Visualization of Multidimensional Shape and Texture Features in Laser Range Data using Complex–Valued Gabor Wavelets

M. H. Gross, Associate Member IEEE
R. Koch
Institute for Information Systems
Computer Science Department
Swiss Federal Institute of Technology
ETH–Zentrum
8092 Zürich
Tel: +41 1 632 7114
Fax: +41 1 632 1172
E–mail: grossm@inf.ethz.ch

Index Terms: Feature–based visualization, feature extraction, wavelet transform, Gabor wavelets, multidimensional visualization, subspace mapping, principal components, shape and texture analysis.

Abstract

This paper describes a new method for visualization and analysis of multivariate laser range data using complex–valued non–orthogonal Gabor wavelets, principal component analysis and a topological mapping network. The initial data set that provides both shape and texture information is encoded in terms of both amplitude and phase of a complex valued 2D image function. A set of carefully designed oriented Gabor filters performs a decomposition of the data and allows for retrieving local shape and texture features. The feature vector obtained from this method is multidimensional and in order to evaluate similar data features, further subspace methods to transform the data onto visualizable attributes, such as R,G,B, have to be determined. For this purpose, a feature–based visualization pipeline is proposed consisting of principal component analysis, normalization and a topological mapping network. This process finally renders a R,G,B subspace representation of the multidimensional feature vector. Our method is primarily applied to the visual analysis of features in human faces – but is not restricted to that.

1 Introduction

One of the major challenges of vision research was, and still is, to develop methods for the automatic modeling of complex geometric objects or scenes. In spite of countless efforts during the last decades [2], [42], [32], [36] there is not yet a generic solution to this problem. Electronic photogrammetry, however, has invented active vision methods, like laser range finders [40], that are widely used to yield complex surface shape information elegantly and efficiently. Once only available in military applications for robust terrain following navigation systems, different types of laser scanners nowadays serve as helpful tools for automatic modeling tasks in many different areas [57]. Recent advancements in the development of laser scanners allow the capture of shape and color information of the object in question [59]. Hence, sophisticated methods to encode and analyze range image maps have become essential requirements [5], [62].

Human faces in particular are attractive objects for laser range scanners [53], [45], since they contain complex geometric features and intrinsic symmetries on the one hand [33], and are very appropriate for human visual analysis on the other hand. Accurate geometric data from human faces can be used to enhance the information in simple photographs and allow for the development of higher quality techniques for facial surgery, facial recognition, facial reconstruction, simulation of aging, etc. The question arises how to encode this shape and texture data into representations that visualize important data features.

A lot of research has been done to find appropriate geometric model descriptions for regular [18] and irregular [19], [47] surface data and nonuniform rational B–splines (NURBS) have been shown to be a flexible scheme for controlling complex shapes. Yet, it turns out to be very difficult to obtain robust features from such descriptions. Moreover, when considering range data, there needs to be a way to also encode and analyze texture. Apart from that, wavelets and multiresolution analysis, as proposed by [14], [29] or [41] have become very attractive for many applications. In computer graphics mostly orthonormal wavelets and their separable 2D and 3D extensions have been used for hierarchical data decomposition and approximation. Taking advantage of the compact coding scheme and of the local support of the basis functions in [44], [26] and [60], different approaches for volume rendering and isosurface reconstruction have been suggested. [51], for instance, used wavelets for fast radiosity computations and [34] for control of 3D morphing.

Especially in image processing, the power of wavelets has been investigated for feature extraction and analysis [7], [11], [55]. While in [11] orthonormal wavelets are stressed, [15] employes nonorthogonal and nonseparable 2D–Gabor wavelets for image analysis. Other interesting research is reported by [8], who combines graph matching with Gabor functions for face recognition. It is evident that in most cases the requirement for compact support and orthonormality along with a smooth wavelet shape and a dense spatial orientation staggering for high performance in both analysis and coding presents a problem. In those cases, where the analysis properties of the basis function are superior to the approximation and coding behavior, non–orthogonal wavelets have to be considered as well.

This paper addresses two fundamental areas of scientific visualization: First, it describes how to extract multidimensional features from complex data sets, such as laser range images, using a complex–valued decomposition scheme with Gabor wavelets. Secondly, it provides a generic scheme to visualize these features in orthogonal subspaces. For this purpose we consider shape and texture as amplitude and phase of a complex–valued 2D image function and perform a hierarchical decomposition with a carefully designed set of Gabor wavelets. Although there is no straightforward way to perfectly reconstruct the initial data set from the filter pyramid, the Gabor wavelet has been shown to be a much more powerful feature descriptor than an orthonormal wavelet [25]. Specifically in contrast to the non–separable 2D Gabor filters, the tensor product extensions of orthogonal or biorthogonal wavelets [55] accomplish compact coding, but their directional selectivity is much poorer with regard to the diagonal components. Moreover, Gabor wavelets meet uniquely the lower bound of space–frequency resolution as it is stated by the Heisenberg principle. Hence, Gabor wavelets provide advantages for any type of feature analysis, that bases on local spectral estimates, such as the one we introduce in this paper. The complex–valued interpretation of the image function allows to encode the range and texture information elegantly and to compute the Gabor decomposition efficiently via FFT methods. In particular, convolutions with complex–valued Gabor functions turn out to become simple multiplications with Gaussians in Fourier space.

However, once the decomposition of the data is computed, the result is a multidimensional feature vector at each surface point and the problem arises how to inspect it visually. To this end,

subspace projections have to be applied in terms of a principal component analysis of the wavelet features. After normalization we use a modified topological mapping neural network, as proposed in [23] or [28] to accomplish cluster analysis and mapping onto the R,G,B color space. The generic feature–based visualization pipeline, introduced here, exemplifies to some degree the demand for complex multidimensional feature–based visualization techniques, as stated in [50].

The organization of the paper is as follows: First of all we briefly elaborate the mathematical basics of the Gabor function and its relationship to non–orthogonal wavelet decompositions. This section also elucidates how to build 2D Gabor wavelets in the frequency domain. Section 3 sheds light on our encoding scheme for laser range data sets and illustrates how to get local data features from this transform. Chapter 4 addresses the feature based visualization pipeline we employ with special emphasis on principal component analysis and topological mapping networks. Finally, in chapter 5 we present results on the performance of our method applied to range data of human faces. R,G,B representations of multidimensional face features are depicted as derived from the Gabor wavelets and how this method generalizes onto different data sets is outlined.

2 Mathematical Foundations

2.1 The Gabor Function

The Gabor function [21] used to decompose the range data in our approach is of fundamental importance in signal processing [1] and has been widely used in a broad range of applications [7], [15]. It provides an effective way to analyze images and has been elaborated as a framework for understanding the orientation and spatial frequency selective properties of simple cortical neurons [16]. The use of Gabor functions for data analysis therefore has its foundations both in signal analysis and in biology. Gabor functions of different frequency range and orientation can provide a non–orthogonal function basis for any finite energy function $f \in \boldsymbol{L}^2\boldsymbol{R}^n$ [3] and one way to rapidly obtain data features is to compute a Gabor transform as an expansion of f. The 2D version of the complex–valued Gabor function can be expressed as below. Its basis consists of a Gaussian envelope of a harmonic oscillation term. Important inherent advantages are infinite smoothness and exponential decay in frequency, but it has to be stated again that there is no straightforward way to reconstruct the data from the expansion basis. The Gabor function can

be defined in both the spatial and frequency domain, although the latter one turns out to be easier. In Cartesian coordinates, we obtain $\psi^G(x,y)$:

$$\psi^G(x,y) = e^{-\pi\left(\frac{(x-x_0)^2}{\sigma_x{}^2}+\frac{(y-y_0)^2}{\sigma_y{}^2}\right)} e^{-2\pi j(u_0(x-x_0)+v_0(y-y_0))} \quad (1)$$

(x_0, y_0) stand for the translation of its origin and (u_0, v_0) specify the modulation coordinates in the frequency domain. The effective width and length are given by (σ_x, σ_y) that specify the elliptic envelope.

