Gabor Wavelet Networks for Efficient Head Pose Estimation

Volker Krueger [a] and Gerald Sommer [b]

[a] *Center for Automation Research, University of Maryland, A.V. Williams Building, College Park, MD 20742, USA*

[b] *Computer Science Institute, Christian-Albrechts University Kiel, Preußerstr. 1-9, 24105 Kiel, Germany*

Abstract

In this paper we first introduce the Gabor Wavelet Network (GWN) as a model-based approach for effective and efficient object representation. GWNs combine the advantages of the continuous wavelet transform with RBF networks. They have additional advantages such as invariance to some degree with respect to affine deformations. The use of Gabor filters enables the coding of geometrical and textural object features. Gabor filters as a model for local object features ensure considerable data reduction while at the same time allowing *any* desired precision of the object representation ranging from sparse to photo-realistic representation. As an application we present an approach for the estimation of head pose based on the Gabor Wavelet Networks. Feature information is encoded in the wavelet coefficients. An artificial neural network is then used to compute the head pose from the wavelet coefficients.

Key words: Gabor wavelet networks, pose estimation

1 Introduction

Recently, model-based approaches for object representation and recognition, such as the bunch graph approach, principal component analysis (PCA), eigenfaces and active appearance models, have received considerable interest [1; 2; 3; 4]. In these approaches, the term “model-based” is understood in the sense that a set of training objects is given in the form of grey value pixel images while the model “learns” the variances of the grey values (PCA, eigenfaces) or, respectively, the Gabor filter responses (bunch graph). With this, model knowledge is given by the variances of pixel grey values, which means that the actual knowledge representation is given on a pixel basis; this is a representation that is independent of the object itself.

In this paper we introduce an object representation that is based on Gabor Wavelet Networks [5] and show its advantages for the pose estimation problem. Gabor Wavelet Networks (GWNs) combine the advantages of Radial Basis Function (RBF) networks and Gabor wavelets: GWNs represent an object as a linear combination of Gabor wavelets and the parameters of each single Gabor functions (such as orientation, position and scale) are individually optimized to reflect the particular local image structure. Gabor Wavelet Networks have several advantages:

(1) GWN allow an efficient and sparse coding while coding is adaptive to the task at hand.
(2) Gabor filters are good feature detectors [6] and the optimized parameters of each of the Gabor wavelets are directly related to the underlying image structure.
(3) The wavelet coefficients (or weights) of each of the Gabor wavelets are linearly related to the filter responses and with that they are also directly related to the underlying local image structure.
(4) The precision of the representation can be varied to *any* desired degree ranging from a coarse representation to an almost photo-realistic one by simply varying the number of used wavelets.
(5) By their very nature, GWNs are invariant to affine deformations without shear and homogeneous illumination changes[7; 8].

Each single point is extensively discussed in [7] and will be addressed shortly in Section 2.

The use of Gabor filters implies a model for the actual representation of the object information. In fact, as we will see, the GWN represents object information as a set of local image features, which leads to a higher level of abstraction and to considerable data reduction.

The variability in precision and data reduction are important advantages for the pose estimation application that we discuss here. Reasons are as follows:

(1) Because the parameters of the Gabor wavelets and the weights of the network are directly related to the structure of the training image and the Gabor filter responses, a GWN can be seen as a task oriented optimized filter bank: given the number of filters, a GWN defines the set of filters that extracts the maximal possible image information.
(2) For real-time applications, one wants to keep the number of filterings low to save computational resources and it makes sense in this context to relate the number of filterings to the amount of image information really needed for a specific task: In this sense, it is possible to relate the precision in representation to the specific task and to increment the number of filters if more information is needed. This, we call *progressive attention*.
(3) The variability affects the training speed of neural networks, that correlates with the dimensionality of the input vector.

The term *progressive attention* was first used in the context of image encoding[9]. It refers to the fact that an object is considered as a collection of image features and as more information about the object is needed to fulfill a task, more features are extracted from the image.

This paper contains two parts. In the first part (Section 2), we will give a short introduction to our GWNs and their features that are relevant for the pose estimation application. We discuss each point mentioned above, including the invariance properties, the abstraction properties and specificity of the wavelet parameters and weights for object representation and task oriented image filtering.

In the second part (Section 3), we present in detail our pose estimation approach and the results of our experiments. For this, we exploit the optimality of the filter bank and the *progressive attention* property to speed up the response time of the system and to optimize the training of the neural network. Also, we discuss how pose estimation results depend on the number of filters used.

The last Section concludes with some final remarks.

