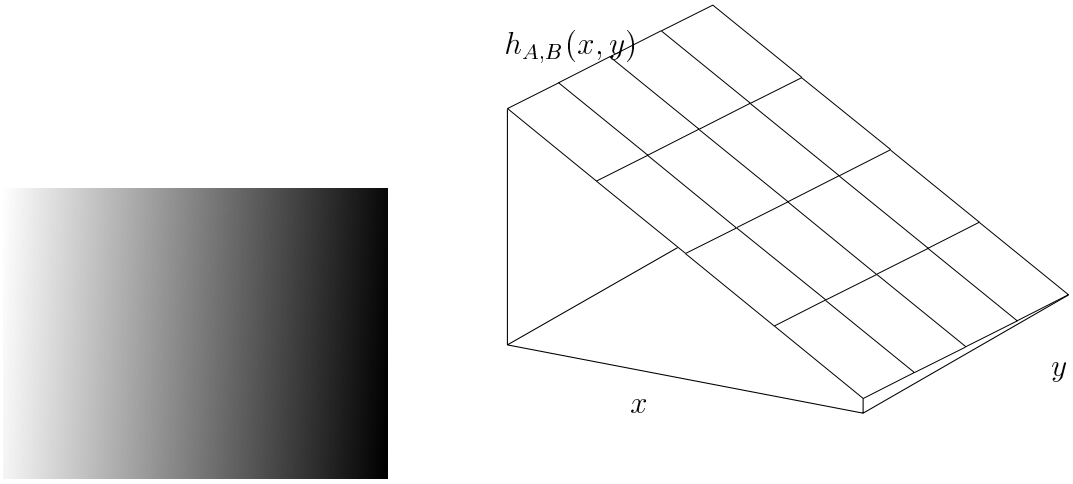
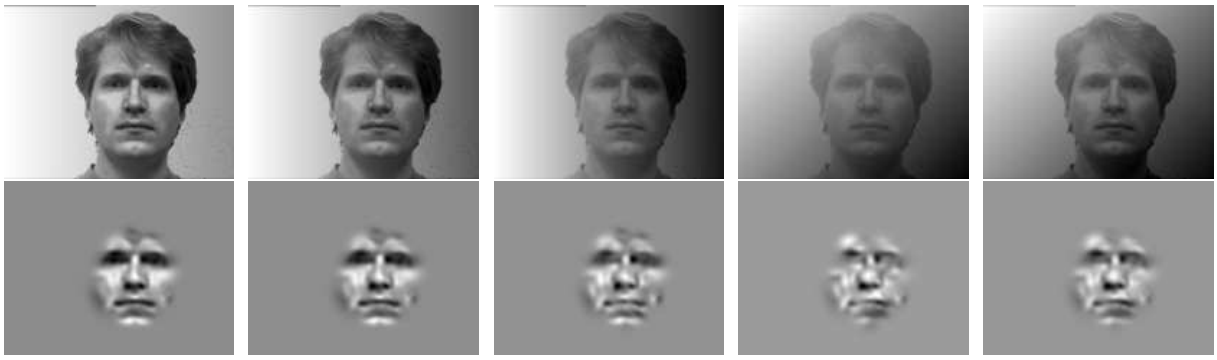


Figure 5.8. These two images show a gray-value surface for $A = 3$ and $B = 0.15$.

An example of such a plane is shown in Fig. 5.8. The brightness plane is added to the facial image to synthesize various illumination conditions. Example images can be seen in the top row of Fig. 5.9. We consider only "global" illumination changes, which can be closely approximated by such brightness planes. Specularities and self-shadowing are not considered here. $A = B = 0$ refers to frontal illumination, as the surface normal is parallel to the facial normal and orthogonal to the image plane. The illumination direction, i.e. the orientation of the surface normal, can be calculated directly: $90 - \arctan(-1/A)$ gives the angle between the normal and the x-axis, and $90 - \arctan(-1/B)$ gives the angle between the normal and the y-axis. For example, for $A = 1$ and $B = 0$, the face is illuminated from $45°$ from the left; for $A = 0$ and $B = 1$, the face is illuminated from $45°$ from the top. Examples are shown in Fig. 5.9. The notations below the images refer to the parameters in eq. (5.19). It can be seen in the images and in Tables 5.10 and 5.11 that for smaller angles with $A, B < 3$, the representation is only marginally affected by illumination variations. For larger angles, distortions increase.

5.5.3 Experimental Results

The visual impression of Fig. 5.9 is confirmed in Tables 5.10 and 5.11. In the first table the inverted Euclidean distance measure $1/d^2$ is used. The similarity degrades quickly as the illumination angle becomes less orthogonal to the face. The measures should also be compared to the ones in Table 5.6; measures above 0.002 can be considered as "recognized". In the second table the normalized correlation d^c is used as the measure. Here the decrease in similarity is less



A=1; B=0 A=2; B=0 A=5; B=0 A=5; B=5 A=3; B=3

Figure 5.9. Various images of "subject01" (top) and the results of applying the operator $\mathcal{T}_{\Psi}^{\mathcal{P}(f)}(f)$ (bottom). To calculate these examples, the GWN $(\Psi, \mathbf{v})$ of Fig. 2.4, $\hat{I}_{4,6}$, with 52 wavelets was applied. The notations below the images refer to the parameters of the added gray-level surface according to eq. (5.19). They correspond to an illumination of $\arctan(-1/A)^\circ$ from the left and $\arctan(-1/B)^\circ$ from the top. It can be seen that the illumination variations are well compensated for smaller angles ($A, B < 3$). For larger angles, the reconstruction quality degrades. This can also be seen in Tables. 5.10 and 5.11.

drastic, due to the normalization of contrast. By again comparing the results with Table 5.7, we see that measures above 0.85 can be considered as "recognized".

The tables can be generalized to the other subjects. Most illumination variations, e.g. $A = 0 \ \ldots \ 4$, with $B = 0$ for measurement d^c, and $A = 0 \ \ldots \ 3$, with $B = 0$ for d^2, are compensated, and recognition results on the subjects with "normal" expressions approach 100%. For stronger illumination variations, the recognition rates degrade rapidly.

5.5.4 Analysis and Comments on the Experimental Results

The experiments in this section have confirmed our expectations: Global illumination changes are well compensated by the Gabor wavelet representation because DC-free Gabor functions are used. The reason is that the filter response of each Gabor wavelet function is invariant to homogeneous illumination changes that occur within its support. Therefore, if the illumination change in a test image can be assumed to be locally homogeneous, the global illumination change can be completely compensated. Clearly, the degree of global illumination change depends on the support of each of the Gabor filters. Large-scale Gabor filters with large supports will allow smaller global variations than small-scale Gabor functions.

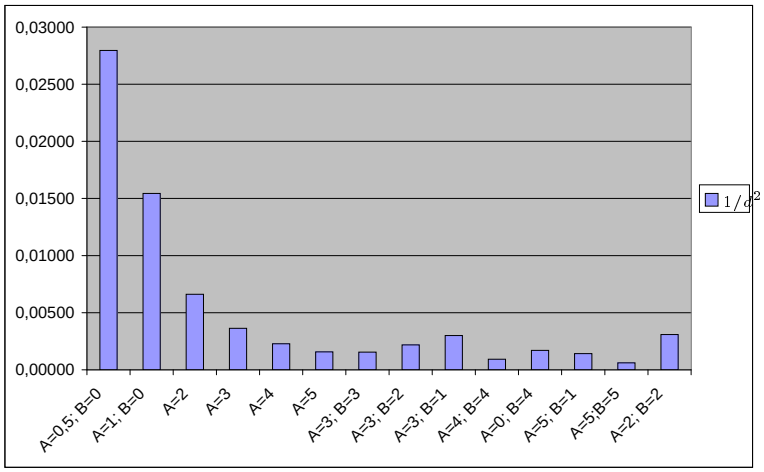


Figure 5.10. This table shows the similarity measurement $1/d^2$ for the images of subject01 under illumination variations applied to the reference image in Fig. 5.4, left. Higher values indicate higher similarity between the images. We see that the similarities vary with the illumination angle. For near-frontal illumination the similarities are best. These results should be compared to the results in Tables 5.6 and 5.7.

It was stated in the discussion of the previous section that the GWN that we used for our experiments contained mostly large-scale wavelets. Since the compensation of illumination variations assumes homogeneous illumination within the support of each filter, large-scale wavelets have a negative effect because the homogeneity assumption is violated more easily in large regions than in small regions.

5.6 Discussion

The major importance of this chapter was to show how image information is represented by GWNs. This was investigated in the context of face recognition experiments. Such experiments require precise representation of individuals, but they also require generality for independence of expression. Our approach is strictly appearance-based, where the identity of a person's face is judged by its appearance in a probe image. We have not used any geometrical model information about faces or their possible expressions, in order to ensure that the representation be as general as possible. In order to have the possibility of generalizing from faces to general

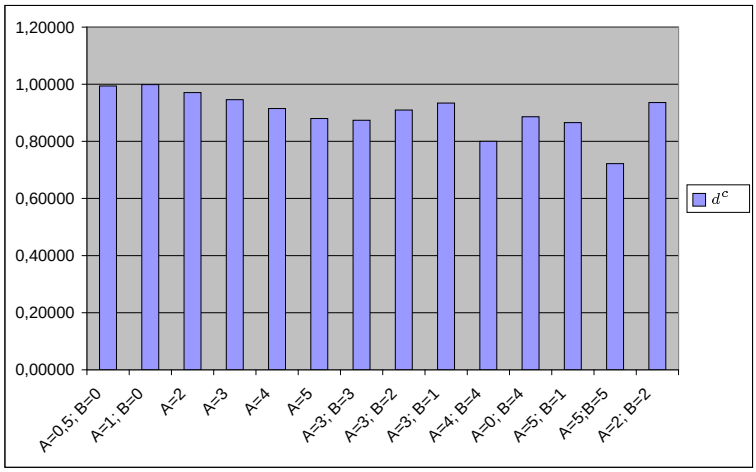


Figure 5.11. This table shows the distance measurements d^c for the images of subject01 under illumination variations applied to the reference image in Fig. 5.4, left. Higher values indicate higher similarity between the images. We see that the similarities vary with the illumination angle. For near-frontal illumination the similarities are best. These results should be compared to the results in Tables 5.6 and 5.7.

objects, this is very important.

Our experiments have shown that GWNs encode image information not only by means of the weights w_i. In fact, even more information is encoded in the parameter vector $\mathbf{n}_i$ of each of the optimized Gabor wavelets. The parameter vectors, which encode the orientations, positions and scales of the wavelets, are very important, as they ensure that the GWN is able to model the structure of a specific face. Just as this structure differs for different individuals, so do the optimized parameter vectors $\mathbf{n}_i$, and with them, the different optimized GWNs.

Affine deformations of a face are compensated by the ability of the GWN to adapt to such deformations. However, arbitrary deformations that are not affine cannot be modeled, and the GWN representation fails. This situation occurs, for example, when a GWN is used to represent a face on which it has not been optimized, because the relative positions of the facial features are completely different from those in the original face.