Gabor functions are localized in both space and in frequency and they moreover uniquely achieve the theoretical lower bound of uncertainty as it is dictated by the Heisenberg relation:

$$\Delta_{xy}\,\Delta_{uv} \geqq \frac{1}{4\pi} \quad (2)$$

The resolutions in space Δ_{xy} and in frequency Δ_{uv} are computed by the 2nd order moments and correspond to the variances and covariances of the Gabor functions.

$$\Delta_{xy} = \sqrt{\frac{1}{\|\psi^G(x,y)\|^2}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(y-\bar{y})^2\,(x-\bar{x})^2\,\left|\psi^G(x,y)\right|^2 dx\,dy} \quad (3)$$

$$\Delta_{uv} = \sqrt{\frac{1}{\|G(u,v)\|^2}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(v-\bar{v})^2\,(u-\bar{u})^2\,\left|G(u,v)\right|^2 du\,dv} \quad (4)$$

$G(u,v)$: Fourier transform of $\psi^G(x,y)$

$\bar{x}, \bar{y}, \bar{u}, \bar{v}$: first order moments (centers)

Since eq. 1 does not yet account for spatial orientation, frequency ω_0 and orientation Θ_0 of the Gabor function are defined by

$$\omega_0 = \sqrt{{u_0}^2 + {v_0}^2} \ , \ \Theta_0 = \arctan\left(\frac{v_0}{u_0}\right) \tag{5}$$

The coordinates x and y can be rotated by replacing them as

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos\Theta_0 & \sin\Theta_0 \\ -\sin\Theta_0 & \cos\Theta_0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \tag{6}$$

respectively. Note that rotation leads to cross terms xy and to a nonseparable expression.

The complex oscillation term represents the orthogonal quadrature components *sine* and *cosine*. Fig. 1 illustrates the real (symmetric) and imaginary (antisymmetric) component of the Gabor function from eq. 1. The shape of both functions strongly resembles an oriented wavelet function.

Fig. 1. a) Real (cosine) and
b) imaginary (sine) component of the complex valued 2D Gabor function in the spatial domain for $u_1 = 0.18$, $v_1 = 0.07$, $\omega_1 = 0.2$, $x_0 = y_0 = 0$, $\sigma_u = \sigma_v = 0.07$, $a = 2$.

The effective number of oscillations under the Gaussian envelope can be expressed as ω_0/σ_0.

Fortunately, the Fourier transform $G(u,v)$ of $\Psi^G(x, y)$ has exactly the same functional form with inverted or interchanged parameters. For functions located at the origin it reduces to a simple Gaussian in the frequency plane. Introducing the spatial frequencies (u, v) we obtain the following expression:

$$G(u, v) = e^{-\pi\left(\frac{(u-u_0)^2}{{\sigma_u}^2} + \frac{(v-v_0)^2}{{\sigma_v}^2}\right)} \ e^{-2\pi j\left(x_0(u-u_0) + y_0(v-v_0)\right)} \tag{7}$$

with $\sigma_u = 1/\sigma_x$ and $\sigma_v = 1/\sigma_y$.

This relationship which is depicted in fig. 2, shows the Fourier transforms of the sine and of the cosine wavelet respectively. Due to the symmetry and antisymmetry of the harmonics the negative Gaussian peaks cancel each other and the remaining function reduces to one Gaussian con-

trolling the wavelet's position, size and orientation. Evidently, implementations of Gabor filters can be accomplished easily employing eq. 7.

Fig. 2. a) Fourier transform of the Gabor function of fig. 1.
It's decomposition into b) *sine* and c) *cosine* components.

From there, it is straightforward to analyze data using Gabor functions. The so–called Gabor–transform [1] as it is often stressed in the signal processing literature turns out to be a short time Fourier Transform (STFT) using a basis of Gabor functions. In the 2D–case, we get

$$GT = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y)\ w^*(x - \tilde{x},\ y - \tilde{y})\ e^{-2\pi j\,(ux\,+\,vy)}\ dx\ dy \tag{8}$$

where w is a Gaussian window of constant size σ_x, σ_y = const.

2.2 Definition of a Gabor Family

The constant size of the envelope that is assumed in eq. 8 leads to an uniform partitioning of the frequency plane when expanding *f*. Thus the localization of the Gabor transform is equal throughout the entire spectrum. However, sophisticated data analysis should provide a high spatial resolution for high frequencies and a low resolution of lower frequencies. Therefore, we have to adapt the size of the Gaussian window to its frequency position. Due to the scaling theorem of the Fourier transform shifting and increasing the width and length of the Gaussian in frequency space go in hand with different modulation and a decrease of width and length of the wavelet's envelope in the spatial domain. In fig. 3a both frequency position and effective width of the Gaussian were doubled. Fig. 3b illustrates the corresponding real part of the wavelet. Obviously, we can generate a whole family of self–similar functions of this type, simply by shifting and scaling a Gaussian in the frequency plane.

Fig. 3. The effect of scaling and shifting in frequency space:
a) Gabor function of different size, position and frequency in Fourier domain.
b) Real part of the resulting wavelet in spatial domain
($u_2 = 0.37$, $v_2 = 0.15$, $\omega_2 = 0.4$, $x_0 = y_0 = 0$, $\sigma_u = \sigma_v = 0.15$, $a = 2$).

Hence, a scheme to construct a set of self–similar Gabor functions as an expansion basis for f can be set up.

The idea of building an expansion basis from self–similar functions of different size and orientation immediately leads to the generic wavelet transform, whose mathematical principles are described in [13], [41], [37] or [14]. Basically, it can be considered as the inner product of any finite energy function f with a set of self–similar basis functions $\psi_{a,b}$, that are derived from each other by scaling and shifting of one prototype using the parameters $a, b \in \mathbb{R}$ respectively. Its 2D representation can be addressed as:

$$CWT = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \psi_{a,b}(x,y)\ f^{*}(x,y)\ dxdy\ = \left\langle \psi_{a,b}\ ,\ f \right\rangle \tag{9}$$

with

$$\psi_{a,b}(x,y) = \frac{1}{\sqrt{|a_x\ a_y|}} \psi\left(\frac{x - b_x}{a_x}, \frac{y - b_y}{a_y}\right) \tag{10}$$

The brackets <,> denote the inner product operation.

Its Fourier transform is given by

$$\Psi_{a,b}(u,v) = \sqrt{|a_x\ a_y|}\ e^{-2\pi j(b_x + b_y)}\ \Psi(a_x u, a_y v) \tag{11}$$

Note that the normalization forces the bases to hold Parseval's energy equivalent.

In wavelet theory, the bases are usually assumed to satisfy the constraints of orthonormality

$$\left\langle \psi_l\ , \psi_k \right\rangle = \delta(l - k),\ \ \delta : \textit{Kronecker–delta–function} \tag{12}$$

and band–pass behavior:

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \psi_{a,b}(x,y)\ dx\ dy\ =\ 0 \tag{13}$$

Note furthermore, that due to its definition the Gabor function has an intrinsic DC–fraction in it's real (cosine) component. Hence, any wavelet derived from it will not meet eq. 13. More specifically, any two scaled and shifted versions of Gabor type will not satisfy eq. 12 as well.