2 Introduction to Gabor Wavelet Networks

Wavelet Networks were first introduced by [10], and the use of Gabor functions is inspired by the fact that they are recognized to be good feature detectors [6].

To define a GWN, we start by taking a set of N odd, real-valued Gabor wavelet functions $\mathbf{\Psi} = \{\psi_{\mathbf{n}_1}, \ldots, \psi_{\mathbf{n}_N}\}$ of the form

$$\begin{aligned}\psi_{\mathbf{n}}\Big(x, y\Big) = \exp\Bigg(-\frac{1}{2}\Big[s_x\left((x-c_x)\cos\theta - (y-c_y)\sin\theta\right)\Big]^2 \\ + \Big[s_y\left((x-c_x)\sin\theta + (y-c_y)\cos\theta\right)\Big]^2\Bigg) \\ \cdot\ \sin\Big(s_x\left((x-c_x)\cos\theta - (y-c_y)\sin\theta\right)\Big)\ , \qquad (1)\end{aligned}$$

with $\mathbf{n} = (c_x, c_y, \theta, s_x, s_y)^T$. Here, c_x, c_y denote the translation of the Gabor wavelet, s_x, s_y denote the dilation and θ the orientation. The choice of N is arbitrary and is related to the maximal representation precision of the network. The parameter vector $\mathbf{n}$ (translation, orientation and dilation) of the wavelets may be arbitrarily chosen at this point. In order to find the GWN for image I, the energy functional

$$E = \min_{\mathbf{n}_i, w_i \text{ for all } i} \|I - \sum_i w_i \psi_{\mathbf{n}_i}\|_2^2 \qquad (2)$$

is minimized with respect to the weights w_i and the wavelet parameter vector $\mathbf{n}_i$. We therefore define a Gabor Wavelet Network as follows:

Definition: Let $\psi_{\mathbf{n}_i}$, $i = 1, \ldots, N$ be a set of Gabor wavelets, and let I be a DC-free image and w_i and $\mathbf{n}_i$ chosen according to the energy functional (2). The vector of Gabor wavelets $\mathbf{\Psi} = (\psi_{\mathbf{n}_1}, \ldots, \psi_{\mathbf{n}_N})^T$ and the weight vector $\mathbf{w} = (w_1, \ldots, w_N)^T$ then define the *Gabor Wavelet Network* $(\mathbf{\Psi}, \mathbf{w})$ for image I.

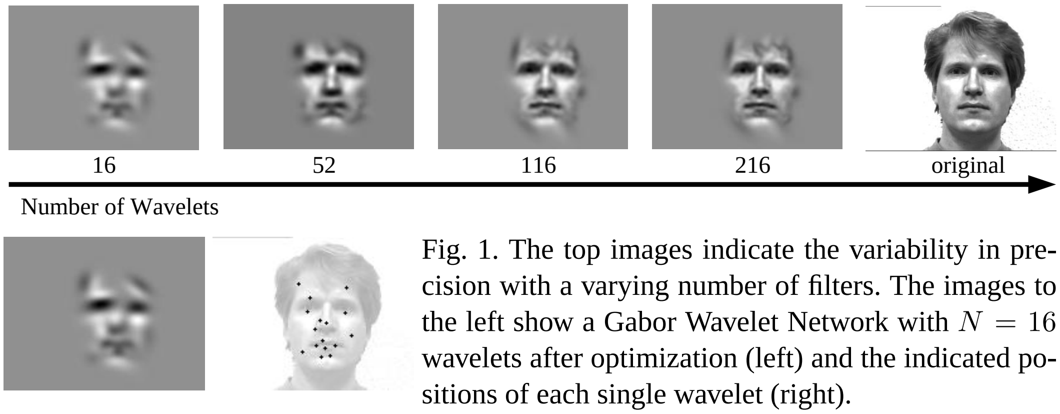

Fig. 1. The top images indicate the variability in precision with a varying number of filters. The images to the left show a Gabor Wavelet Network with $N = 16$ wavelets after optimization (left) and the indicated positions of each single wavelet (right).

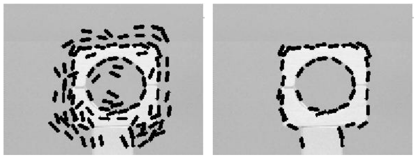

Fig. 2. The figure shows images of a wooden toy block on which a GWN was trained. The black line segments sketch the positions, sizes and orientations of all the wavelets of the GWN (left), and of some automatically selected wavelets (right).