This situation also occurs, in a more moderate manner, when the facial expression of a face changes. However, when this happens, the relative positions of the facial features are not likely

to change much, so representation with the same GWN is still successful. A counter-example to this is the "surprised" expression, where the eyebrows are raised, the eyes are wide open, and the mouth is deformed so that the representation fails.

A Gabor wavelet has to be positioned precisely on the image feature it is supposed to represent. If it is not positioned there precisely, how large the displacement can become before it becomes visible in the image depends on the scale of the wavelets. For large wavelets the displacement can be much larger than for small wavelets. In all our experiments we have used rather large-scale wavelets; this can be seen by examining the re-mapped images. If we had used small-scale wavelets, the GWN would have been less robust with respect to changes in expression.

In all our experiments we used GWNs with $N = 52$ wavelets each. Changing the number of wavelets caused the recognition rate to decrease. We have found that a decrease in N reduces the precision, and in some sense the information content, of the representation, and the GWNs became less descriptive. An increase in N, on the other hand, leads to the networks becoming able to represent more than just the structure of the face they were optimized on so that the GWNs become less distinctive. A more precise evaluation of this observation would be interesting, but is beyond the scope of this thesis.

In Section 5.5 it was argued that the robustness of the GWN with respect to illumination changes increases as the scale of the wavelets decreases. On the other hand, with small-scale wavelets, robustness with respect to facial expression decreases. The best choice of scale therefore depends on the situation and the task.

This chapter has also shown how GWNs can be used for the automatic recognition of faces. A recognition rate of 96% was achieved. The recognition approach presented here should be regarded as a rudimentary system that could well be enhanced to achieve higher recognition rates. However, it should be admitted that this system has a principal drawback. For each gallery face a GWN has to be optimized and stored, and during recognition a probe face has to be processed by each GWN in the gallery. This requires much more computation time than the other recognition approaches that were mentioned earlier. This problem can possibly be solved by applying the progressive attention scheme: Start the search for the correct individual with a small set of wavelets, and increase the number of wavelets until a unique person is found. It should also be mentioned that only small databases were used in our experiments, and generalization of the above results to larger databases would not be easy. Experiments on this and on enhancement of the system will be left for future research.

Chapter 6

Using Gabor Wavelet Networks for Pose Estimation

In Chapter 3 we explained that filtering of a function with the wavelets of a GWN, and reconstructing a function by a weighted superposition of the wavelets, are closely related. Indeed, we have shown that the weights are linearly related to the filter responses of the wavelets by eq. (2.36). In other words, the responses of the filters of GWNs already contain all the information that is needed for reconstruction.

In the previous chapters we have explained that GWNs allow great data reduction and efficiency in object representation. The relation between filter responses and weights also allows us to exploit these advantages for image filtering. In other words, if we use a GWN with, e.g., $N = 52$ wavelets, their 52 filter responses suffice to represent almost the entire facial image. The image need not be filtered (convolved) with each of the 52 filters. Instead, the filter response refers only to the application of the filter at a single position which is determined by the parameters of each Gabor function and the superwavelet.

In this chapter we will investigate this property and show how GWNs can be used to define optimized and efficient filtering schemes that are able to extract much more information from images than, e.g., filtering schemes in which the Gabor wavelets are homogeneously distributed.

The application we will use in our investigations is pose estimation. In order to understand how the pose estimation will work and in what context the image filtering must be carried out, we will begin with a short introduction to general concepts, techniques and approaches.

The detection of the head pose and gaze detection of a human will be a major feature of future *human-computer interaction* (HCI) systems [Colombo and Bimbo, 1997; Daugman, 1997; Gavrila, 1999; Pavlovic *et al.*, 1997]. Various kinds of cooperative gaze detection systems exist,

but they are cumbersome and require hardware to be connected to the head of the user. In this chapter we are interested in non-cooperative gaze detection systems that leave the user free from wearing any hardware. Non-cooperative gaze detection, together with speech recognition, will allow very natural interactions with computer systems [Tock and Craw, 1996; Maggioni, 1995]. Today's speech recognition systems are already quite successful, but gaze detection systems are still under development, and it is not yet clear how the goal of reliable, precise, and fast gaze detection (real-time response of the system is a major requirement) can be achieved.

Most experimental systems for pose or gaze detection are monocular systems. For such systems, two major approaches exist: The first is a two-stage approach: Based on camera images of the user's face, the 3-D pose of the head is computed. In a second step, the eyes are examined to compute the orientation of the eyeballs relative to the pose of the head. The head pose, together with the relative orientation of the pupils, allows computation of the gaze direction. The second approach is a direct approach that allows direct detection of the user's gaze: An image of the user's face is processed as a whole, including all the facial features such as eyes and mouth, in order to estimate the gaze directly. The intermediate step of head pose computation is omitted.

It is clear that both steps of the two-stage approach must be carried out as precisely as possible. The computation of the head pose needs to be especially precise so that the localization of the iris and the computation of its position relative to the head is simplified.

In the experiments presented in this chapter we will concentrate on the first step of the two-stage approach which estimates the pose of the user's head. The approach that we will present is appearance-based. The input images are filtered using an optimized filtering scheme given by a GWN. The filter responses are then fed into an appropriately trained ANN which computes the 3-D head pose.

In the next section we will present an introduction to important terms and techniques. In the following section, we will give an introduction to related work, describe typical approaches, and give the necessary background. In Section 6.3 we will describe our experiments. In these experiments we will use GWNs for optimized filtering. We will also introduce a progressive attention scheme in this context, and will show how the computation speed and the quality of the pose estimation results can be controlled. We will also present results on the quality, robustness and efficiency of GWNs for 3-D head pose estimation.

6.1 Foundations

In this section we will present an introduction to important terms and techniques related to the estimation of pose and gaze.

By the term 3-D *pose* of a head we mean the orientation of the head's coordinate system relative to the camera-centered world coordinate system [Faugeras, 1993; Haralick and Shapiro, 1992]. For the projection process from the camera-centered world coordinate system onto the camera image plane, different projection models can be assumed, such as *perspective projection*, *weak perspective projection*, and *orthographic projection* [Faugeras, 1993].

Usually weak perspective projection is assumed; this is valid when the face appears approximately flat to the camera, i.e. when the face is coplanar with the camera-centered world coordinate system and when the depth changes on the face are small compared with the distance between the face and the camera and with the focal length. Weak perspective projection results in significant perspective distortion when the face is viewed from a close range with a short focal length lens.

By the *gaze direction* of a user we understand the direction in which the user is looking.

The eye corners and mouth corners of a face define a plane (implying that these four points are approximately co-planar) which we call the *face plane*. The pose of the face plane in 3-D space is given by its normal, the so-called *facial normal*. This normal can be uniquely determined from two angles:

- the *slant* σ, the angle between the optical axis (z-axis) and the facial normal in 3-D space,
- the *tilt* τ, the angle between the image normal and the x-axis.

In camera-centered coordinates, the facial normal $\hat{\mathbf{n}}$ is given by

$$\hat{\mathbf{n}} = \left(\sin\sigma\cos\tau, \sin\sigma\sin\tau, -\cos\sigma\right)^T \quad . \tag{6.1}$$

Two principal approaches exist to computing the 3-D pose of a head: the *model-based approach* and the *appearance-based approach*. By a *model-based approach* we mean a top-down approach, in which the 3-D pose is determined using an *a-priori* given 3-D geometric model of the head. This model is usually built from facial features, and further information is provided by their relative positions. Common landmark features are, e.g., the eye corners, mouth corners, nose tip and nostrils. These model features are matched against the camera image to find their projected positions. The model is then used to calculate the relationship between the 3-D model, which is aligned with the world coordinate system, and the head coordinate system, which is aligned with the facial features in the image. Model-based approaches usually assume a calibrated camera system.

By an *appearance-based approach* we mean a bottom-up approach in which the 3-D pose is computed from the object's appearance in the image, without using an explicit 2-D or 3-D

model. To compute the 3-D pose, we determine what 3-D pose could have resulted in the observed 2-D appearance. Ambiguities are handled by making simple assumptions. Most appearance-based approaches process the camera image and feed it into an neural network. The network is trained using camera images of various poses of a face, together with the ground truth of the corresponding facial normals.

In a strict sense, the model-based approach relies solely on prior knowledge while the appearance-based approach rejects the use of prior knowledge. As we will see in the next section, it is difficult to categorize most approaches; they can be regarded as lying between the two extremes, and differ in the amount of prior knowledge that they use. We will therefore call an approach model-based iff it uses any explicit knowledge about the 2-D or 3-D geometrical structure of the object. Otherwise, we will call it appearance-based. Appearance-based approaches may use prior knowledge about local image features, but may not use any knowledge about their geometric relations.

6.2 Related Work

Most pose detection approaches are model-based [Gee and Cipolla, 1994; Horprasert *et al.*, 1996; Ballard and Stockman, 1992; Petraki, 1996; Stiefelhagen *et al.*, 1997]. All of them use explicit prior knowledge about the 3-D geometry of faces. The major differences between the various approaches are the choice of the projection model and the face model. In [Gee and Cipolla, 1994] a weak perspective projection model is assumed. This model is simple and generic, and makes use of facial features that allow reliable estimation of facial pose across a wide variety of subjects. Geometric model knowledge is given by a set of four distances between the corners of the eyes, mouth, and nostrils. Gee and Cipolla argue that these cues do not change much for different facial expressions; however, they do not provide experimental evidence for this statement. They present experiments using two methods of estimating the facial normal. Both methods allow estimation of slant and tilt. The first method uses 3-D information provided by the above-mentioned facial features and the nose tip. The second method exploits planar skew-symmetry results from [D.Mukherjee *et al.*, 1993]. The authors report an accuracy of up to 3° for clean data and up to 6° for noisy data (zero-mean Gaussian noise with standard deviation 0.02). The implementation of the approach is rudimentary. A feature tracking algorithm tracks the five feature points. The tracking speed is reported as 100 Hz on a Sun Sparc 10, but no details are given about the tracking method that was used. The accuracy results are derived theoretically and were not verified in the on-line experiments.

In [Horprasert *et al.*, 1996] a perspective projection model is used. The same five points

are used as a face model, four points are located at the eye-corners and one on the nose-tip. In the absence of structure, five points are usually not sufficient for recovering orientation when a perspective model is used. The authors therefore combine the projective invariance of cross ratios (from face symmetry) and statistical modeling of face structure (from anthropometry) to estimate the rotation angles. The four points at the eye-corners constitute a line. From the orientation of this line, the roll angle (rotation angle about the face normal) can be directly recovered. To recover the slant angle, the cross ratio of the four eye corners is used under the assumption that they are collinear and that the eyes have equal width. To recover the tilt, other assumptions about face geometry are needed, which are variable with respect to gender, race and age [Chellappa *et al.*, 1995]; relevant data are taken from anthropometric tables. [Horprasert *et al.*, 1996] report an accuracy of 0.5° to 5.0°, but they do not specify whether clean or noisy data was used to achieve these results, and the results were not verified with on-line experiments.

Appearance-based approaches, which are fast but imprecise, are generally based on color [Chen *et al.*, 1998; Darrell *et al.*, 1996; Schiele and Waibel, 1995].