2.3 Frequency Decomposition with Gabor Wavelets

Depending on the job for which they are tailored, additional constraints on the bases are often demanded, such as compact support, smooth shape, fast decay, closed–form descriptions, etc. It has turned out that the construction of bases satisfying these criteria is a non–trivial optimization process and usually a compromise has to be found, as semi–orthonormality [56]. However, relaxing the orthonormality and bandpass constraint, we can construct a wavelet basis using the Gabor functions of different scale, shift and orientation [15]. In this case a complete set of self–similar functions is defined by

$$\psi^{G}_{mpq\Theta}(x,y)\ =\ a^{m}\ \psi^{G}(x',y') \tag{14}$$

$$\begin{aligned} x' &= a^{m}\ ((x\ \cos\Theta\ +\ y\ \sin\Theta)\ -\ p) \\ y' &= a^{m}\ ((-x\ \sin\Theta\ +\ y\ \cos\Theta)\ -\ q) \end{aligned} \tag{15}$$

Here, we assume a circular scaling with $a_x = a_y = a$. The parameters represent transformations of the mother wavelet in scale *(a,m)*, translation *(p,q)* and orientation *(Θ)*. The index *m* runs from lower to higher frequencies in our paper, since the wavelets are constructed accordingly as in appendix A.

As stated earlier, the most effective way to construct a multiresolution wavelet basis is to find appropriate criteria for specifying the Gaussian in the frequency plane. Once constraints for setting *a* and $\Delta\Theta$ are found, a complete set of Gaussians represents our filter pyramid and the decomposition is obtained by subsequent filtering and inverse Fourier transform. In our implementa-

tion, we define the decomposition using a lower frequency bound ω_0 and an angular resolution $\Delta\Theta$. Figure 4 shows how the frequency plane is decomposed by superimposing Gaussian filters. The mathematical details of the construction rules we propose here are described in Appendix A at the end of this paper.

Fig. 4. Decomposition of the frequency plane with Gabor wavelets. The parameters are $\omega_0 = 0.1$, $\Delta\Theta = \pi/4$, $a = 2$ according to Appendix A.

Let $i(x,y)$ be a 2D image function and $I(u,v)$ be it's Fourier transform, we obtain a set of convolution products $g_{m,i}(x,y)$ by multiplying $G_{m,i}(u,v)$ with $I(u,v)$ and subsequent inverse Fourier transform:

$$g_{m,i}(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} G_{m,i}(u,v)\ I(u,v)\ e^{2\pi j(ux+vy)}\ du\ dv \tag{16}$$

where $G_{m,i}(u,v)$ denotes the Fourier transform of $\psi^{G}_{mpq\Theta}(x,y)$ according to Appendix A.

This scheme provides a more effective way of building up the wavelet pyramid – a way which is preferable to computing all inner products of the scaled and shifted Gabor functions $< i(x,y), \psi_{mpq\Theta} >$ explicitly. This is because the convolution products computed by eq. 16 contain the inner products of all shifted versions of a wavelet given at m and Θ. The implementation scheme from above corresponds to a filter bank, indicated in fig. 5.

Fig. 5. Filter bank, that implements the Gabor decomposition.

The method introduced represents the feature extraction module in our visualization pipeline and can be employed to decompose laser range data sets.

It should also be noted here that wrap–around problems and cut–off errors arising from the FFT of finite data sets have to be avoided. Therefore, we strongly recommend reflecting the data at their boundaries and fading them out exponentially. All computations in this paper are performed that way.

3 Shape and Texture Coding using Gabor Wavelets

3.1 Characteristics of Laser Range Data

This section elaborates the decomposition of data sets using the Gabor wavelets introduced above. Although our further investigations are focussed on laser range data, the decomposition scheme as well as the feature based visualization pipeline is not restricted to that. We will show that in particular when dealing with complex–valued data this method provides an elegant and efficient method for visual data inspection. Consequently, any type of shape and texture information, such as terrain data and aerial photographs, are well suited for this scheme.

We employ laser range data from a Cyberware laser scanner, which provides highly accurate shape and color values of scanned objects. The data sets can be interpreted as R,G,B range images defined on a cylindrical coordinate system, as depicted in fig. 6. The grid size is 512 x 480 at a resolution in the range of about 0.3 mm. A typical scan is rendered in fig. 7, that shows the range information (fig.7a) and the color mapped as a texture onto the shape (fig.7b).

The scanner moves a diffuse line light source around the object synchronously to the range detector. This is the reason why the illumination is preserved constant at each surface point and the texture information can be interpreted in terms of local surface properties. It is not affected by specific illumination conditions. It has to be mentioned, that the albedo of human facial skin is very high and a respective reflection model has been developed by [31].

Due to the cylindrical coordinate system of the scanner, the data have to be projected into Cartesian coordinates representing the 2D image function in *(x,y)*. The projection can be considered as taking the normalized z_S–coordinate of the scanner system as the y–coordinate of the image and the angular coordinate γ_S as it's x–coordinate. Consequently, this operation unrols the cylinder onto a plane, as in fig. 6b.

$$\begin{aligned} z_S &\mapsto y \\ \frac{\gamma_S}{2\pi} &\mapsto x \end{aligned} \tag{17}$$

where (z_S, r_S, γ_S) denote the cylinder coordinates of the scanning system.

Fig. 6. a) Coordinate system employed for data acquisition with the laser scanner.
b) Transformation of the range data onto a rectangular grid.

Fig. 7. Shape and texture obtained from a female face (Sylvia) and its encoding scheme.
a) Range rendered with Gouraud–shading.
b) Texture–mapped color information.
c) Encoded range information.
d) Encoded grey–level texture information.
(raw laser range data: courtesy provided by Computer Graphics Center, Darmstadt, Germany)

3.2 Complex–valued Coding of Shape and Texture Data

The question arises how to encode both color and shape in order to decompose them efficiently with the Gabor pyramid. A straightforward way would be to treat R,G,B and range separately in terms of four different decompositions. This is however very expensive and provides highly correlated and high–dimensional features. Referring back to section 2, the Gabor function is composed of a symmetric and an antisymmetric oscillation term, that account for the real and imaginary components of the data, respectively. Moreover, sine and cosine wavelets are orthogonal to each other. Hence, in our approach range and color are interpreted as amplitude and phase of a complex–valued image function that is convoluted with a complex–valued wavelet.

Let *r(x,y), g(x,y), b(x,y), s(x,y)* be the color and range information at a particular point *(x,y)*. In order to compute the amplitude *a(x,y)* we extract luminance according to the rules of colorimetry [61]:

$$a(x,y) \;=\; 0.31\;\; r(x,y) \;+\; 0.59\;\; g(x,y) \;+\; 0.10\;\; b(x,y) \tag{18}$$

In fact, the CCD sensor is not calibrated explicitly, but most systems lie near the CIE standard of that equation. The range function is taken immediately as phase of the image function and the description of *i(x,y)* is obtained as:

$$i(x,y) = a(x,y)\ e^{2\pi j s(x,y)} \tag{19}$$

with its real part $i_R(x,y)$ and its imaginary part $i_I(x,y)$:

$$\begin{aligned} i_R(x,y) &= a(x,y)\ \cos(2\pi s(x,y)) \\ i_I(x,y) &= a(x,y)\ \sin(2\pi s(x,y)) \end{aligned} \tag{20}$$

The range function is assumed to be normalized between 0 and 1.

Fig. 7c,d depicts also, how the data appears, as encoded by means of phase and amplitude. Although information is lost by transforming color into the scalar–valued luminance, shape and grey values are, however, still superior to color information when extracting features from objects [32]. Furthermore the efficient encoding scheme provides computational advantages with FFT methods.