The parameter vectors $\mathbf{n}_i$ are chosen from *continuous* phase space $\mathbb{R}^5$ [8] and the Gabor wavelets, being continuous functions, are positioned with sub-pixel accuracy. This is precisely the main advantage over the discrete approach [8; 11]. 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 selectively chosen in the continuous case to reflect *precisely* the local image structure. This assures that a maximum of the image information is encoded.

Using the optimized wavelets $\mathbf{\Psi}$ and weights $\mathbf{w}$ of the Gabor wavelet network of an image I, I can be (closely) reconstructed by a linear combination of the weighted wavelets:

$$\hat{I} = \sum_{i=1}^{N} w_i \psi_{\mathbf{n}_i} = \mathbf{\Psi}^T \mathbf{w} \ . \tag{3}$$

Of course, the quality of the image representation and reconstruction depends on the number N of wavelets used and can be varied to reach almost any desired precision. An example is in Fig. 1, top, where reconstructions $\hat{I}$ with variable $N = 16, 52, 116, 216$ wavelets are shown. An additional example is in Fig. 1, bottom row. The image to the left shows a reconstruction with 16 wavelets and the right image indicates the corresponding wavelet positions. It should be pointed out that at each indicated wavelet position, just *one* single wavelet is located.

2.1 Feature Representation with Gabor Wavelets

It was mentioned in the introduction that the Gabor wavelets are recognized to be good feature [6] detectors, that are directly related to the local image features by eq. (2). This means that an optimized wavelet has e.g. ideally the exact position and orientation of a local image feature. An example is given in Fig. 2. The figure shows the image of a little wooden toy block, on which a Gabor Wavelet Network was trained. The left image shows the positions, scales and orientations of the wavelets as little black line segments. By thresholding the weights, the more "important" wavelets may be selected, which leads to the right image. Ideally, each Gabor wavelet should be positioned *exactly* on the image line after optimization[7]. Furthermore, since a large weight indicates that the corresponding wavelet represents an edge segment (see Section 2.2), the wavelets encode local geometrical object information. In reality, however, interactions with other wavelets of the network have to be considered so that most wavelet parameters reflect the position, scale, and orientation of the image line closely, but not precisely. This fact is clearly visible in Fig. 2, right.

As it can be seen in Fig. 1 an object can be represented very well with a relatively small set of wavelets. This considerable data reduction is achieved by the introduction of the model for local image primitives, i.e. the Gabor wavelets.

2.2 Direct Calculation of Weights

Gabor wavelet functions are not orthogonal. For a given family $\mathbf{\Psi}$ of Gabor wavelets it is therefore not possible to calculate a weight w_i by a simple projection of the Gabor wavelet $\psi_{\mathbf{n}_i}$ onto the image (as it is being done for orthogonal wavelets). Instead one has to consider the family of *dual* wavelets $\tilde{\mathbf{\Psi}} = \{\tilde{\psi}_{\mathbf{n}_1} \ldots \tilde{\psi}_{\mathbf{n}_N}\}$. The wavelet $\tilde{\psi}_{\mathbf{n}_j}$ is the *dual* wavelet to the wavelet $\psi_{\mathbf{n}_i}$ iff

$$\langle \psi_{\mathbf{n}_i}, \tilde{\psi}_{\mathbf{n}_j} \rangle = \int \psi_i(x) \tilde{\psi}_k(x) dx = \delta_{i,j} = \begin{cases} 1 \text{ if } i = j \\ 0 \text{ if } i \neq k \end{cases} . \tag{4}$$

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

$$\mathbf{\Psi}^T \tilde{\mathbf{\Psi}} = \left(\langle \psi_{\mathbf{n}_i}, \tilde{\psi}_{\mathbf{n}_j} \rangle \right)_{i,j} = \mathbb{1} \; . \tag{5}$$

In other words the *dual* wavelets compensate for the non-orthogonality of the Gabor wavelets; the wavelet coefficients (weights) can now be computed from the image I by the projection of their *dual* wavelets onto I:

$$w_i = \langle I, \tilde{\psi}_{\mathbf{n}_i} \rangle \; . \tag{6}$$

We find $\tilde{\psi}_{\mathbf{n}_i}$ to be

$$\tilde{\psi}_{\mathbf{n}_i} = \sum_j \left(\Psi_{i,j} \right)^{-1} \psi_{\mathbf{n}_j} \text{ , where } \Psi_{i,j} = \langle \psi_{\mathbf{n}_i}, \psi_{\mathbf{n}_j} \rangle \; . \tag{7}$$