In [Schiele and Waibel, 1995] the face is tracked as a flesh-color blob. The slant angle is detected by feeding a 32×32 subsampled image of the flesh-color blob region into the 32×32 input neurons of a Multilayer Perceptron (MLP). The MLP consists of 50 hidden units, 3 output units that indicate the gaze directions *left*, *straight* and *right*, and 15 output units that correspond to possible head directions $(-70, -60, \ldots, +60, +70)$ degrees. The MLP was trained with four sets of 15 images of 7 different people. The 15 images corresponded to the different directions, ranging from -70 to $+70$ degrees. The experimental results showed 95.65% correct detection of the head directions *left*, *straight*, *right*, and an average error of 12° for the detected slant angle. The speed was ≈ 10 Hz on an HP9000/735.

In [Chen *et al.*, 1998] an extended color model is used to describe the flesh and hair color of the tracked person. The average error is claimed to be 6.8° (tilt), 5.7° (slant) and 2.9° (roll), but there was no investigation of stability.

An appearance-based approach similar to the one of [Schiele and Waibel, 1995] is investigated in [Ábrahám-Mumm, 1998; Bruske *et al.*, 1998]. The approach allows computation of slant and tilt. The head is again tracked as a color blob. A square at the detected blob position in the image defines a region of interest (ROI). Within the ROI, complex 2-D Gabor filters (see eq. 2.17) are homogeneously distributed. Different filtering schemes were investigated; within the ROI, the filters were homogeneously distributed on a lattice varying from 4×4 to 8×8 positions. At each of these positions, between four and eight differently oriented filters were applied and the range $[0, \pi)$ of possible orientations was equidistantly sampled.

The energies of the complex filter responses were fed into an ANN. A subspace variant of

the Local Linear Map (LLM) [Ritter *et al.*, 1991] is used as an ANN for learning the input-output mapping [Bruske and Sommer, 1998].

The results were very promising; the reported mean errors were between $0.64°$ (4×4 filters, 4 orientations (0, $\frac{\pi}{4}$, $\frac{\pi}{2}$ and $\frac{3}{4}\pi$)) and $0.58°$ (8×8 filters, 4 orientations) (see also Table 6.1). The errors were computed as follows: Let $\hat{p}$ be the estimated slant, $\hat{y}$ the estimated tilt and let p and y be the ground truth values. Then the error is defined as

$$e = \sqrt{(\hat{p} - p)^2 + (\hat{y} - y)^2} \ . \tag{6.2}$$

sampling scheme	mean error	max. error
3×3	0.87	2.78
4×4	0.64	1.88
6×6	0.61	1.74
8×8	0.58	1.82

Table 6.1. This table shows experimental results for pose estimation based on Gabor filter responses and an ANN.

An appearance-based approach to the estimation of gaze was presented in [Varchmin *et al.*, 1997]. In a first step, an adaptive color histogram segmentation method roughly determined a region of interest that includes the face. Within the ROI, facial features such as mouth edges, nostrils and pupils were detected. In the last stage, the feature positions and a detailed analysis of the eye regions were used to estimate the gaze direction: The feature positions and the images of the eyes provide the input to an LLM network. 625 training images were used to train the network. The user was required to fixate a 5×5 grid on the computer screen. The minimal errors after training for slant and tilt were $1.5°$ and $2.5°$, respectively, while the system speed was 1 Hz on a SGI.

In [Klingspohr *et al.*, 1997] an approach is presented that assumes the head pose is known, and computes the positions of the pupils relative to the head in order to compute the gaze direction. The approach detects the irises of the eyes using a Hough transform. The circles of the irises deform to ellipses when the eyes rotate. The approach performs robust parameter estimation of the ellipses. The accuracy was $2.3°$.

6.3 Head Pose Estimation with Gabor Wavelet Networks

The results reported in [Ábrahám-Mumm, 1998; Bruske *et al.*, 1998], were very promising. However, the filtering scheme that was used was rather rudimentary and straightforward. We will argue that this scheme has two major drawbacks that considerably limit the precision of the approach:

1. In Section 2.6 we explained that for GWNs, precise positioning of novel images before computing new weights is very important. This clearly also holds for the filtering in [Ábrahám-Mumm, 1998; Bruske *et al.*, 1998]. In that work, however, color blob tracking, which is a very imprecise tracking method, was used. In fact, the results in Table 6.1 were obtained only under optimal illumination conditions, and even under those conditions, the computation was very sensitive to noise. The results presented in Table 6.1 can be regarded as an experimentally evaluated upper bound on the precision that this approach can offer.

2. One can also question the homogeneous filtering scheme that was used and ask what information is contained in a set of Gabor filter responses when the filters are homogeneously distributed. We have analyzed this question in Section 3.3 and have found, that the loss of image data is severe.

It is therefore reasonable to assume that accurate selection of the parameters of each Gabor filter, and precise positioning of these wavelets in novel images, would result in a much lower mean slant/tilt error than that achieved in [Ábrahám-Mumm, 1998; Bruske *et al.*, 1998]. It is reasonable to hope that the stability with respect to illumination and camera noise would also increase considerably. Specifically, we argue that

- precise tracking, which assures exact positioning of the filters, increases accuracy, and in particular, robustness to illumination.

- precise and specific selection of the filter parameters (positions, scales and orientations) increases the accuracy of pose estimation. This allows a reduction in the number of filters and filter applications that are needed, which has a positive effect on computation speed.

We have therefore redone the experiments that were presented in [Ábrahám-Mumm, 1998]. We have tried to change only the tracking and the filtering and to avoid any changes in the neural network, the training, and the experimental setup. However, the camera used in our experiments was different. Also, in our experiments the slant and tilt angles had to be within intervals of

Figure 6.1. The first image shows the original doll face image I. The second and third images show the reconstructions $\hat{I}_{16}$a and $\hat{I}_{52}$ with $N = 16$ and $N = 52$ wavelets, respectively.

$\pm 20^\circ$. This is the interval in which all the facial features are visible to the camera, which is a requirement for successful tracking.

The experimental setup in [Ábrahám-Mumm, 1998]'s and our experiments was as follows: The head of a doll, shown in Fig. 6.1, was connected to a robot arm. This setup was used so that the ground truth values for slant and tilt were known. A monocular camera was positioned at a distance of approximately 100 cm and its visual axis was directed approximately toward the origin of the head's coordinate system. The same ANN was used in both experiments [Bruske *et al.*, 1998]. In both [Ábrahám-Mumm, 1998]'s and our experiments, the best experimental results were obtained with 400 training samples.

In our experiments we replaced the color tracking approach by the wavelet tracker of Chapter 4. We optimized a GWN on the doll's head (Fig. 6.1). For optimization of the GWN we again used the optimization scheme that was introduced in Section 2. We used a GWN with $N = 52$ wavelets (see Fig. 6.1, right image).

For training we used 400 images that showed the doll's head with different slant and tilt angles, each within $\pm 20^\circ$, with 2° steps. For testing we used 200 images of the doll's head, while the slant and tilt were randomly chosen from the $\pm 20^\circ$ interval.

For each training and test frame we proceeded in two steps:

1. Optimal reparameterization of the GWN by using the positioning operator $\mathcal{P}$. This was done automatically by the tracker.

2. Calculation of optimal weights for the optimally repositioned GWN using the projection operator $\mathcal{T}$.

Fig. 6.2 shows some example images.

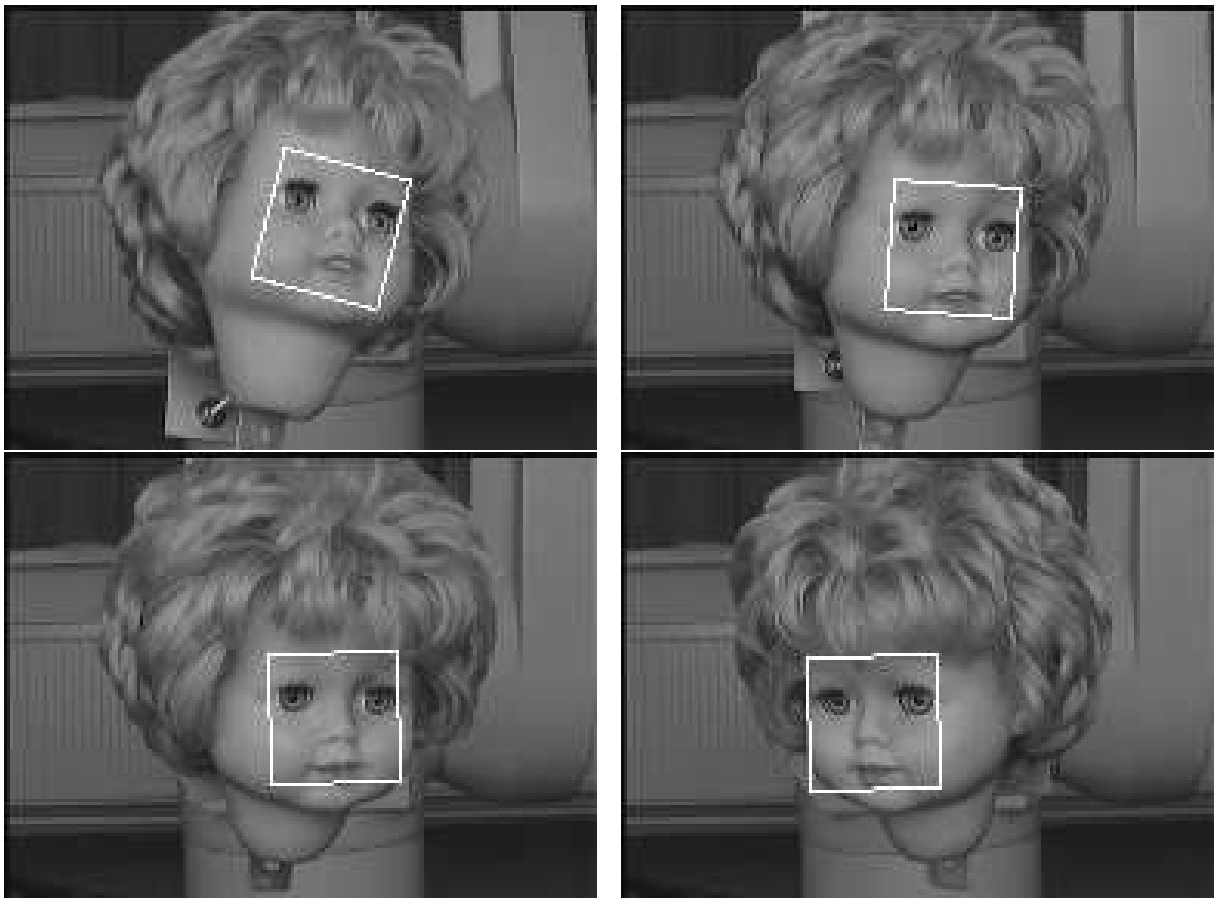


Figure 6.2. These images show different orientations of the doll's head. The head is connected to a robot arm so that the ground truth is known. The white square indicates the detected position, scale and orientation of the GWN.