Once the data is encoded as in eq. 19, we can compute it's Fourier transform $I(u,v)$ and apply the Gabor decomposition, as defined with eq. 16. This is illustrated in fig. 8, which presents a decomposition of the data from fig. 7. The convolution products $g_{m,i}(x,y)$ are arranged according to the orientation preference of their wavelet. The depth of the pyramid is $M=4$, at $\Delta\Theta = \pi/4$ and the DC parts are presented in the middle of each picture, respectively. Due to the complex–valued results of $g_{m,i}(x,y)$, the inverse Fourier transform produces two resolution pyramids, one for the real and one for the imaginary component, respectively. These pictures elucidate the orientation and frequency selectivity of the wavelets. The low–pass functions at $m=0$ only account for rough global features since their spatial location is rather bad. The high–pass wavelets at $m=3$, however, reveal the fine–grain details in the data. Obviously, due to the non–orthogonality, the different responses are correlated.

Fig. 8. Decomposition of Sylvia's image:
a) Real part of the Gabor pyramid.
b) Imaginary part of the Gabor pyramid.

Note, that the pyramid is not subsampled in this picture. A strict representation according to the definitions of the WT, however would require further sampling of the $g_{m,i}(x,y)$ at the current rate given by the choice of ω_0 and $\Delta\Theta$.

3.3 Extracting Local Data Features

Evidently, this complex valued decomposition pyramid from fig. 8 gives a multiresolution view onto the data. Due to the localization properties of the Gabor function a set of multidimensional local data features can easily be derived from it just by evaluating the convolution products $g_{m,i}(x,y)$ at any point of interest. Fig. 9 illustrates the respective procedure of local feature extraction. Here, we assume the decomposition pyramid arranged into a multidimensional set, where the dimension index k ranges from 1 (DC–part) to $K = 2\,M\,\lfloor \pi / \Delta\Theta \rfloor + 2$. In the further sections of the paper we refer to a feature vector $\boldsymbol{g}_l = (g_{l1},..,g_{lK})^T$ as the set of scalar–valued coefficients $\left\{ g^{\mathrm{re}}_{m,i}(x_l, y_l), g^{\mathrm{im}}_{m,i}(x_l, y_l) \right\}$ evaluated at any data point (x_l, y_l) according to the illustrations of fig. 9. This interpretation of the features implicitely transforms the former complex–valued vector into a scalar–valued one of double length. Hence, the following steps in our pipeline rely on scalar–valued computations although PCA, normalization and Kohonen map could be extended. This however would restrict the method very much to the specific application of this paper without any gain.

Fig. 9. Extraction of local data features from the wavelet decomposition.

Although the reconstruction of the data from the Gabor decomposition requires the non–trivial computation of a dual frame [3], it is still of interest to locally expand the data, as in fig. 10. In this picture, the feature vectors are superimposed only within the area indicated around the right eye and all others are neglected. Range and amplitude values are computed according to eq. 19 and mapped as displacements (fig. 10a) and texture (fig. 10b) onto a plain cylinder. Obviously, they encode important data features, such as curvature or texture modulations around the center of interest.

Fig. 10. Local reconstruction of the face shape and texture superimposing the wavelet features within the demarcation of fig. 9:
a) Range around right eye.
b) Texture around right eye.

4 Visualization of Multidimensional Face Features

4.1 Overview

In the section above, we proposed a method for getting multidimensional data features by decomposing them with multiresolution Gabor wavelets. Unfortunately, there is no immediate way to visualize the topology of this feature space. Hence, we have to perform additional data analysis steps to map the most important features down into a subspace that can be visualized.

The problem of visualizing multidimensional features in complex dataspaces is one of the key issues in scientific visualization [43], [50] and is addressed in many applications, such as fluid flow and tensor visualization [35], statistical data visualization [63] or multispectral imaging [28]. The main point is to find orthogonal projection methods that preserve the most important data features. One basic method that addresses this problem is the principal component analysis (PCA). It optimizes the mapping procedure in a least–square error sense based on data statistics.

Consequently, we propose a visualization pipeline as figured out in fig. 11. This concept comprises an embedded framework using different methods proposed by one of the authors in [25] and [27]. Once the feature vectors $\boldsymbol{g}_l$ are extracted from the decomposition, they are fed into a subsequent processing pipeline of PCA–analysis, normalization and clustering into R,G,B color space.

As mentioned earlier, the Gabor wavelets provide non–orthonormal expansions of the data and the resulting feature vector is more or less highly correlated, depending on the choice of ω_0 and $\Delta\Theta$. The decorrelation can be accomplished by further expansion into its principal components.

Fig. 11. Feature–based visualization pipeline employed.

This first breakdown of the number of data dimensions is controlled by thresholding the eigenvalues associated to each eigenvector. The coefficients of the feature vector transformed by the eigenvector matrix are decorrelated and have to be normalized. Visualization of the data is figured out by a self–organizing neural network, as proposed in [23]. This network clusters the data and maps them automatically onto the R,G,B color space with the constraint of coding similar data features in terms of similar colors. The method has the advantage of reducing similar features

to a limited set of clusters, rather than to visualize a whole subspace. Hence, the results obtained are much more expressive than an immediate mapping of the first three eigenvectors into R,G,B.

4.2 Visualizing Principal Components

Generally, dimensionality reduction is strongly connected with finding subspaces satisfying respective optimization criteria. In statistical data analysis [20], we can find several techniques for this task. One of the most famous methods, taking the optimization as an eigenvalue problem, is the principal component analysis or Karhunen–Loève expansion. In general, this method aims to find a subset of principal directions in a data set of any statistical distribution. The corresponding basis vectors satisfy the constraint of orthonormality and the subspace defined by these vectors diagonalizes the covariance matrix of the data. The transformation of an initial data set into the space of principal components can be formulated as follows:

Let$\{\ \boldsymbol{g}_l\ \}$: $l = 1$,.., L be a data set of the dimension K and $\overline{\boldsymbol{g}}$ a mean vector:

$$\bar{\boldsymbol{g}} = \frac{1}{L}\sum_{l=1}^{L}\boldsymbol{g}_l \tag{21}$$

The principal component analysis essentially looks for a set of orthonormal vectors $\boldsymbol{o}_k$ with the constraint of

$$\lambda_k = \frac{1}{L}\sum_{l=1}^{L}(\boldsymbol{o}_k^T\ (\boldsymbol{g}_l - \bar{\boldsymbol{g}}))^2 \stackrel{!}{=} \max \tag{22}$$

and

$$\boldsymbol{o}_k^T\boldsymbol{o}_j = \delta(k - j) \tag{23}$$

$\boldsymbol{o}_k$ and λ_k represent the eigenvectors and eigenvalues of the covariance matrix $\boldsymbol{C}$, which can be estimated as

$$\boldsymbol{C} = \frac{1}{L}\sum_{l=1}^{L} (\boldsymbol{g}_l - \bar{\boldsymbol{g}})(\boldsymbol{g}_l - \bar{\boldsymbol{g}})^T \tag{24}$$

Note, that its dimension is given by K $\times$ K.

Hence, the set of $\boldsymbol{o}_k$ diagonalizes the covariance matrix, as

$$\begin{bmatrix} (c_{11} - \lambda_k) & c_{12} & . . & c_{1K} \\ c_{21} & (c_{22} - \lambda_k) & & c_{2K} \\ \vdots & & . . & \vdots \\ c_{K1} & c_{K2} & . . & (c_{KK} - \lambda_k) \end{bmatrix} \begin{bmatrix} o_{k1} \\ \vdots \\ o_{kK} \end{bmatrix} = 0 \tag{25}$$

There are numerous methods for numerically handling this approach for huge covariance matrices [20], such as singular value decomposition.

After solving this eigenvector problem, the feature vectors $\boldsymbol{g}_l$ have to be projected into the eigenspace spanned by the $\boldsymbol{o}_k$

$$\phi_{lk} = \boldsymbol{o}_k^T \boldsymbol{g}_l \tag{26}$$

and the coordinates in eigenspace are obtained by ϕ_{lk}.