See Appendix A for a proof. Given a vector, $\mathbf{\Psi}$, of optimized wavelets of a GWN, the dual vector, $\tilde{\mathbf{\Psi}}$, therefore allows an orthogonal projection of an image J onto

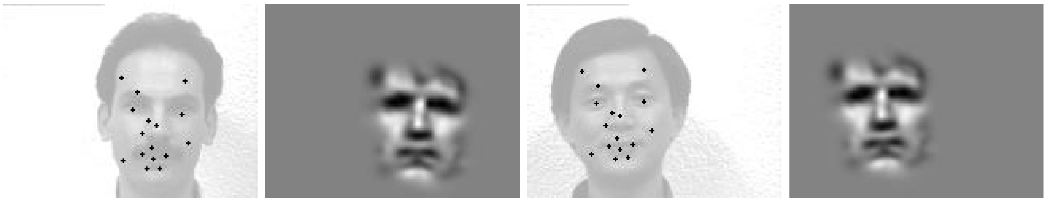

Fig. 3. The images show the positions of each of the 16 wavelets after reparameterizing the wavelet net and the corresponding reconstruction. The reconstructed faces show the same orientation, position and size as the ones they were reparameterized on.

the closed linear span of $\mathbf{\Psi}$, i.e.

$$\hat{J} = \left(J\tilde{\mathbf{\Psi}}\right)\mathbf{\Psi} = \sum_{i=1}^{N} w_i \psi_{\mathbf{n}_i} \text{ , with } \mathbf{w} = J\tilde{\mathbf{\Psi}} \tag{8}$$

2.3 Reparameterization of Gabor Wavelet Networks

The task of finding the optimal position, scale and the orientation of a given GWN for a new image is important. Here, PCA, bunch graphs and GWN have similar properties: In the case of PCA and bunch graph representations, it is important to ensure that corresponding pixels are aligned into a common coordinate system, while in case of the GWN, local image primitives are aligned. Given a corresponding GWN, we are interested in finding the correct position, orientation and scaling of the GWN so that the wavelets are positioned as precisely as possible on the same facial features as in the original image. For this we assume the GWN to be rigid in the sense that the small Gabor wavelets are not allowed to move relative to each other; the entire GWN may be deformed geometrically so that it is aligned with the coordinate system of the object in the new image. An example for a successful deformation can be seen in Fig. 1, where in the bottom right image the wavelet positions of the *original* wavelet network are marked and in Fig. 3, left, where in new images the wavelet positions of the same GNW are marked, but deformed and *reparameterized* accordingly to fit the new faces. The right images in Fig. 3 show the reconstruction of the deformed GWN of Fig. 1, now showing the same position, rotation and scale of the new faces (left).

To formalize the idea, the reparameterization of a GWN is established by using a *superwavelet* [12]:

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

$$\mathbf{\Psi_n}(\mathbf{x}) = \sum_i w_i \psi_{\mathbf{n}_i}(\mathbf{SR}(\mathbf{x} - \mathbf{c})), \tag{9}$$

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

A superwavelet $\mathbf{\Psi_n}$ is again a wavelet that has the same wavelet parameter set (dilation, translation and rotation) as the Gabor wavelets used above. This means that the parameter vector $\mathbf{n}$ affects the superwavelet $\mathbf{\Psi}$ as a whole and the small wavelets ψ are not affected invididually. This also means that we can also handle and optimize the superwavelet in the same way as we did it with each of the small Gabor wavelets in eq.(2): For a new image J we deform the superwavelet by optimizing its parameters $\mathbf{n}$ with respect to the energy functional E:

$$E = \min_{\mathbf{n}} \|J - \mathbf{\Psi_n}\|_2^2 \tag{10}$$

The above functional allows us to define the operator

$$\begin{aligned} \mathcal{P}_{\mathbf{\Psi}} : \mathbb{L}^2(\mathbb{R}^2) &\longmapsto \mathbb{R}^5 \\ J &\longrightarrow \mathbf{n} = (c_x, c_y, \theta, s_x, s_y) \ , \end{aligned} \tag{11}$$

where $\mathbf{n}$ minimizes the energy functional E of the above equation. In eq. (11) $\mathbf{\Psi}$ is defined to be a superwavelet. The reparameterization works quite robust and has also been successfully applied for a wavelet based affine real-time face tracking [13]. See [7] for a thorough discussion.

2.4 Related Work

There are other models for image interpretation and object representation. Most of them are based on PCA [14], such as the eigenface approach [2]. The eigenface approach has shown its advantages expecially in the context of face recognition. Its major drawbacks are its sensitivity to perspective deformations and to illumination changes. PCA encodes textural information only, while geometrical information is discarded. Furthermore, the alignment of face images into a common coordinate system is still a problem.