6.3.1 Experimental Results

The weight vector that was calculated with the operator $\mathcal{T}$ was then fed into the ANN [Bruske *et al.*, 1998]. We achieved a minimal mean slant/tilt error of 0.21° for a GWN with 52 wavelets and a minimal mean slant/tilt error of 0.30° for a GWN with 16 wavelets. The maximal errors were 0.65° for 52 wavelets and 0.72° for 16 wavelets, respectively. These results show that when we use the GWN with 52 wavelets, even the maximal error is as low as the mean error of the 8×8 filtering scheme with four orientations (0.58°). The errors were calculated according to the error function in eq. (6.2). The experiments were carried out under varying illumination conditions, and the results were reproducible. A summary and a comparison with the approaches that were mentioned above is given in Table 6.2.

method	minimal mean error
geometrical approach[Gee and Cipolla, 1994; Petraki, 1996]	$1.6°$
color, ANN [Schiele and Waibel, 1995]	$12 - 15°$
stereo information[Xu and Akatsuka, 1998]	$\approx 4°$
Gabor filter, ANN[Bruske *et al.*, 1998]	$0.64°$
GWN with 16 Gabor wavelets	$0.30°$
GWN with 52 Gabor wavelets	$0.21°$

Table 6.2. This table shows a summary of different approaches and the minimal mean errors for slant/tilt angle estimation of the head pose that were achieved.

6.3.2 Introduction to DCS Networks

In this subsection we briefly discuss the artificial neural network (ANN) that was employed in our pose estimation experiments, using the description and terminology in [Bruske, 1998].

The ANN is a Dynamic Cell Structure (DCS) based network [Bruske and Sommer, 1995] which was introduced and further enhanced in [Bruske and Sommer, 1995; Bruske and Sommer, 1998]. DCS-based networks are RBF-based ANNs that utilize an efficient local subspace construction method based on optimally topology preserving maps (OTPM).

The architectural characteristics of a DCS network are sketched in Fig. 6.3. It shows (1) a hidden layer with RBFs with possibly variable parameters, (2) a dynamic layer with a lateral connection structure between basis functions (units), and (3) a layer of output units. During training, a competitive Hebbian learning rule is used to activate and adapt the RBF units in the neighborhood of the current stimulus. The neighborhood relation is given by the simultaneously learned topology. Using the Hebbian learning rule adapts the lateral connection structure to an OTPM.

In this chapter we use a subspace variant of Ritter's Local Linear Map (LLM) [Ritter *et al.*, 1991], which is called a Subspace-DCS (SDCS) based network [Bruske and Sommer, 1998]. The SDCS allows us to exploit the fact that images of the head of a single person that differ solely in slant and tilt lie on a 2-D manifold in image space [Murase and Nayar, 1995; McKenna *et al.*, 1996].

The SDCS enhances the DCS by applying principal component analysis (PCA) to each local subspace. Given a training set $T \subset \mathbb{R}^n$ and an $N > 0$, the batch-variant proceeds in four stages:

1. A set of N centers $S = \{\mathbf{c}_1, \ldots, \mathbf{c}_N\}$ are computed as the output of a vector quantization

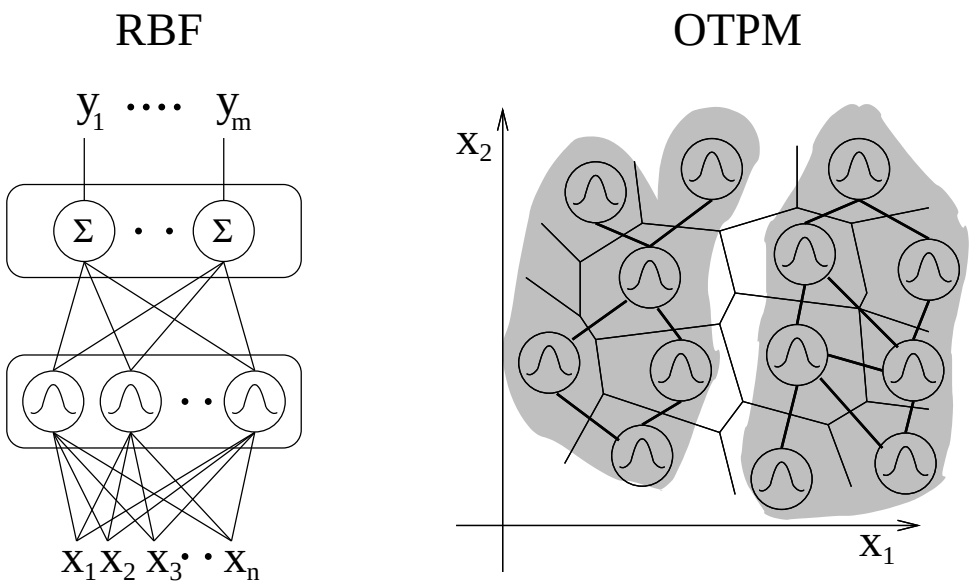


Figure 6.3. DCS networks are RBF networks (left) with an additional lateral connection structure between the nodes. The connections are formed by competitive Hebbian learning and approximately Optimal Topology Preserving Maps (OTPMs) (right).

algorithm applied to the training set T.

2. The graph G is calculated as an optimally topology-preserving map, $OPTM_T(S)$, of S given T.

3. For each node $i \in G$, PCA is performed on the set of m_i difference vectors $(\mathbf{c}_{1_i} - \mathbf{c}_i, \ldots, \mathbf{c}_{m_i} - \mathbf{c}_i)$, where $(\mathbf{c}_{j_i} - \mathbf{c}_i)$ is the difference vector between $\mathbf{c}_i$ and a directly neighboring center $\mathbf{c}_{j_i}$.

4. The eigenvectors that correspond to the smallest eigenvalues are discarded.

The results of this four-stage process are N sets of eigenvectors $\{e_1^i, \ldots, e_{l_i}^i\}$, $l_i < m_i$ that span a local subspace with center c_i. These eigenvectors allow us to project an input stimulus x into the relevant subspace, i.e. the subspace of the best matching unit (bmu):

$$\mathbf{x}^s = \mathbf{c}_{bmu} + \sum_{i=1}^{l_{bmu}} \left((\mathbf{x} - \mathbf{c}_{bmu})^T e_i^{bmu} \right) e_i^{bmu} , \tag{6.3}$$

where $\mathbf{c}_{bmu}$ is the center of the best matching unit.

The advantage of the SDCS network is clear: Discarding small eigenvectors allows us to reduce noise in the input. Furthermore, limiting the number of eigenvectors used allows us to reduce the complexity of the ANN so that its application becomes feasible even for very high dimensional input spaces.

6.4 Progressive Attention Scheme for Pose Estimation

If gaze detection systems are to be included in a human-computer interface, real-time speed is a major requirement. At the same time, the HCI is allowed to consume only a small portion of the available computer power. Keeping the number of filterings low is clearly a prerequisite to achieving this goal.

It is possible to apply a progressive attention scheme in this situation. Changing the numbers of Gabor wavelets used allows control of the tracking, as we have seen in Chapter 4. Just as the progressive attention scheme can be used to control the precision of a template for visual tracking, it can also be used to control the number of filterings and the filtering speed. In this section we investigate how the precision of pose estimation changes with the number N of Gabor wavelets used. We will also investigate how the efficiency can be further increased by taking the responses of the Gabor filters as inputs instead of the weights.

Taking the filter responses as inputs to the ANN has a further advantage over taking the weights as inputs. Since the Gabor wavelets are non-orthogonal, the weights depend on all the wavelets that are used (see eq. 2.33). When the number of wavelets is increased, the vector input to the ANN must be completely recomputed so that network training has to be completely redone. The filter responses, on the other hand, can be used directly, without intermediate projections, as presented above. Their values are independent of the number of filters used, so that when their responses are used as inputs, only a single ANN needs to be retrained.

In our experiments, GWNs of different sizes were used. The GWNs were all derived from the GWN of the preceding section by choosing the wavelets in order of their normalized decreasing weights (see Section 3.2). When a GWN of a certain size was used, the computed weights (filter responses) were fed into the neural network.

Figure 6.4 shows examples of GWNs with 16, 20, 32, 40 and 52 wavelets derived from the GWN of the preceding section.

We then used the GWNs for tracking and filtering, in order to compute for each GWN the mean and the maximal error.

Figure 6.4. These images show different GWNs for the puppet head, with 16, 20, 32, 40 and 52 wavelets.

6.4.1 Experimental Results

The resulting estimation errors (in degrees) for GWNs with 4 - 52 wavelets are shown in Fig. 6.5. The weight vectors were used as input to the ANN.

Fig. 6.6 shows the mean errors and the maximal errors (in degrees) of the pose estimations computed from the filter responses, for GWNs with 4 - 52 wavelets.

The results are quite similar for both experiments. In Fig. 6.7 the mean estimation errors in Figs. 6.5 and 6.6 are plotted against each other. The errors for GWNs with 16-52 wavelets are shown. A summary of the results is given in Table 6.3.

	weights		responses	
Number of Wavelets	mean error	max. error	mean error	max. error
16	0.30	0.72	0.37	0.91
52	0.21	0.65	0.23	0.53

Table 6.3. This table gives a summary of the estimation errors with varying numbers of Gabor Wavelets. Shown are the mean and maximum errors for the experiments on the weights and on the filter responses.

6.5 Discussion and Conclusions

In this chapter we have demonstrated that GWNs offer an optimized scheme for the filtering of images.

In image filtering one always wants to extract information out of the image. An optimized filtering scheme allows the extraction of a maximum amount of image information for a given number of applied filters.

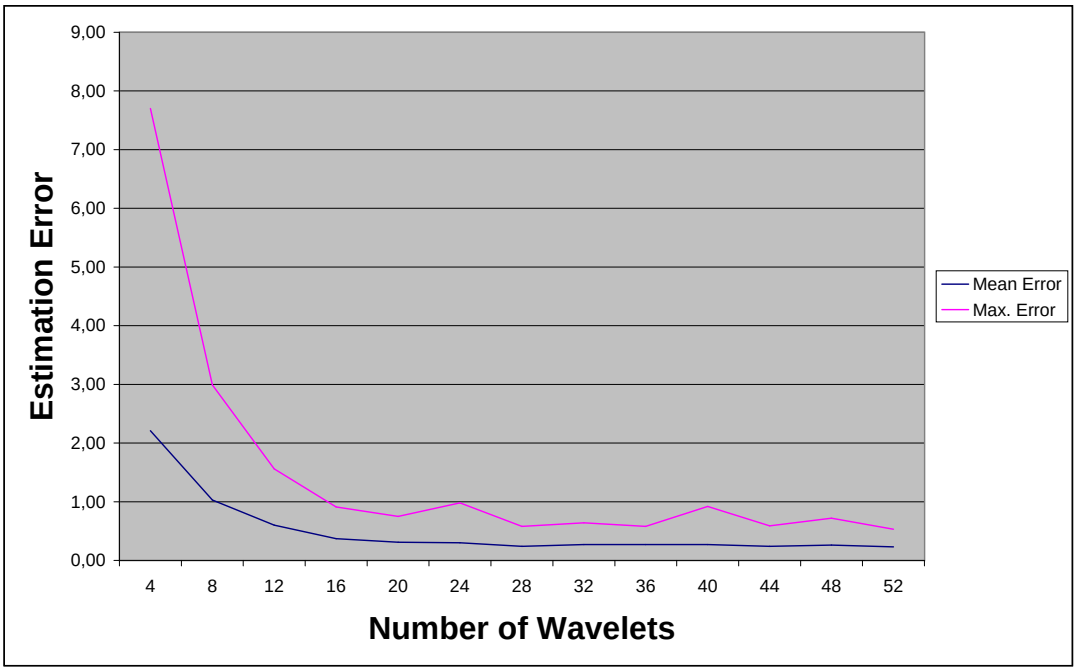


Figure 6.5. This figure shows the decrease in the error in pose estimation with an increasing number of wavelets. For these plots, the weights were computed with the operator $\mathcal{T}$, and were fed into the ANN. Shown are plots of the mean error and the maximal error (in degrees). The wavelets were chosen according to the progressive attention scheme, in decreasing order.