Conversely, any datum $\boldsymbol{g}_l$ can be expanded as a linear combination of $\boldsymbol{o}_k$ and ϕ_{lk}

$$\boldsymbol{g}_l = \sum_{k=1}^{K} \phi_{lk} \boldsymbol{o}_k \tag{27}$$

The dimensionality reduction can now be achieved by ordering the eigenvectors $\boldsymbol{o}_k$ according to the absolute value of their corresponding eigenvalue λ_k and by taking only the most significant subset for the data expansion in eq. 27.

The vector $\Phi_l = (\phi_{l1}, .., \phi_{lK})^T$ can be interpreted as the decorrelated feature vector.

The remaining eigenvectors define a subspace that minimizes the average error of information lost by the reduction of the number of dimensions.

This method is very popular for any kind of signal analysis and image coding, i.e. for removing correlation from data like face images [23].

The principles of PCA force most of the data energy to be concentrated in a few significant eigenvectors. Therefore, once the PCA based feature vectors Φ_l are computed for each initial $\boldsymbol{g}_l$ they have to be normalized for further processing. Since the a priori probability of these vectors is unknown, we assume uniform distribution and normalize their components according to:

$$\phi_k^{min} = \min_l(\phi_{lk}) \qquad \phi_k^{max} = \max_l(\phi_{lk}) \tag{28}$$

$$\tilde{\phi}_{lk} = \frac{\phi_{lk} - \phi_k^{min}}{\phi_k^{max} - \phi_k^{min}} \tag{29}$$

Fig. 12 shows the 6 most significant feature coordinates $\tilde{\phi}_{l1}$,.., $\tilde{\phi}_{l6}$ for $l \in \{x, y\}$ and the 3 least significant ones as computed from the laser range data set from fig. 7. The respective eigenvalues are indicated below the picture, as well.

Fig. 12. a) Coordinates of the feature vectors of Sylvia's image encoded in eigenspace (see also fig. 7).

One straightforward way to visualize the multidimensional feature space of $\tilde{\Phi}_l$ as it is derived from the Gabor pyramid is to encode it into R,G,B. This is illustrated in fig. 13 where a smooth color representation of the 3 most significant coordinates in eigenspace is depicted. Although this presentation reveals a first sketch of similarities in the multidimensional feature space in terms of similar colors, yet it does not provide a sophisticated feature extraction and the visual interpretation of the resulting images remains difficult. Hence, additional clustering algorithms are required to group similarities in the data. This is figured out by a topological mapping neural network explained below.

Fig. 13. Distribution of the coordinates of the 3 most important eigenvectors of Sylvia's image encoded into R, G and B, respectively.

4.3 Topological Mapping

The 3D extension of the Kohonen map was introduced by the authors in [22] and employed in different applications, such as [23] or [9]. It is basically a self-organizing network which is trained with or without supervision. It aims at an organization of the input patterns to a topological structure represented by its neurons, where the relations between different patterns are preserved as much as possible.

The Kohonen map is a two-layered network. The first layer of neurons can be considered as similar to a group of sensors picking up the data, entirely connected to a second, 3D–layer: the competitive layer. Figure 14 shows the topology of the network. The weights associated with the connections are adjusted during training where only one single neuron in the competitive layer can be active at a time. This neuron represents the cluster to which the data set belongs in the spirit of c-means clustering. Due to the training rules explained below, the Euclidian distance between two neurons reacting on different input patterns can be a measure that determines the similarity of the two patterns.

The training of the network is determined by presenting feature vectors $\tilde{\Phi}_l$ randomly to the input layer of the network whose connection weight vectors $\boldsymbol{m}_h$ of all competitive neurons h are initialized by random values. We choose K input neurons according to the data dimension and define a Euclidean distance d_h between $\tilde{\Phi}_l$ and $\boldsymbol{m}_h$ with

$$d_h = \| \tilde{\Phi}_l - \boldsymbol{m}_h \| = \sqrt{\sum_{k=1}^{K} (\tilde{\phi}_{lk} - m_{hk})^2} \tag{30}$$

The neuron e with the minimum distance is then activated, where

$$d_e = \min_h (d_h) \tag{31}$$

The updating of the weights m_{hk} associated with the neurons is only performed within a proximity ($h \in N_e(t)$) of e. This proximity $N_e(t)$ is reduced with increasing training time t. The updating conforms to Eq. 32, where $\eta(t)$ represents a time-dependent learning rate:

$$m_{hk}^{(t+1)} = m_{hk}^{(t)} + \Delta m_{hk}^{(t)} \tag{32}$$

$$\Delta m_{hk}^{(t)} = \begin{cases} \eta(t)(\tilde{\phi}_{lk} - m_{hk}^{(t)}), & \textit{if neuron } h \in N_e(t) \\ 0, & \textit{otherwise} \end{cases} \tag{33}$$

$$\eta(t) = \eta_0(1 - \frac{t}{T_{max}}), \quad t \in [0,..,T_{max}] \tag{34}$$

This rule refers directly to c-means clustering [17], where the $\boldsymbol{m}_h$ represent the centroids. The time-dependent neighborhood can be described for rectangular areas as follows:

$$\begin{aligned} x_e - d(t) < x < x_e + d(t) \\ y_e - d(t) < y < y_e + d(t) \\ z_e - d(t) < z < z_e + d(t) \end{aligned} \tag{35}$$

where

$$d(t) = d_0(1 - \frac{t}{T_{\max}}), \quad t \in [0,..,T_{\max}] \tag{36}$$

Fig. 14. a) 3D Kohonen feature map and the representation of its neurons as entities in the RGB color space.
b) Class assignment to single neurons.

The network has two essential properties related to our problem:

First, it separates clusters of the presented data by mean vectors $\mathbf{m}_h$ that are associated as weights to the neurons. Secondly, it performs a topological ordering of the competitive neurons in a sense

that neighboring neurons in the layer represent similar clusters in multidimensional space and thus it achieves a further dimensionality reduction.

Since the neurons in the competitive layer are ordered topologically, neighboring neurons react to similar data vectors in the input space. The mapping can be interpreted as reduction of any *K*-dimensional input pattern into 3 dimensions, preserving the topology of the data as much as possible. The resolution of this discrete 3D space is given by the number of competitive neurons, i.e., by the clusters.

Referencing the axes of the cube with the primaries R, G, and B, each neuron of the competitive layer represents a discrete entity in R, G, B space, i.e., it corresponds to a particular color triplet. Thus, the similarity of the color provided by the reacting neuron refers to a neighborhood in *K*-dimensional space.

Hence, the dynamics of the network are strongly related to the PCA analysis. In this case, however, not only all feature vectors are projected into an orthogonal subspace, but a discrete subset of centroids is computed, representing the features. This topological structure is further represented in terms of R,G,B colors. Therefore, in our approach, the usage of the PCA for huge data sets is limited to a rough scaling of the dimensionality and any further fine–tuning is accomplished by the Kohonen map.

Figure 15 demonstrates the impact of clustering and visualizing the features from Sylvia's image. In order to realize the geometric relationships, the output of the Kohonen map is rendered as a texture onto the facial shape. The parameters were selected with 8x8x8 neurons and $M=4$, $\omega_0 = 0.1, \Delta\Theta = \pi/4$. In fig. 15a the inital 26–dimensional feature vector $\boldsymbol{g}_l$ was feed immediately into the network, which furnishes the R,G,B values at each surface point. The network clearly distinguishes shape features, such as local curvature or textural consistency between the left and right hemisphere of the face. The similarity of colors clearly appears in spatially coherent regions unless in those of frequency components spread throughout the spectrum. Regions of high, low and strong directional curvature are separated, such as around mouth, nose and eyes. Fig. 15b contrasts to those results using a first rough breakdown of the dimensions from 26 to 7 through the usage of the PCA. On the one side, the regions extracted appear even more homogeneous, since some of the fine grain information was cut off by the PCA. On the other hand, however,

the topology preservation of the Kohonen map does not reveal as evidently as in the previous picture. This is because the PCA encodes the variance of the features which has not to correspond to coherent facial regions.