Another PCA based approach is the active appearance model (AAM)[3]. This approach enhances the eigenface approach considerably by including geometrical information. This allows an alignment of image data into a common coordinate system while the formulation of the alignment technique can be elegantly done with techniques of the AAM framework. Also, recognition and tracking applications are presented within this framework [4]. An advantage of this approach was demonstrated in [3]: the authors showed the ability of the AAM to model, in a photo-realistic way, almost any face gesture and gender.

The bunch graph approach [1] is based, on the other hand, on the discrete wavelet transform. A set of Gabor wavelets are 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*. An object is then represented by a set of jets, that encode each a single local texture patch of the object. The jet topology, the so-called an *image graph*, encodes geometrical object information. A precise positioning of the image graph onto the test image is important for good matching results and the positioning is quite a slow process. The feature detection capabilities of the Gabor filters are not exploited since their parameters are fixed and an adaption to different precision levels has not been considered so far.

3 Pose Estimation with GWN

In this section, we present our approach for the estimation of the pose of a head using Gabor Wavelet Networks. There exist many different approaches for pose estimation, including pose estimation with color blobs [15; 16], pose estimation using a geometrical approach [17], stereo information [18] or neural networks [19], to cite just a few. While in some approaches, such as in [16], only an approximate pose is estimated, other approaches have the goal to be very precise so that they could even be used as a basis for gaze detection such as in [20]. The precision of the geometrical approach [17] was extensively tested and verified in [21]. The minimal mean pan/tilt error that was reached was $> 1.6°$. In comparison to this, the neural network approach in [19] resulted in a minimal pan/tilt error of $> 0.64°$.

The good results in [19] were achieved by first detecting the head using a color tracking approach. Within this region of interest, 16 sets of 4 complex Gabor filters with different orientations of 0, $\frac{\pi}{4}$, $\frac{\pi}{2}$ and $\frac{3}{4}\pi$ were evenly distributed on a 4×4 grid. The 128 filter responses of these 64 complex Gabor filters were then fed into a neural network similar to LLM [22] which computed the values for pan and tilt. Table 1 presents a summary of the experimental results in [19]. The error measure used is given as the Euclidean distance between the 2D groundtrouth vector containing pan and tilt, and the 2D vector that contains the two computed values.

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

Table 1
This table summarizes the experimental results for the pose estimation technique according to

Consider Fig. 4: It shows the same person as Fig. 1, but represented by homogeneously distributed Gabor filters, instead of optimized ones. The very right image in Fig. 4 is represented by 512 homogeneously distributed Gabor wavelets. While the approach in [19] uses such a homogeneous scheme for filtering the input images, it is reasonable to assume that a proper context based choice of the Gabor filters would lead to yet better pose estimation results. In our experiments we therefore trained a GWN on an image I showing a doll's head. The wavelets have been constrained to be located within the inner face region to prevent distraction from the

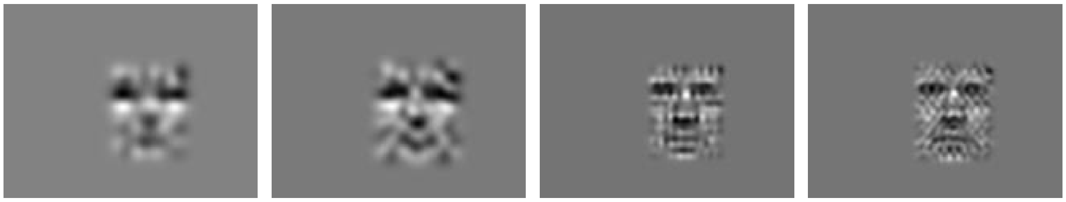

Fig. 4. These images show, qualitatively, what image information is contained in a set of Gabor filter responses, when filtering is done with (from left, top to right, bottom) 4×4 homogeneously distributed Gabor filters with 4 and 8 orientations, or with 8×8 homogeneously distributed filters with 4 and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

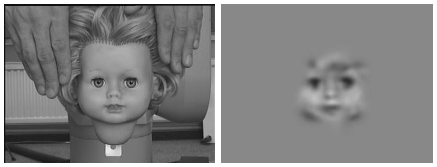

Fig. 5. The left image shows the original doll face image I, the right image shows its reconstruction $\hat{I}_{52}$ using the reconstruction formula with an optimized wavelet net $\mathbf{\Psi}$ of just $N = 52$ odd Gabor wavelets, distributed over the inner face region. For optimization, the scheme that was introduced in Section 2 was applied.

background. Fig. 5 shows the trained GWN, in Fig. 6 the white boxes refer to the inner face region considered by the GWN. For training the GWN we used the optimization scheme presented in Section 2 with $N = 52$ Gabor wavelets. As we show in the next Sections, pose estimation results improve considerably by replacing the homogeneous scheme of [19] by a GWN.