This was seen in the results of our pose estimation experiment: When we used an optimized filtering scheme, the mean error of the estimated poses was much smaller than when a homogeneous filtering scheme was used. Furthermore, a smaller mean error was achieved with as few as 16 filterings (in comparison with 128 filterings!).

An optimized filtering scheme also allows us to reduce the complexity of subsequent computations.

So far, the term *optimized filtering* has been used in a rather intuitive manner. *Optimized filtering* is usually related to a certain task: The task defines the relevant data, and *optimized filtering* allows the relevant data to be extracted efficiently. Prior knowledge about what data is relevant to a certain task is therefore needed. In this chapter the given task was estimation of the

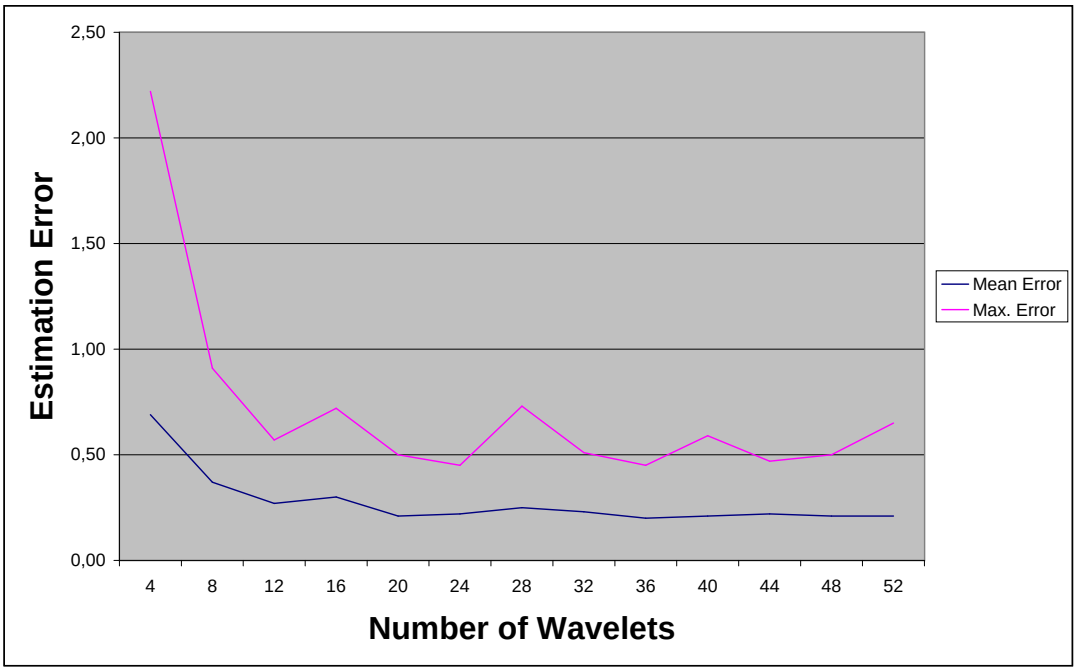


Figure 6.6. This figure shows the decrease in the error in pose estimation with an increasing number of wavelets. For these plots, the filter responses were directly fed into the neural network. Shown are plots of the mean error and the maximal error (in degrees). The wavelets were chosen according to the progressive attention scheme, in decreasing order.

pose of a head. Precisely what the relevant data is in this context is difficult to answer. There are two possibilities:

1. Finding the relevant data, i.e. finding the right filtering scheme for extracting the relevant data, could be done by learning.

2. Alternatively, one can simply try to use *all* the image data. The goal is then to find an efficient filtering scheme that allows us to extract all the data. The filtering scheme in this case is found beforehand and is optimized so that the number of filters used is small and the amount of extracted data is large.

Clearly, the first possibility needs further research and investigation. The second possibility,

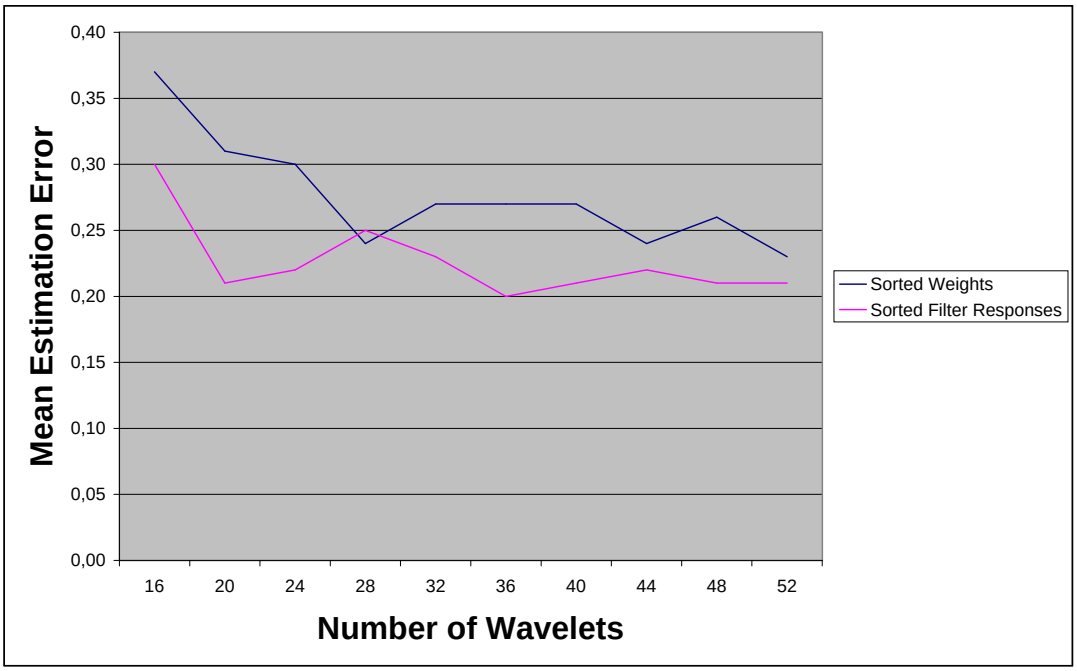


Figure 6.7. This figure shows the decrease in the error in pose estimation with an increasing number of wavelets. Shown are plots of the mean errors that were obtained from the weights and from the filter responses. The wavelets were chosen according to the progressive attention scheme, in decreasing order.

however, is solved by the GWNs.

The amount of information is here measured by the sum of squared differences (SSD) between the original image and its reconstruction. The loss in information is consequently given by the error in eq. (2.23). In Chapter 4 we argued that the progressive attention scheme allows a task-oriented representation. There the task was *visual face tracking*. The tracking was done by minimizing the SSD between an input image and a template image. The template was given as a GWN, and since the wavelets were chosen so that the energy functional (2.23) was minimized, the more wavelets were used, the smaller was the energy functional and the more stable was the tracking.

In this chapter, the progressive attention property of GWNs has been extended to image

filtering. The more image information is needed, the more filtering is done, and the better is the precision of the results. This could clearly be seen in the results in Section 6.4: The more filter responses were used, the smaller was the mean error of the estimated pose.

In connection with the gaze detection problem, we think that the approach presented here can be extended to recover the gaze direction of a person. For this we propose using a large GWN for the analysis of the head, and two smaller GWNs for the analysis of the eye regions. The large GWN can be used for tracking. This would allow automatic positioning of the two small GWN at the correct positions. An ANN can then be trained on the filter responses of all three networks.

The training can even be done automatically while the user is working on the computer: Assuming that the user gazes at the mouse pointer when he/she clicks the mouse, each mouse click supplies a ground truth value.

As explained in Chapter 4, the tracking approach allows affine tracking of a face. Assuming weak perspective, it is possible to derive the pose of the head from the affine deformation parameters of the reparameterized superwavelet. However, this remains to be investigated. The stability of this approach should correlate with the stability results for the tracking approach with respect to the progressive attention scheme.

Chapter 7

Conclusions and Outlook

In this thesis we have given an extensive and thorough introduction to Gabor Wavelet Networks. We have shown through various experiments that GWNs can be used for an efficient and task-specific representation of individual objects. The optimization scheme of GWNs allows us to find networks that reflect individual object properties. The optimized networks are then individual enough to allow reliable identification. Furthermore, the representation is robust with respect to minor local changes, which means, in the case of face recognition, that individuals can be recognized in spite of different facial expressions and poses. In our experiments, we achieved recognition rates as high as 96%. On the "surprised" expression, the recognition approach often failed, which shows a limitation of the GWN approach. Without this particular expression, we would have reached recognition rates of almost 98%. The recognition rates were reached in our experiments straightforwardly and without the use of any heuristics. Most recognition approaches, such as those used in the FERET test [Phillips *et al.*, 1998], made extensive use of heuristics to increase recognition rates for the specific test set. Using such heuristics, it is likely that the recognition rate of our approach could be further increased.

Apart from object representation, we have shown that GWNs can also be used as an optimized scheme for filtering. We have shown that there is a close relationship between image filtering and image representation. Consequently, it was reasonable to assume that GWNs could also be used for optimized filtering, i.e. to extract a maximal amount of image information from an image. This property has been tested in a pose estimation experiment. The experiment showed that a GWN, used as an optimized filtering scheme, is able to improve pose estimation by up to a factor of three, in comparison with an often-used homogeneous filtering scheme: from 0.64° with 128 homogeneously distributed filters to 0.3° with only 16 optimized GWN filters, and 0.21° with 52 filters. Moreover, these results were achieved even in real experiments

involving, e.g., camera noise. Also, in this experiment no heuristics were used. The results were achieved straightforwardly by simply applying the techniques provided by the GWN approach.

In our opinion, our results are due to the fact that GWNs combine the advantages of appearance-based and feature-based approaches. On the one hand, GWNs are able to capture and evaluate all the pixel value information*, which is an appearance-based feature. On the other hand, GWNs are representations that are, through optimization, closely linked to the object features. This is clearly a feature-based property that common appearance-based approaches do not have. The feature-based representation adds considerable robustness with respect to changes in illumination, contrast, affine variations and local image changes. The reason for this robustness is that Gabor filters are good feature detectors. For example, during reparameterization, each one of the Gabor wavelets "looks" for a local feature, i.e. it "looks" for a local minimum. When summed, this leads to a steep and deep local minimum, which is clearly a great advantage for the reparameterization procedure.