Fig. 15. Cluster analysis and visualization of features derived from Sylvia's image:
a) Immediate subspace mapping using the Kohonen map only.
b) Cascaded mapping with PCA and Kohonen map.

In order to stress the influence of the texture information and the importance of the PCA, fig. 16 depicts results derived from a segmentation of the texture data of Sylvia's image without including range. Figure 16a shows the b/w texture image and fig.16b its segmentation using PCA preprocessing. Due to local distortion in the texture, the image looks more noisy than those presented in fig. 15. The smoothing properties of the range data cannot be harvested in this case. These results strike even more in Fig. 16c where the features are segmented without PCA preprocessing. One important aspect is related to the invariance against affine transforms. In principle, rotation and scaling can be interpreted as a shifting of the respective coefficients in our feature vector. This is because of the properties in Fourier space, where rotation is transformed into a rotated spectrum and scaling results in a frequency modulation. Hence, the respective basis functions in the multiresolution pyramid respond in different frequency channels. Due to the finite dimensionality of the feature vector, some coefficients are pushed out of the vector, whereas others will enter at the lower and higher frequency bounds. Because of the Euclidian metrics employed for segmentation, we have to keep the most important coefficients in our vector in order to preserve the topology and the distances in feature space. For this reason, a PCA based preprocessing helps to extract those coefficients and to keep the segmentation tolerant against rotation and scaling. Translation invariance of the segmentation is achieved by the localized features. An illustration of this important subject is given in fig. 16d – 16f, where a scaled and rotated version of the texture data is presented and the segmentations are generalized from, fig. 16b,c. Evidently, the color similarity and thus the preservation of similar features is performed much better when using the PCA. For instance, the high frequency regions around the eyes, nostrils and mouth appear in bluish colors in contrast to cheeks and forehead, that come out in orange, red and green.

Fig. 16. Influence of range data and affine transforms on the segmentation:
a) Texure data of Sylvia's image.

b) Segmentation with PCA.
c) Segmentation without PCA.
d) Scaled and rotated version of the image.
e) Generalization with PCA.
f) Generalization without PCA.

Besides non–supervised clustering, classification is an essential part in feature–based visualization. In this case, the method has to match previously trained patterns. For supervised classification, however, each neuron – and also each cluster centroid – has to be assigned to a certain class, depending on the definition of the user. This can be done by interactive selection of training areas and by a majority voting of each neuron stimulated by the training set (see fig. 14b). After this, each neuron has an associated class and the network is able to classify. However, during the organization process the goal was to find a limited set of centroids representing the data in a *c*–means sense rather than to find optimal placements of the decision boundaries in a minimum error (Bayes) sense. For this reason, the network can be once again trained with a supervised postprocessing in order to move corresponding centroids towards the Bayes decision boundary and to improve the classification result.

This postprocessing is well known as learning vector quantization (LVQ 1, 2, 3) [38] and the self–stabilizing type 3 can be described as follows:

For a given input pattern $\tilde{\Phi}_l$ let $\boldsymbol{m}_h$ and $\boldsymbol{m}_j$ be the closest centroids to $\tilde{\Phi}_l$. We modify the centroids according to

$$\begin{aligned} \boldsymbol{m}_h^{(t+1)} &= \boldsymbol{m}_h^{(t)} - \alpha[\tilde{\Phi}_l - \boldsymbol{m}_h^{(t)}] \\ \boldsymbol{m}_j^{(t+1)} &= \boldsymbol{m}_j^{(t)} + \alpha[\tilde{\Phi}_l - \boldsymbol{m}_j^{(t)}] \end{aligned} \tag{37}$$

/ if* $-\boldsymbol{m}_h$ *and* $\boldsymbol{m}_j$ *are the two closest weight vectors to the input vector,*
$-$*input and* $\boldsymbol{m}_j$ *belong to the same class,*
$-$*input and* $\boldsymbol{m}_h$ *belong to different classes and*
$-$*input falls into the window */*

$$\boldsymbol{m}_e^{(t+1)} = \boldsymbol{m}_e^{(t)} + \varepsilon\alpha[\tilde{\Phi}_l - \boldsymbol{m}_e^{(t)}]$$
/* *if* *input,* $\boldsymbol{m}_i$ *and* $\boldsymbol{m}_j$ *belong to the same class* $(e \in \{i,j\})$
where ε *and* α *are small constants* */ (38)

The window is defined as a symmetric area around the midplane of $\boldsymbol{m}_h$ and $\boldsymbol{m}_j$. Then $\tilde{\Phi}_l$ falls into the window if

$$\min(\frac{d_h}{d_j}, \frac{d_j}{d_h}) > \mu \tag{39}$$

$$\mu = \frac{1 - \varkappa}{1 + \varkappa} \tag{40}$$

Where d_h and d_j are the two distances of $\tilde{\Phi}_l$ to $\boldsymbol{m}_h$ and $\boldsymbol{m}_j$. The threshold μ is calculated according to eq. 40 and the relative window size $\varkappa$ is chosen to be about 20%.

A detailed study of LVQ and of related methods can be found in [38].

5 Applications

5.1 Face Data Base

The goal of the following investigation reports how the previously introduced method performs and generalizes on multiple range image data sets. For this purpose, a data base of 10 entities was built up, as shown in fig. 17. It consists of both male and female range images that were clipped to the depicted masks. Both range and color information is presented. The left 7 images were employed to train the method, whereas the right 3 images were used to investigate the behavior of the generalization. The Gabor decomposition was applied separately on each of the 7 training samples and the Kohonen map was trained by randomly selecting 1 mil. feature samples from the resulting pyramids. We used a constant map size of 8x8x8 neurons, i.e. clusters. The parame-

ters for the decomposition were selected to $\Delta\Theta = \pi/4$ and $\omega_0 = 0.1$ and an additional overlap of 2 was added to tune the localization of the wavelets.

Fig. 17. Initial face data base a) with and b) without texture information.
Upper rows and left: Range images used for training,
lower rows right: Range images used for generalization.

5.2 Results

In order to emphasize the influence of the PCA figure 18a illustrates the results obtained by the method first without PCA. In this case, the features were mapped immediately from 34 dimensions to R,G,B using the Kohonen map. Obviously, the wavelets extract similar facial features that appear as similar colors on each face, such as tip of the nose in red–orange colors, or the cheeks that come up either in blue–green or violet. The mouth, mostly revealed in red–pink and moustache boundaries are extracted, as are eye brows (light orange). It is interesting to verify, that the left and right hemispheres are clearly distinguished and because of the different colors associated, belong to different clusters. Some of the results reveal a more or less symmetric or antisymmetric color map, depending on the person analyzed. At this point, our method can be contrasted with curvature analysis, or slicing methods. In particular, when the results on the training data are compared with those obtained on the generalization data set (lower row), this technique evidently outperforms robust results, since color similarities referring to anatomic regions are rather well preserved. In general, due to the influence of the high–frequency components, within a single face there is hardly any large coherently clustered region. The similaritiy of colors in neighbored clusters is, however, preserved in regions of low spatial frequencies.

Figure 18b depicts the same investigation, but in this case with a PCA downmapping of the dimensions from an initial 34 to 7, preserving 99 % of the signal energy encoded in the eigenvectors. The topology preservation of the Kohonen map is not as obvious as in the previous figure, since the PCA accounts for the maximum variance in the features. One of the advantages of the PCA is that the regions clustered by the neural network appear in most cases more homogeneous and coherent because some fine grain detail information gets lost with the downscaling procedure. In both computations, the most discontinuous regions appear around the eyes. This is due

to the discontinuouities in range, which spread the local spatial frequencies throughout the spectrum. The pupils of most of the test persons, however, are detected in similar colors.