3.1 Experimental Setup

In order to be comparable with the approach in [19] we used in our experiments the same neural network and the same number of training examples as described in [19]. In [19] a subspace variant of the Local Linear Map (LLM) [22] was used for learning input - output mappings [23]. There, the LLM rests on a locally linear (first order) approximation of the unknown function $f : \mathbb{R}^n \mapsto \mathbb{R}^k$ and computes its output as (winner-take-all-variant) $y(\mathbf{x}) = \mathbf{A}_{\text{bmu}}(\mathbf{x} - \mathbf{c}_{\text{bmu}}) + \mathbf{o}_{\text{bmu}}$. Here, $\mathbf{o}_{\text{bmu}} \in \mathbb{R}^k$ is an output vector attached to the best matching unit (zero order approximation) and $\mathbf{A}_{\text{bmu}} \in \mathbb{R}^{k \times n}$ is a local estimate of the Jacobian matrix (first oder term). Centres are distributed by a clustering algorithm. Due to the first order term, the method is very sensitive to noise in the input. With a noisy version $\mathbf{x}' = \mathbf{x} + \boldsymbol{\eta}$ the output differs by $\mathbf{A}_{\text{bmu}}\boldsymbol{\eta}$, and the LLM largely benefits from projecting to the local subspace, canceling the noise component of $\boldsymbol{\eta}$ orthogonal to the input manifold. As basis functions, normalized Gaussians were used.

The doll's head was connected to a robot arm, so that the pan/tilt ground truth was known. During the training and testing, the doll's head was first tracked using our wavelet based face tracker [13] for a proper positioning of the GWN. For each frame we proceeded in two steps:

(1) optimal reparameterization of the GWN (tracking) by using the functional (10)
(2) calculating the optimal weights for the optimally repositioned GWN with eqs.

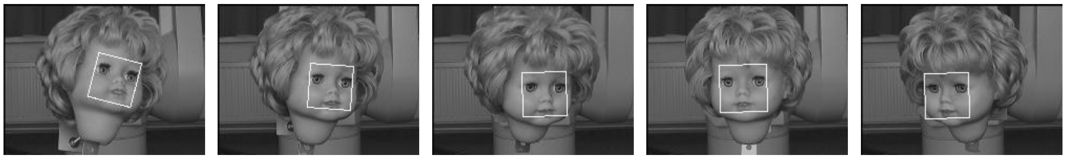

Fig. 6. The figure show different example images of the doll’s head as captured by the camera. The head is connected to a robot arm so that the ground truth is known. The white square indicates how the GWN (superwavelet) was reparameterized during the tracking step.

(7) and (8).

(3) Feeding the optimal weights into an ANN to compute estimations for pan and tilt.

See Fig. 6 for example images for steps one and two. The training was done as it was described in [19]: We used 400 training images, evenly distributed within the range of $\pm 20^\circ$ in pan and tilt direction (this is the range where all face features appeared to be visible).

3.2 Experimental Results

With the experimental setup described above, we reached a mean pan/tilt error of 0.21° for a GWN with 52 real-valued wavelets and a minimal mean pan/tilt error of 0.30° for a GWN with 16 real-valued wavelets. The maximal errors were 0.65° for 52 wavelets and 0.72° for 16 wavelets, respectively (see Tab. 2 for a summary).

Since the filter responses and weights are linearly related by eq. (7), one might hope that using the filter responses directly might lead to similar results. In fact, the observed mean pan/tilt error was found to be 0.37° for 16 wavelets and 0.23° for 52 wavelets. The maximal errors were 0.91° and 0.53°, respectively. Using filter responses instead of weights simplifies computation considerably: While Gabor Wavelets are non-orthogonal wavelets, the weights are all correlated (see eq. (7)), so that for each number N of wavelets a separate neural network has to be trained. Using the filter responses, only a single neural network needs to be trained.

	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 2
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.

The reported results are averaged over several repetitions of our experiment, the variance of the reported mean errors were $\approx 1^\circ$.