Our experiments have revealed a further property of GWNs, which is probably even more important than the properties mentioned above, and which we denoted by the term *progressive attention*. The progressive attention property of GWNs allows us to control the precision and complexity of the representation by dynamically varying the number of Gabor functions used. Wavelet theory supplies the mathematical basis for this [Daubechies, 1992; Louis *et al.*, 1994]. Dynamic perception is an important preliminary to successful construction of active vision systems, because it allows the cost and complexity of the successive computations to be controlled [Bajcsy, 1988; Bajcsy, 1992; Sandini and Dario, 1990; Aloimonos, 1994; Sommer, 1995].

From the definition of GWNs, their close relationship to neural networks is obvious. Indeed, the name *Gabor Wavelet Network* derives from this fact. However, as pointed out by [Reyneri, 1999], GWNs, or wavelet networks in general, introduce a completely new type of neural network, closely related to Radial Basis Function Networks (RBF Networks). But distinct differences have to be pointed out. RBF Networks appear to be traditionally associated with radial functions in a single-layer network [Broomhead and Loewe, 1988]. The characteristic of radial functions is that their response decreases (or increases) monotonically with increasing distance from a central point. In contrast, the mother wavelets, used for wavelet networks, are not necessarily radial functions. In particular, the odd Gabor function, which is the mother wavelet of the GWNs, is non-radial.

There are advantages in the fact that one can choose a function which particularly suits a given problem. Odd Gabor functions, e.g., have been shown to be very useful for the represen-

*The mean is discarded and the contrast is normalized.

tation of faces. Furthermore, since they are good edge detectors, one can predict and understand their roles and properties within the networks; this has been discussed in Chapter 3. This, in fact, is a further important difference: RBF networks are regarded as *non-parametric* models since their weights and other parameters are not meant to have any particular meaningful relation to the problem they are applied to. Estimating values for the weights and parameters is never the primary goal. Instead, the primary goal is to approximate the underlying function [Orr, 1999]. On the other hand, for GWNs (or wavelet networks in general), estimating weights and parameters and approximating the underlying functions are two closely related tasks. While the radial basis functions of RBF networks are generally completely *independent* of the data they are supposed to represent, the basis functions of (Gabor) Wavelet Networks are particularly *intended* (and chosen) to reflect the properties of the underlying functions.

Wavelet networks have received little attention in recent publications. Lately, in [Reyneri, 1999], the relations between artificial neural networks, wavelet networks and fuzzy systems have been discussed, but wavelet networks were considered only in a very simplified fashion: Only radial wavelets were considered, which limits the potential of wavelet networks considerably.

We would like to argue that, because of the close relation between the data and the basis functions, (Gabor) Wavelet Networks offer new potential that goes, beyond the potential of RBF Networks. At least for 2-D functions and the shapes of human faces, this has been partially shown in this thesis. We think that this can be generalized to other N-D functions. The application of (Gabor) Wavelet Networks in classification problems also needs closer investigation.

Notation

Overview of the mathematical symbols and notation, in order of their use.

$\langle f, g \rangle$	Scalar product in $\mathbb{L}^2(\mathbb{R}^n)$: $\langle f, g \rangle = \int_{-\infty}^{\infty} f(x)g(x)\,dx$
$DC(f)$	DC of function f: $DC(f) = \int_{-\infty}^{\infty} f(x)\,dx$
$\mathbf{X}^T$, $\mathbf{x}^T$	Transposed matrix and vector, respectively
diag	Diagonal matrix
$\mathbb{L}^2(\mathbb{R})$	Space of square integrable functions $f : \mathbb{R} \longmapsto \mathbb{R}$
$\mathbb{L}^2(\mathbb{R}^2)$	Space of square integrable functions $f : \mathbb{R}^2 \longmapsto \mathbb{R}$
ψ, ϕ	1-D or 2-D mother wavelet
$\psi_{a,b}$	1-D wavelet ψ with 1-D dilation parameter a and 1-D translation parameter b
$\mathbf{c}$	Translation vector $\mathbf{c} = (c_x c_y)^t$
s_x, s_y	Dilation parameter
s_{xy}	Shear parameter of superwavelet
θ	Rotation parameter
$\mathbf{n}$	Parameter vector of wavelet $\mathbf{n} = (c_x, c_y, \theta, s_x, s_y)^T$
	Parameter vector of superwavelet $\mathbf{n} = (c_x, c_y, \theta, s_x, s_y, s_{xy})^T$
$\mathbf{R}$	Rotation matrix
$\mathbf{S}$	Dilation matrix
N	Number of wavelets within a wavelet network
w_i	i-th weight of weight vector $\mathbf{w}$
$\psi_{\mathbf{n}_i}, \phi_{\mathbf{n}_i}$	i-th wavelet of a family of wavelets Ψ, $\mathbf{\Phi}$, with parameter vector $\mathbf{n}_i$
$\mathbf{v}, \mathbf{w}, \mathbf{v}', \mathbf{w}'$	Weight vectors $\mathbf{w} = (w_1, \ldots, w_N)^T$, etc...
$\Psi, \mathbf{\Phi}$	Family of wavelets $\Psi = (\psi_{\mathbf{n}_1}, \ldots, \psi_{\mathbf{n}_N})^T$
$(\Psi, \mathbf{w})$	Gabor Wavelet Network

f, g	Continuous functions
$\hat{f}, \hat{g}$	Remaps of functions f, g, represented with a Gabor Wavelet Network
I, J	Discrete gray value images
$\hat{I}, \hat{J}$	Remaps of images I, J, represented with a Gabor Wavelet Network
$\boldsymbol{\delta}$	Difference vector between two weight vectors $\boldsymbol{\delta} = (\mathbf{v} - \mathbf{w})$
δ_i	t-th component of $\boldsymbol{\delta}$
$\delta_{i,j}$	Dirac function $\delta_{i,j} = \begin{cases} 1, & \text{if } i = j \\ 0, & \text{if } i \neq j \end{cases}$.
$(\Psi_{i,j})$	Gram's matrix $(\Psi_{i,j}) = (\langle \psi_i, \psi_j \rangle)$
$\Psi_{\mathbf{n}}$	Superwavelet
$d^2_{\Psi}(\cdot,\cdot)$	Euclidean distance measurement, defined with respect to the wavelet family Ψ.
$d^c_{\Psi}(\cdot,\cdot)$	Normalized cross-correlation between two images, defined with respect to the wavelet family Ψ.
$\mathcal{G}$	Gallery: set of gallery images
$\mathcal{U}$	Probe image
VAR$(\cdot)$	Variance
SSD$(\cdot,\cdot)$	Sum of squared differences between two images
$\tilde{\phi}, \tilde{\psi}$	Function that is dual (biorthogonal) to ϕ, ψ
$\tilde{\boldsymbol{\Phi}}, \tilde{\Psi}$	Family of functions that is dual to $\boldsymbol{\Phi}$
$\mathcal{T}_{\Psi}(f)$	Operator to compute optimal weight vector for image f with respect to wavelet family Ψ
$\mathcal{P}_{\Psi}(g)$	Reparameterization operator: reparameterized superwavelet $\Psi_{\mathbf{n}}$ on function g, such that E is minimized
E	Energy functional

List of Figures

List of Tables

Bibliography

[Ábrahám-Mumm, 1998] E. Ábrahám-Mumm. Bestimmung der Gesichtspose mit künstlichen neuronalen Netzen. Master's thesis, University of Kiel, 1998.

[Aloimonos, 1993] Y. Aloimonos. Active vision revisited. In Y. Aloimonos, editor, *Active Perception*, pages 1–18. Lawrence Erlbaum, Hillsdale, 1993.

[Aloimonos, 1994] Y. Aloimonos. What I have learned. *Computer Vision, Graphics, and Image Processing*, 60:74–85, 1994.

[A.S.Georghiades *et al.*, 1999] A.S.Georghiades, P.N. Belhumeur, and D.J. Kriegman. Illumination-based image synthesis: Creating novel images of human faces under differing pose and ighting. In *Proc. IEEE Workshop on Multi-View Modeling and Analysis of Visual Scenes*, pages 47–54, 1999.

[Bajcsy, 1988] R. Bajcsy. Active perception. *Proceedings of the IEEE*, 76:996–1005, 1988.

[Bajcsy, 1992] R. Bajcsy. Active and exploratory perception. *Computer Vision, Graphics, and Image Processing*, 56:31–40, 1992.

[Ballard and Stockman, 1992] P. Ballard and G.C. Stockman. Computer operation via face orientation. In *Proc. Int. Conf. on Pattern Recognition*, pages 407–410, The Hague, The Netherlands, Aug. 30-Sep. 3, 1992.

[Belhumeur *et al.*, 1997] P.N. Belhumeur, J.P. Hespanha, and D.J. Kriegman. Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19:711–720, 1997.

[Bishop, 1995] C.M. Bishop. *Neural Networks for Pattern Recognition*. Clarendon Press, Oxford, 1995.

[Blake and Isard, 1998] A. Blake and M. Isard. *Active Contours*. Springer, 1998.

[Broomhead and Loewe, 1988] D.S. Broomhead and D. Loewe. Multivariate functional interpolation and adaptive networks. *Complex Systems*, 2:321–355, 1988.

[Brown and Terzopoulos, 1994] C. Brown and D. Terzopoulos, editors. *Real-time Computer Vision*. Publications of the Newton Institute, 1994.

[Brown, 1994] C. Brown. Towards general vision. *Computer Vision, Graphics, and Image Processing*, 60:89–91, 1994.

[Brunelli and Poggio, 1993] R. Brunelli and T. Poggio. Face recognition: Features versus templates. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 15:1042–1052, 1993.

[Bruske and Sommer, 1995] J. Bruske and G. Sommer. Dynamic cell structure learns perfectly topology preserving map. *Neural Computation*, 7:845–865, 1995.

[Bruske and Sommer, 1998] J. Bruske and G. Sommer. Intrinsic dimensionality extimation with optimally topology preserving maps. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 20:572–575, 1998.

[Bruske *et al.*, 1998] J. Bruske, E. Abraham-Mumm, J. Pauli, and G. Sommer. Head-pose estimation from facial images with subspace neural networks. In *Proc. Int. Neural Network and Brain Conf.*, pages 528–531, Beijing, China, 1998.

[Bruske, 1998] J. Bruske. *Dynamische Zellstrukturen*. PhD thesis, University of Kiel, June 1998. Technical Report 9809.

[Calderón, 1964] A. P. Calderón. Intermediate spaces and interpolation, the complex method. *Stud. Math.*, 24:113–190, 1964.

[Canny, 1986] J. Canny. A computational approach to edge detection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 8:679–698, 1986.

[Chellappa *et al.*, 1995] R. Chellappa, C. Wilson, and S. Sirohey. Human and machine recognition of faces: A survey. *Proceedings of the IEEE*, 83:704–740, 1995.

[Chen *et al.*, 1998] Q. Chen, H. Wu, T. Shioyama, T. Shimada, and K. Chihara. A robust algorithm for 3D head pose estimation. In *Proc. Int. Conf. on Pattern Recognition*, pages 1356–1359. Brisbane , Australia, Aug. 16-20, 1998.