Fig. 18. Face features as similarities of colors mapped as textures to the range data.
a) without PCA and
b) including PCA.

Other interesting investigations can be accomplished by trying to retrieve similar regions and landmarks in the face data base. For this purpose the Kohonen map was trained with a supervised LVQ postprocessing on the results of fig. 18b using 100000 additional cycles. During this procedure, only features within the training areas demarcated onto the face in fig. 19a were fed randomly into the network. Four different feature types were selected, one each for tip of the nose (blue), nostrils (yellow), pupils (green) and corner of the mouth (red). Figure 19b depicts the results, where the distribution of the trained features throughout each face is presented. It can be stated, that all regions are properly retrieved, such as the tip of the nose or pupils. Considering the especially small training areas, features similar to the corner of the mouths are detected mostly around the eyes, eye brows, and moustache, where discontinouities along with high local curvature in range can also be located. Furthermore, green regions appear systematically around the throat of each person. This is due to an artifact arising from the linear interpolation of the scanned data, which usually lacks in the green region. Although both nostrils and pupils are dark regions in the texture map, they are clearly separated in all images. The diagram in fig. 20 compares again the results within the training areas of fig. 19. Besides the two peaks within the nostrils the overall error is relatively small.

Fig. 19. Detection of similar features extracted from a supervised classification postprocess.
a) Training areas demarcated on faces.
b) Detection of the similar regions.

Fig. 20. Error rate for matching the significant regions within the training areas.

Conversely, if a robust detection of previously learned training areas is required for matching or recognition, a procedure can be set up as indicated in fig. 21. In this simulation, the method was trained exclusively on Sylvia's image and the DC components were neglected in the feature vector. Hence, the segmentation looks noisy, no color similarities can be recognized, and the result is badly shaped for visual inspection. The LVQ applied subsequently, was trained to distinguish,

left and right eye (green and blue), tip of the nose (red), and left and right corner of the mouth (pink and yellow). Fig. 21c shows the performance of the detection. The landmarks are retrieved almost perfectly without further errors because of the highly localized features.

In order to illustrate the importance of the range and texutre information for the outperformance of the method the results are contrasted to those achieved by using range and texture data only, as depicted in fig. 21d and fig. 21e. Although both computations are performed with the same parameters, yet the retrieval of the landmarks is figured out much worse than in fig. 21c.

Note again, that the texture data is recorded under normalized conditions and hence represents facial surface properties and is not affected by specific illumination.

Fig. 21. Detection of facial feature points in Sylvia's face:
a) RGB representation of the segmentation.
b) Training areas.
c) Performance of detection of the landmarks.
d) Performance of detection of the landmarks using range data only.
e) Performance of detection of the landmarks using texture data only.

All computations of this paper had been performed on a SGI–Indigo^2 workstation with a R4400 processor and 64 MB main memory. The file–oriented implementation of the method requires to generate the appropriate filters, which took about 205 CPU seconds. Generation and normalization of the feature vectors was figured out in 61 CPU seconds and the PCA took 48 s. The self–organization of the Kohonen map was computed in 157 s with 200000 cycles and 7 coefficients whereas the same procedure needed 472 s for the full feature vector without PCA. Hence the PCA increases the computational performance of the method. The segmentation and classification was done in 42 s respectively in 83 s. Note, that the figures are valid for one image.

6 Conclusion

We introduced a generic and robust method for feature–based visualization of multidimensional data sets by decompositions with Gabor wavelets. Results have been presented, that show the capabilities of the method, even in very complex data sets. Although it was primarily applied to encode and analyze laser range data from human faces it is not restricted to and can be extended to any kind of complex data, such as 3D– or 4D–velocity fields. However, one of the main aspects when encoding data in the way we propose is, that there is no immediate way to perfectly recon-

struct them in the general case. Furthermore, the current scheme does not accomplish any subsampling of the pyramid and requires extensive memory. This is specifically a serious criteria in multidimensional applications. [3] for instance, shows that wavelet frame theory, however, defines bounds for the design of dual function spaces for a perfect reconstruction of Gabor frames. Since Gabor wavelets are powerful feature extractors, further research has to be conducted to find perfect reconstruction frames within these strict limitations and to achieve more compact coding schemes. Other alternatives have to be considered, such as nonseparable extensions of biorthonormal wavelets preserving directional selectivity.

Another important aspect is how the method performs under noise. Due to the spectral estimation–like method, we expect much more robust results than those, for instance, given by curvature analysis techniques for range data. In this context, a common strategy is to weight the wavelet coefficients according to their frequency localization when using them for coding. For data analysis, however, the weights have to be selected much more carefully and strongly depend on the signal characteristics. This leads immediately to higher–order spectral estimation methods, which should be in the scope of future activities as well.

7 Acknowledgement

The authors wish to thank the Computer Graphics Center in Darmstadt, from where the raw laser range data set was provided. Thanks also to S. Spanier–Mason for proofreading the English manuscript. The authors are also grateful to the referees for providing very useful criticism.

8 References

[1] A. Akansu, and R. Haddad, *Multiresolution Signal Decomposition. Transforms, Subbands, Wavelets.* Telecommunications Series, Boston: Academic Press, 1992.

[2] J. Aloimonos and D. Shulman, *Integration of Visual Modules. An Extension of the Marr Paradigm.* New York: Academic Press, 1989.

[3] Benedetto, Frazier, *Wavelets: Mathematics and Applications*. New York: CRC–Press, 1994.

[4] D. Bergeron and G. Grinstein, "A reference model for the visualization of multi-dimensional data," *Proceedings of the EUROGRAPHICS '89*, Elsevier (North–Holland), pp. 393–399, 1989.

[5] P.J. Besl and R.C. Jain, "Invariant surface characteristics for 3D object recognition in range images," *Computer Vision, Graphics and Image Processing*, vol. 33, no. 1, pp 33–80, 1986.

[6] W. Böhm, G. Farin and J. Kahmann, "A survey of curve and surface methods in CAGD," *Computer Aided Geometric Design*, vol. 1, pp. 1–59, 1984.

[7] A. C. Bovik, M. Clark and W. Geisler, "Multichannel texture analysis using localized spatial filters," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 12, no. 1, pp. 55 – 73, 1990.

[8] J. Buhmann, J. Lange, and C. van der Malsburg, "Distortion invariant object recognition by matching hierarchically labeled graphs. *Proc. IJCNN International Joint Conference on Neural Networks*, Washington: IEEE, pp. 155–159, 1989.

[9] Ch. Busch, and M. Gross, "Interactive neural network texture analysis and visualization for surface reconstruction in medical imaging," *Computer Graphics Forum*, vol. 12, no. 3, pp. C49–C60, 1993.

[10] R. Buse and Z. Liu, "Feature extraction and analysis of handwritten words in grey–scale images using gabor filters," *Proc. First IEEE Conf. on Image Processing*, vol.I, pp.164–168, 1994.

[11] T. Chang, and C. Kuo, "Texture analysis and classification with tree–structured wavelet transform," *IEEE Trans. on Image Proc.*, vol. 2, no. 4, pp. 429 – 441, 1993.

[12] C. Chui, *An Introduction to Wavelets*. Boston: Academic Press, 1992.

[13] ————(Edt.), *Wavelets: A Tutorial in Theory and Applications*. Boston: Academic Pess, 1992.

[14] I. Daubechies, "The wavelet transform, time–frequency localization and signal analysis," *IEEE Trans. on Inform. Theory*, vol. 36, pp. 961 – 1005, 1990.

[15] J. G. Daugman, "High confidence visual recognition of persons by a test of statistical independence," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 15, no. 11, pp. 1148 – 1161, 1993.

[16] ————, "Two–dimensional spectral analysis of cortical receptive field profiles," *Vision Research*, vol. 20, pp. 847 – 856, 1980.