The experiments were carried out on an experimental setup, which thus far only allows off-line computation. On a 450 MHz Linux Pentium an on-line system should reach a speed of $>\approx 5$ fps for the 52 wavelet network and $>\approx 10$ fps for the 16 wavelet network [1]. It was shown in [7] that the computation time increases linearly with increasing number of used Gabor wavelets.

In comparison, for the *gaze* detection in [20], 625 training images were used, with a 14-D input vector, to train an LLM-network. The user was advised to fixate a 5×5 grid on the computer screen. The minimal errors after training for pan and tilt were $1.5°$ and $2.5°$, respectively, while the system speed was 1 Hz on a SGI (Indigo, High Impact).

3.3 Progressive Attention Scheme for Pose Estimation

We have discussed above that the Gabor Wavelet Networks allow a variability in precision by changing the number N of used Gabor wavelets. This property was called *Progressive Attention*.

To use the progressive attention scheme for the pose estimation the wavelets are sorted, with respect to their weights, in decreasing order. Therefore, wavelets with a large weight are considered to be more important, which is in accordance with the wavelet theory [8]. The weights that are considered here are those of the vector $\mathbf{w}$ given by the respective Gabor Wavelet Networks $(\mathbf{\Psi}, \mathbf{w})$. We have then evaluated the above experiments for a varying wavelet number N. The graphs in fig. 7 show how the mean and maximal pan/tilt error changed with an increasing number of Gabor wavelets. Fig. 7, left, shows the results where the estimation is based on the weights, Fig. 7, right, shows the estimation results based on the filter responses, respectively. The graphs indicate that the precision of the pose estimation in a certain range correlates with the number of wavelets used, which is in accordance with the progressive attention scheme. Above a certain number of wavelets the precision of the estimated pose is nearly independent of the number of wavelets.

4 Conclusions

The contributions of this article is twofold: First, we introduced the concepts of the *Gabor Wavelet Network* and the *Gabor superwavelet* that enable data reduction and the *progressive attention* approach. In the second Section we discussed these various properties in detail. In [24; 13], GWNs have already been used successfully for wavelet based affine real time face tracking and pose invariant face recognition. Further, we exploited all the advantages of the GWN for the estimation of the head pose. The experimental results show quite impressively that it is sensible for an object representation to reflect the specific individual properties of the object rather

[1] This is a conservative estimation, various optimizations should allow higher frame rates.

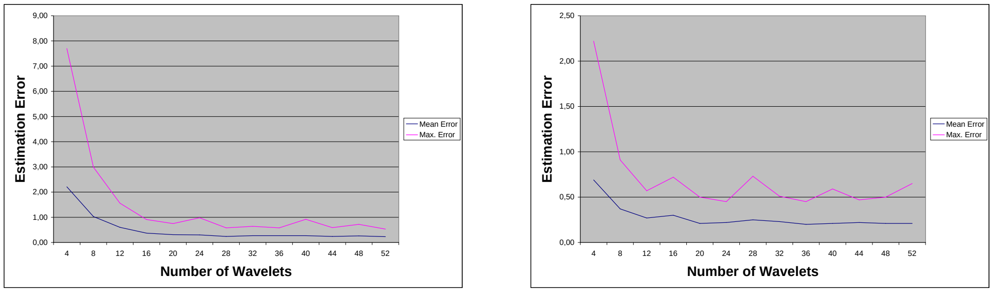

Fig. 7. The left figure shows the decrease in pose estimation error with an increasing number of wavelets. For these plots, weights were computed with eq. (8), which were fed into the ANN. Shown are the plots for the mean error and the maximal error (in degrees). The right figure shows the decrease in pose estimation error with an increasing number of wavelets. For these plots, the filter responses were directly fed into the neural network. Shown are the plots for the mean error and the maximal error (in degrees).

than being independent of the individual properties such as general representations are. This can especially be seen when comparing the presented approach with the one in [19]: While having used the same experimental setup and the same type of neural network, the precision of the presented approach is twice as good with only 16 coefficients (vs. 128 coeffs.), and three times as good with only 52 coeffs. Furthermore, the experiments show, how the precision in pose estimation and the system speed change with an increasing number of filters. A controllable variability of precision and speed has a major advantage: The system is able to decide how precise the estimation should be in order to minimize the probability that the given task is not fulfilled satisfactorily. In the future we are planning to incorporate the experimental setup into an on-line system. An enhancement for the evaluation of the positions of the irises for a precise estimation of gaze will also be tested.

Acknowledgment

The images used are derived from the Yale Face Database. We thank Sven Bruns for his help with the experiments.