[Chui, 1992] C.K. Chui. *An introduction to wavelets*. Academic Press, 1992.

[Colombo and Bimbo, 1997] C. Colombo and A. Del Bimbo. Interaction through eyes. *Robotics and Autonomous Systems*, 19:359–368, 1997.

[Cootes *et al.*, 1998] T.F. Cootes, G.J. Edwards, and C.J. Taylor. Active appearance models. In *Proc. European Conf. on Computer Vision*, volume 2, pages 484–498, Freiburg, Germany, June 1-5, 1998.

[Costen *et al.*, 1996] N.P. Costen, I.G. Craw, G.J. Robertson, and S. Akamatsu. Automatic face recognition: What representation. In *Proc. European Conf. on Computer Vision*, volume 1, pages 504–513, Cambridge, UK, April 15-18, 1996.

[Cox *et al.*, 1996] I. Cox, J. Ghosn, and P. Yianilos. Feature-based face recognition using mixture-distances. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 209–216, Seattle, WA, June 21-23, 1996.

[Craw *et al.*, 1999] I. Craw, N. Costen, T. Kato, and S. Akamatsu. How should we represent faces for automatic recognition. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 21:725–736, 1999.

[Darrell *et al.*, 1996] T. Darrell, B. Moghaddam, and A. Pentland. Active face tracking and pose estimation in an interactive room. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 67–72, Seattle, WA, June 21-23, 1996.

[Daubechies, 1988] I. Daubechies. Orthonormal bases of compactly supported wavelets. *Commun. Pure Appl. Math*, 41:909–996, 1988.

[Daubechies, 1990] I. Daubechies. The wavelet transform, time-frequency localization and signal analysis. *IEEE Trans. Information Theory*, 36, 1990.

[Daubechies, 1992] I. Daubechies. *Ten Lectures on Wavelets*. Society of Industrial and Applied Mathematics, 1992.

[Daugman, 1985] J. Daugman. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized two-dimensional visual cortical filters. *J. Opt. Soc. Am.*, 2:1160–1168, 1985.

[Daugman, 1988] J. Daugman. Complete discrete 2D Gabor transform by neural networks for image analysis and compression. *IEEE Trans. Acoustics, Speech, and Signal Processing*, 36:1169–1179, 1988.

[Daugman, 1997] J. Daugman. Face and gesture recognition: Overview. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(7):675–676, 1997.

[Delagnes *et al.*, 1995] P. Delagnes, J. Benois, and D. Barba. Active contour approach to object tracking. *Pattern Recognition Letters*, 15:171–178, 1995.

[Denzler and Niemann, 1996] J. Denzler and H. Niemann. 3D data driven prediction for active contour tracking with application to car tracking. In *Machine Vision Application*, pages 204–207, Tokyo, Japan, 1996.

[D.Mukherjee *et al.*, 1993] D.Mukherjee, A. Zisserman, and M. Brady. Shape form symmetry – Detecting and exploiting symmetry in affine images. Technical Report OUEL 1988/93, Oxford University Department of Engineering Science, June 1993.

[du Buf, 1993] J.M.H du Buf. Response of simple cells: Events, interferences, and ambiguities. *Biological Cybernetics*, 68:321–333, 1993.

[Edelman *et al.*, 1992] S. Edelman, D. Reisfield, and Y. Yeshurun. Learning to recognize faces from examples. In *Proc. European Conf. on Computer Vision*, pages 787–791, Santa Margherita, Italy, May 23-26, 1992.

[Edwards *et al.*, 1998] G.J. Edwards, T.F. Cootes, and C.J. Taylor. Face recognition using active appearance models. In *Proc. European Conf. on Computer Vision*, volume 2, pages 581–595, Freiburg, Germany, June 1-5, 1998.

[Faugeras, 1993] O. Faugeras. *Three-dimensional Computer Vision*. MIT Press, Cambridge, MA, 1993.

[Feichtinger and Strohmer, 1998] H.G. Feichtinger and T. Strohmer, editors. *Gabor Analysis and Algorithms*. Birkhäuser, Boston, 1998.

[Funt *et al.*, 1998] B. Funt, K. Barnard, and L. Martin. Is machine colour constancy good enough? In *Proc. European Conf. on Computer Vision*, pages 445–459, Freiburg, Germany, June 1-5, 1998.

[Gabor, 1946] D. Gabor. Theory of Communications. *J. of Inst. of Electronical Engineering*, 93:429–457, 1946.

[Gavrila, 1999] D. Gavrila. The visual analysis of human movement: A survey. *Computer Vision and Image Understanding*, 73:82–98, 1999.

[Gee and Cipolla, 1994] A. Gee and R. Cipolla. Determining the gaze of faces in images. *Image and Vision Computing*, 12:639–647, 1994.

[Govindaraju, 1996] V. Govindaraju. Locating human faces in photographs. *Int. J. of Computer Vision*, 19:129–146, 1996.

[Granlund, 1997] G. H. Granlund. From multidimensional signals to the generation of responses. In G. Sommer and J.J. Koenderink, editors, *Algebraic Frames for the Perception-Action Cycle*, pages 29–53, Kiel, Germany, September, 1997.

[Grossmann and Morlet, 1984] A. Grossmann and J. Morlet. Decomposition of hardy functions into square integrable wavelets of constant shape. *SIAM J. Math. Anal.*, 15:723–736, 1984.

[Haralick and Shapiro, 1992] R.M. Haralick and L.G. Shapiro. *Computer and Robot Vision*. Addison-Wesley, Reading, MA, 1992.

[Hayakawa, 1994] H. Hayakawa. Photometric stereo under a light-source with arbitrary motion. *J. Opt. Soc. Am.*, 11:3079–3089, 1994.

[Herpers and Sommer, 1998] R. Herpers and G. Sommer. An attentive processing strategy for the analysis of facial features. In H. Wechsler et al., editor, *Face Recognition: From Theory to Applications*, pages 457–468. Springer, 1998.

[Herpers *et al.*, 1995] R. Herpers, H. Kattner, H. Rodax, and G. Sommer. GAZE: An attentive processing strategy to detect and analyze t he prominent facial regions. In *Proc. Int. Workshop on Automatic Face and Gesture Recognition*, pages 214–220, Zurich, Switzerland, June 26-28, 1995.

[Hong *et al.*, 1998] H. Hong, H. Neven, and C.von der Malsburg. Online facial expression recognition based on personalized galleries. In *Proc. Int. Conf. on Automatic Face and Gesture Recognition*, Nara, Japan, April 14-16, 1998.

[Hong, 1991] Z.-Q. Hong. Algebraic feature extraction of images for recognition. *Pattern Recognition*, 24:211–219, 1991.

[Horn, 1986] B.K.P. Horn. *Computer Vision*. MIT Press, Cambridge, MA, 1986.

[Horprasert *et al.*, 1996] T. Horprasert, Y. Yacoob, and L. Davis. Computing 3-D head orientation from a monocular image sequence. In *Proc. Int. Conf. on Automatic Face and Gesture Recognition*, pages 242–247, Killington, VT, Oct. 14-16, 1996.

[Ishikawa *et al.*, 1998] T. Ishikawa, H. Serq, S. Morishima, and D. Terzopoulos. Facial image reconstruction by estimated muscle parameters. In *Proc. Int. Conf. on Automatic Face and Gesture Recognition*, Nara, Japan, April 14-16, 1998.

[Jain, 1994] R. Jain. Expansive vision. *Computer Vision, Graphics, and Image Processing*, 60:86–88, 1994.

[Jaquin and Eleftheriadis, 1995] A. Jaquin and A. Eleftheriadis. Automatic location tracking of faces and facial features in video squences. In *Proc. Int. Workshop on Automatic Face and Gesture Recognition*. Zurich, Switzerland, June 26-28, 1995.

[Jolliffe, 1986] I. Jolliffe. *Principal Component Analysis*. Springer Verlag, New York, 1986.

[Jones and Palmer, 1987] J. Jones and L. Palmer. An evaluation of the two-dimensional gabor filter model of simple receptive fields in cat striate cortex. *J. of Neurophysiology*, 58(6):1233–1258, 1987.

[Kirby and Sirovich, 1990] M. Kirby and L. Sirovich. Application of the Karhunen-Loève procedure for characterization of human faces. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 12:103–108, 1990.

[Klingspohr *et al.*, 1997] H. Klingspohr, T. Block, and R.-R. Grigat. Ein echtzeitfähiges System zur Erkennung der Blickrichtung des menschlichen Auges. In *Tag. Bd. Deutsche Arbeitsgemeinschaft für Mustererkennung*, 20. DAGM-Symposium, Stuttgart, 29.Sept.-01.Okt., 1997.

[Kronland-Martinet *et al.*, 1987] R. Kronland-Martinet, J. Morlet, and A. Grossmann. Analysis of sound patterns through wavelet transform. *Int. J. of Pattern Recognition and Artificial Intelligence*, 1:273–302, 1987.

[Krüger *et al.*, 1996] N. Krüger, M. Pötsch, and C. von der Malsburg. Determination of face position and pose with a leaned representation based on labelled graphs. *Image and Vision Computing*, 15:665–673, 1996.

[Krüger *et al.*, 1999] V. Krüger, R. Herpers, K. Daniilidis, and G. Sommer. Teleconferencing using an attentive camera system. In *Proc. Int. Conf. on Audio- and Video-based Biometric Person Authentication*, pages 142–147, 1999.

[Lam and Yan, 1996] K.-M. Lam and H. Yan. Locating and extracting the eye in human face images. *Pattern Recognition*, 29:771–779, 1996.

[Lee, 1996] T. S. Lee. Image representation using 2D Gabor wavelets. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 18:959–971, 1996.

[Leondes, 1966] C. Leondes, editor. *Advances in Control Systems Theory and Applications*, chapter H. Sorenson: Kalman filtering techniques, pages 219–292. Academic Press, 1966.

[Lien *et al.*, 1998] J.J. Lien, J.F. Cohn, T. Kanade, and C.-C. Li. Automated facial expression recognition based on FACS action units. In *Proc. Int. Conf. on Automatic Face and Gesture Recognition*, Nara, Japan, April 14-16, 1998.

[Loeve, 1955] M. Loeve. *Probability Theory*. Van Nostrand, Princeton, N.J., 1955.

[Louis *et al.*, 1994] A. Louis, P. Maaš, and A. Riedler. *Wavelets: Theorie und Anwendung*. Teubner, 1994.

[Lyons and Akamatsu, 1998] M. Lyons and S. Akamatsu. Coding facial expressions with Gabor wavelets. In *Proc. Int. Conf. on Automatic Face and Gesture Recognition*, Nara, Japan, April 14-16, 1998.

[Maggioni, 1995] C. Maggioni. GestureComputer – New ways of operating a computer. In *Proc. Int. Workshop on Automatic Face and Gesture Recognition*, pages 166–171, Zurich, Switzerland, June 26-28, 1995.

[Mallat, 1989a] S. Mallat. Multifrequency channel decompositions of images and wavelet models. *IEEE Trans. on Acoustic, Speech, and Signal Processing*, 37:2091–2110, 1989.

[Mallat, 1989b] S. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 11:674–693, 1989.

[Mallat, 1998] S. Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, 1998.

[Manjunath and Chellappa, 1993] B.S. Manjunath and R. Chellappa. A unified approach to boundary perception: edges, textures, and illusory contours. *IEEE Trans. Neural Networks*, 4(1):96–107, 1993.

[Marr, 1982] D. Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. Freeman, San Francisco, 1982.

[Matas *et al.*, 1999] J. Matas, K. Jonsson, and J. Kittler. Fast face localisation and verification. *Image and Vision Computing*, 17:575–581, 1999.

[Maurer and von der Malsburg, 1995] T. Maurer and C. von der Malsburg. Single-view based recognition of faces rotated in depth. In *Proc. Int. Workshop on Automatic Face and Gesture Recognition*, pages 248–253, Zurich, Switzerland, June 26-28, 1995.

[McKenna *et al.*, 1996] S. McKenna, S. Gong, and J. Collins. Face tracking and pose representation. In *Proc. British Machine Vision Conference*, Edinburgh, 1996.

[Mehrotra *et al.*, 1992] R. Mehrotra, K.R. Namuduri, and R. Ranganathan. Gabor filter-based edge detection. *Pattern Recognition*, 52:1479–1494, 1992.

[Meyer, 1992] Y. Meyer. *Wavelets and Operators*. Cambridge University Press, 1992.

[Michaelis, 1997] M. Michaelis. *Low Level Image Processing using Steerable Filters*. PhD thesis, University of Kiel, 1997. Bericht Nr. 9716.

[Moghaddam and Pentland, 1997] B. Moghaddam and A. Pentland. Probabilistic visual learning for object detection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 17:696–710, 1997.

[Moses *et al.*, 1994] Y. Moses, Y. Adini, and S. Ullman. Face recognition: The problem of compensating for changes in illumination direction. In *Proc. European Conf. on Computer Vision*, pages 286–296, Stockholm, Sweden, May 2-6, 1994.

[Murase and Nayar, 1995] H. Murase and S. Nayar. Visual learning and Recognition of 3-D objects from appearance. *Int. J. of Computer Vision*, 14:5–24, 1995.

[Nakamura *et al.*, 1991] O. Nakamura, S. Mathur, and T. Minami. Identification of human faces based on isodensity maps. *Pattern Recognition*, 24:263–272, 1991.

[Nayar and Murase, 1996] S. Nayar and H. Murase. Dimensionality of illumination in appearance matching. In *Proc. IEEE Conf. Robotics and Automation*, 1996.

[Orr, 1999] M. Orr. Introduction to radial basis function networks. Technical report, Centre for Cognitive Science, University of Edinburgh, 1999.

[Pavlovic *et al.*, 1997] V.I. Pavlovic, R. Sharma, and T.S. Huang. Visual interpretation of hand gestures for human-computer interaction: A review. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19:677–695, 1997.

[Pelc, 1997] A. Pelc. Aufgabenbezogen progressive Repräsentation von Bildern durch Waveletnetze. Master's thesis, University of Kiel, 1997.

[Petraki, 1996] Eleni Petraki. Analyse der Blickrichtung des Menschen und der Kopforientierung im Raum mittels passiver Bildanalyse. Master's thesis, Technical University of Hamburg-Harburg, 1996.

[Phillips *et al.*, 1997] P. Phillips, H. Moon, P. Rauss, and S.Rizvi. The FERET database and evaluation methodology for face recognition algorithms. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 137–143, 1997.

[Phillips *et al.*, 1998] P. Phillips, H. Wechsler, J. Huang, and P. Rauss. The FERET database and evaluation procedures for face recognition algorithms. *Image and Vision Computing*, 16:295–306, 1998.

[Poggio and Beymer, 1995] T. Poggio and D. Beymer. Learning networks for face analysis and synthesis. In *Proc. Int. Workshop on Automatic Face and Gesture Recognition*, pages 160–165, Zurich, Switzerland, June 26-28, 1995.

[Poggio and Girosi, 1990] T. Poggio and F. Girosi. Networks for approximation and learning. *Proceedings of the IEEE*, 78(9):1481–1497, 1990.

[Porat and Zeevi, 1988] M. Porat and Y. Zeevi. The generalized Garbor scheme of image representation in biological and machine vision. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 10:452–468, 1988.

[Press *et al.*, 1986] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling. *Numerical Recipes, The Art of Scientific Computing*. Cambridge University Press, Cambridge, UK, 1986.

[Reisfeld and Yeshurun, 1998] D. Reisfeld and Y. Yeshurun. Preprocessing of face images: detection of features and normalization. *Int. J. of Computer Vision*, 71:413–430, 1998.

[Reyneri, 1999] L. Reyneri. Unification of neural and wavelet networks and fuzzy fystems. *IEEE Trans. Neural Networks*, 10:801–814, 1999.

[Ritter *et al.*, 1991] H. Ritter, T. Martinez, and K. Schulten. *Neuronale Netze*. Addison-Wesley, 1991.

[Rock, 1985] I. Rock. *Wahrnehmung: vom visuellem Reiz zum Sehen und Erkennen*. Spektrum der Wissenschaft, Heidelberg, 1985.

[Rowley *et al.*, 1998] H. Rowley, S. Baluja, and T. Kanade. Rotation invariant neural network-based face detector. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 38–44, Santa Barbara, CA, June 23-25, 1998.

[Sandini and Dario, 1990] G. Sandini and P. Dario. Active vision based on a space-variant sensor. In H. Miura and S. Arimoto, editors, *Robotics Research*. MIT Press, 1990.

[Schiele and Crowley, 2000] B. Schiele and J. Crowley. Recognition without correspondence using multidimensional receptive field histograms. *Int. J. of Computer Vision*, 36:31–50, January 2000.

[Schiele and Waibel, 1995] B. Schiele and A. Waibel. Gaze tracking based on face color. In *Proc. Int. Workshop on Automatic Face and Gesture Recognition*, pages 344–349, Zurich, Switzerland, June 26-28, 1995.

[Shashua, 1992] A. Shashua. *Geometry and Photometry in 3D Visual Recognition*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, 1992.

[Sirovitch and Kirby, 1987] L. Sirovitch and M. Kirby. Low dimensional procedure for the characterization of human faces. *J. Opt. Soc. Am.*, 2:519–524, 1987.

[Sommer, 1995] G. Sommer. Verhaltensbasierter Entwurf technischer visueller Systeme. *Künstliche Intelligenz*, 3(5):42–45, 1995.

[Sommer, 1997] G. Sommer. Algebraic aspects of designing behavior based systems. In G. Sommer and J.J. Koenderink, editors, *Algebraic Frames for the Perception-Action Cycle*, pages 1–28, Int. Workshop, Kiel, Germany, September, 1997.

[Stiefelhagen *et al.*, 1997] R. Stiefelhagen, J. Yang, and A. Waibel. A model-based gaze tracking system. *International Journal of Artificial Intelligence Tools*, 6:193–209, 1997.

[Sung and Poggio, 1994] K.-K. Sung and T. Poggio. Example-based learning for view-based human face detection. Technical report, A.I. Memo No. 1521, CBCL Paper 112, MIT, Cambridge, MA, December 1994.

[Swain and Ballard, 1991] M. J. Swain and D. H. Ballard. Color indexing. *Int. J. of Computer Vision*, 7:11–32, 1991.

[Szu *et al.*, 1992] H. Szu, B. Telfer, and S. Kadambe. Neural network adaptive wavelets for signal representation and classification. *Optical Engineering*, 31:1907–1961, 1992.

[Tock and Craw, 1996] D. Tock and I. Craw. Tracking and measuring drivers eyes. *Image and Vision Computing*, 14:541–547, 1996.

[Toyama and Hager, 1996] K. Toyama and G. Hager. Incremental focus of attention for robust visual tracking. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 189–195, 1996.

[Toyama, 1997] K. Toyama. *Robust Vision-Based Object Tracking*. PhD thesis, Yale University, 1997.

[Tsotsos, 1990] J.K. Tsotsos. Analyzing vision at the complexity level. *Behavioral and Brain Sciences*, 13:423–469, 1990.

[Turk and Pentland, 1991] M. Turk and A. Pentland. Eigenfaces for recognition. *Int. Journal of Cognitive Neuroscience*, 3:71–89, 1991.

[Varchmin *et al.*, 1997] A.C. Varchmin, R. Rae, and H. Ritter. Image based recognition of gaze direction using adaptive methods. In I. Wachsmuth, editor, *Proc. Int. Gesture Workshop*, pages 245–257. Springer, 1997.

[Vetter and Blanz, 1998] T. Vetter and V. Blanz. Estimating coloured 3D face models from single images: An example based approach. In *Proc. European Conf. on Computer Vision*, pages 499–513, Freiburg, Germany, June 1-5, 1998.

[Wiskott *et al.*, 1997] L. Wiskott, J. M. Fellous, N. Krüger, and C. v. d. Malsburg. Face recognition by elastic bunch graph matching. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19:775–779, 1997.

[Xi *et al.*, 1994] X. Xi, R. Sudhakar, and H. Zhuang. On improving eye feature extraction using deformable templates. *Pattern Recognition*, 27:791–799, 1994.

[Xu and Akatsuka, 1998] M. Xu and T. Akatsuka. Detecting head pose from stereo image sequences for active face recognition. In *Proc. Int. Conf. on Automatic Face and Gesture Recognition*, pages 82–87, Nara, Japan, April 14-16, 1998.

[Yang and Huang, 1994] G. Yang and T. Huang. Human face detection in a complex background. *Pattern Recognition*, 27:53–63, 1994.

[Yow and Cipolla, 1997] K. Yow and R. Cipolla. Feature-based human face detection. *Image and Vision Computing*, 15:713–735, 1997.

[Yuille, 1991] A. L. Yuille. Deformable templates for face recognition. *J. of Cognitive Neuroscience*, 3:59–70, 1991.

[Zabrodsky and Peleg, 1990] H. Zabrodsky and S. Peleg. Attentive transmission. *J. of Visual Communication and Image Representation*, 1:189–198, 1990.

[Zeki, 1993] S. Zeki. *A Vision of the Brain*. Blackwell Scientific Publications, 1993.

[Zeng and Sommer, 1996] L. Zeng and G. Sommer. Extracting illumination invariant face representation. *Int. J. of Machine Graphics and Vision*, 5:65–76, 1996.

[Zhang and Benveniste, 1992] Q. Zhang and A. Benveniste. Wavelet networks. *IEEE Trans. Neural Networks*, 3:889–898, 1992.

[Zhang *et al.*, 1998] Z. Zhang, M. Lons, M. Schuster, and S. Akamatsu. Comparison between geometry-based and Gabor-wavelet-based facial expression recognition using multi-layer perceptron. In *Proc. Int. Conf. on Automatic Face and Gesture Recognition*, Nara, Japan, April 14-16, 1998.

[Zhao *et al.*, 1998] W. Zhao, R. Chellappa, and N. Nandhakumar. Discriminant Analysis fo Principal Components for Face Recognition. In *Nara, Japan, April 14-16*, pages 336–341, 1998.