A Appendix

In order to show that the $\tilde{\psi}_{\mathbf{n}_i}$ in eq. (7) are indeed dual to be $\psi_{\mathbf{n}_i}$, we have to verify the bi-orthogonality condition (4):

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

In the second to last row, the i-th column of matrix $(\mathbf{\Psi}_{i,j})$ is multiplied by the k-th row of its inverse, which evaluates to 1 if $i = k$, and to 0 otherwise. Equation (7) is not specific to Gabor wavelets, as one can see in the proof, but holds for *any* function family of finite dimensionality.

References

[1] L. Wiskott, J. M. Fellous, N. Krüger, C. v. d. Malsburg, Face recognition by elastic bunch graph matching, IEEE Trans. Pattern Analysis and Machine Intelligence 19 (1997) 775–779.

[2] M. Turk, A. Pentland, Eigenfaces for recognition, Int. Journal of Cognitive Neuroscience 3 (1991) 71–89.

[3] T. Cootes, G. Edwards, C. Taylor, Active appearance models, in: Proc. European Conf. on Computer Vision, Vol. 2, Freiburg, Germany, June 1-5, 1998, pp. 484–498.

[4] G. Edwards, T. Cootes, C. Taylor, Face recognition using active appearance models, in: Proc. European Conf. on Computer Vision, Vol. 2, Freiburg, Germany, June 1-5, 1998, pp. 581–595.

[5] V. Krüger, G. Sommer, Gabor wavelet networks for object representation, Tech. Rep. 2002, Institute of Computer Science, University of Kiel (2000).

[6] B. Manjunath, R. Chellappa, A unified approach to boundary perception: edges, textures, and illusory contours, IEEE Trans. Neural Networks 4 (1) (1993) 96–107.

[7] V. Krüger, Gabor wavelet networks for object representation, Tech. Rep. CS-TR-4245, University of Maryland, CFAR (May 2001).

[8] I. Daubechies, The wavelet transform, time-frequency localization and signal analysis, IEEE Trans. Information Theory 36 (1990) 961–1005.

[9] H. Zabrodsky, S. Peleg, Attentive transmission, J. of Visual Communication and Image Representation 1 (1990) 189–198.

[10] Q. Zhang, A. Benveniste, Wavelet networks, IEEE Trans. Neural Networks 3 (1992) 889–898.

[11] T. S. Lee, Image representation using 2D Gabor wavelets, IEEE Trans. Pattern Analysis and Machine Intelligence 18 (1996) 959–971.

[12] H. Szu, B. Telfer, S. Kadambe, Neural network adaptive wavelets for signal representation and classification, Optical Engineering 31 (1992) 1907–1961.

[13] V. Krüger, G. Sommer, Affine real-time face tracking using gabor wavelet networks, in: Proc. Int. Conf. on Pattern Recognition, IEEE Computer Society, Barcelona, Spain, Sept. 3-8, 2000.

[14] I. Jolliffe, Principal Component Analysis, Springer Verlag, New York, 1986.

[15] Q. Chen, H. Wu, T. Fukumoto, M. Yachida, 3d head pose estimation without feature tracking, in: Proc. Int. Conf. on Automatic Face and Gesture Recognition, Nara, Japan, April 14-16, 1998, pp. 88–93.

[16] B. Schiele, A. Waibel, Gaze tracking based on face color, in: Proc. Int. Workshop on Automatic Face and Gesture Recognition, Zurich, Switzerland, June 26-28, 1995, pp. 344–349.

[17] A. Gee, R. Cipolla, Determining the gaze of faces in images, Image and Vision Computing 12 (1994) 639–647.

[18] M. Xu, T. Akatsuka, Detecting head pose from stereo image sequences for active face recognition, in: Proc. Int. Conf. on Automatic Face and Gesture Recognition, Nara, Japan, April 14-16, 1998, pp. 82–87.

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

[20] A. Varchmin, R. Rae, H. Ritter, Image based recognition of gaze direction using adaptive methods, in: I. Wachsmuth (Ed.), Proc. Int. Gesture Workshop, Springer, 1997, pp. 245–257.

[21] E. Petraki, Analyse der blickrichtung des menschen und der kopforientierung im raum mittels passiver bildanalyse, Master's thesis, Technical University of Hamburg-Harburg (1996).

[22] H. Ritter, T. Martinez, K. Schulten, Neuronale Netze, Addison-Wesley, 1991.

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

[24] V. Krüger, G. Sommer, Gabor wavelet networks for object representation, in: Proc. of the Int. Dagstuhl 2000 Workshop, LNCS, Springer, 2000, to be published.