[17] R. Duda and P. Hart, *Pattern Classification and Scene Analysis.* New York: John Wiley and Sons, 1973.

[18] G. Farin, *Curves and Surfaces for Computer Aided Geometric Design - A Practical Guide.* 2nd Edition, Boston, New York: Academic Press, 1990.

[19] R. Franke and G. Nielson, "Scattered data interpolation and applications: A tutorial and survey," In H. Hagen and D. Roller (Eds.), *Geometric Modeling: Methods and Their Application*s, pp. 131–160, Berlin: Springer, 1990.

[20] K. Fukunaga, *Introduction to Statistical Pattern Recognition*. 2nd Ed., New York: Academic Press, 1990.

[21] D. Gabor, "Theory of communication," *Journ. Inst. El. En.*, vol. 93, pp. 429–457, 1946.

[22] M. Gross, *Visual Computing.* Berlin: Springer Publishing Company, 1994.

[23] ————, "Subspace methods for the visualization of multidimensional data sets", *Proc. ONR Workshop on Scientific Visualization*, in Rosenblum et al. (Edts.): *Scientific Visualization.* New York: Academic Press, pp. 171–186, 1994.

[24] ————, "Integrated volume rendering and data analysis in wavelet space," *Proc. Dagstuhl Symposium on Scientific Visualization,* to appear, 1995.

[25] M. Gross, R. Koch, L. Lippert and A. Dreger, "Multiscale image texture analysis in wavelet space," *Proc. First IEEE Conf. on Image Proc.*, IEEE Computer Society Press, vol. III, pp. 412–416, 1994.

[26] ————, "A new method to approximate the volume rendering equation using wavelets and piecewise polynomials," *Computers & Graphics*, vol. 19, no. 1, to appear, 1995.

[27] M. Gross and H. Luttermann, "Combining principal component analysis and neural networks for the recognition of human faces - A case study for man-machine-communication," *Proc. of Pacific Graphics*, Singapore: World Scientific, pp. 176–192, 1993.

[28] M. Gross and F. Seibert, "Visualization of multidimensional data sets using a neural network," *The Visual Computer*, vol. 10, no. 3, pp. 145– 59, 1993.

[29] A. Grossmann and J. Morlet, "Decomposition of hardy functions into square integrable wavelets of constant shape," *SIAM Journ. Math. Anal.*, vol. 15, pp. 723–736, 1984.

[30] H. Hagen, "Visualization of large data sets", *Proc. ONR Workshop on Scientific Visualization*, in Rosenblum et al. (Edts.): *Scientific Visualization.* New York: Academic Press, pp. 187–198, 1994.

[31] P. Hanrahan and W. Krueger, "Reflection from layered surfaces due to subsurface scattering", *ACM Computer Graphics, Proc. SIGGRAPH '93,* pp.165–174, 1993

[32] R. Haralick, L. Shapiro, *Computer and Robot Vision.* Vol. I and II, New York: Addison–Wesley, 1993.

[33] L. Harmon, M. Kahn, R. Lasch and P. Ramig, "Machine identification of human faces," *Pattern Recognition*, vol. 13, no. 2, pp. 97–110, 1981.

[34] T. He, S. Wang and A. Kaufman, "Wavelet–based volume morphing," *IEEE Visualization 94*, pp. 85–92, 1994.

[35] L. Hessellkink and T. Delmarcelle, "Visualization of vector and tensor data sets," *Proc. ONR Workshop on Scientific Visualization*, in Rosenblum et al. (Edts.): *Scientific Visualization.* New York: Academic Press, pp. 419–433, 1994.

[36] A. Jain, *Fundamentals of Digital Image Processing.* Englewood Cliffs, N.J.: Prenctice Hall, 1989.

[37] B. Jawerth and W. Swendens, "An overview of wavelet based multiresolution analyses," *University of South Carolina, Department of Mathematics, Internal Report*, 1994.

[38] T. Kohonen, "The self–organizing map," *Proc. IEEE*, vol. 78, no. 9, pp. 1464–1480, 1990.

[39] A. Laine and J. Fan, "Texture classification by wavelet packet signatures," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 15, no. 11, pp. 1186 – 1191, 1993.

[40] T. Lillesand and R. Kiefer, *Remote Sensing and Image Interpretation.* 2nd. ed., New York: John Wiley, 1987.

[41] S. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, no. 7, pp. 674–693, 1989.

[42] D. Marr, *Vision.* San Francisco: Freeman, 1982.

[43] B. H. McCormick, T. A. DeFanti and M. D. Brown (Edts.), *Visualization in Scientific Computing*. ACM Computer Graphics (special issue), vol. 21, no. 6, 1987.

[44] S. Muraki, "Volumetric shape description of range data using 'blobby model'," *Computer Graphics,* vol. 25, no. 4, pp. 227–235, 1991.

[45] ————, "Multiscale 3D edge representation of volume data by a DOG wavelet," *ACM Symposium on Volume Visualization*, 1994.

[46] H. Murase and S. K. Nayar, "Learning object models from appearance," *Proc. Nat. Conf. Art. Intell., AAAI-93*, pp. 836–843, 1993.

[47] G. M. Nielson, "Scattered data modelling," *IEEE CG&A,* vol. 13, no. 1, pp. 60–70, 1993.

[48] ————, "Visualization in scientific and engineering computation," *IEEE Computer*, vol. 9, pp. 58–66, 1991.

[49] A. Pentland, "Equilibrium and interpolation solutions using wavelet bases," *Proc. of the CGI '91*, pp. 507–523, New York: Springer, 1991.

[50] L. Rosenblum, R. Earnshaw, J. Encarnacao, H. Hagen, A. Kaufman, S. Klimenko, G. Nielson, F. Post and D. Thalmann: *Scientific Visualization. Advances and Challenges.* New York: Academic Press, 1994.

[51] P. Schroeder and P. Hanrahan, "Wavelet Radiosity", *ACM Computer Graphics, Proc. SIGGRAPH '93,* pp. 221–230, 1993

[52] E. P. Simoncelli and E. H. Adelson, "Non–separable extensions of quadrature mirror filters to multiple dimensions," *Proc. IEEE*, vol. 78, no. 4, pp. 652–664, 1990.

[53] T. Tanaka and F. Kishimo, "Recordering and visualizing complex shape from range data", *Proc. CGI'92*, Tokyo: Springer, pp. 331–348, 1992.

[54] T. Totsuka and M. Levoy, "Frequency domain volume rendering," *ACM Computer Graphics, Proc. SIGGRAPH '93,* pp. 271–278, 1993.

[55] M. Unser and M. Eden, "Multiresolution feature extraction and selection for texture segmentation," *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. 11, no. 7, pp. 717 – 728, 1989.

[56] M. Unser, A. Aldroubi and M. Eden, "A family of polynomial spline wavelet transforms," *Signal Processing*, vol. 30, pp. 141–162, 1993.

[57] M. Vannier, T. Pilgram, G. Bhatia and B. Brunsden, "Facial surface scanner," *IEEE CG&A*, vol. 11, no.6, pp.72–80, 1991.

[58] M. Vetterli, "Wavelets and filter banks: Theory and design," *IEEE Trans. Signal Proc.*, vol. 40, pp. 2207 – 2231, 1992.

[59] Y. Watanabe and Y. Suenaga, "Synchronized acquisition of three-dimensional range and color data and its applications," *Proc. CGI 91*, pp. 655–665, Berlin: Springer, 1991.

[60] R. Westermann, "A multiresolution framework for volume rendering," *ACM Symposium on Volume Visualization*, 1994.

[61] G. Wyszecki and W. Stiles, *Color Science. Concepts and Methods, Quantitative Data and Formulae*. 2nd ed., New York: John Wiley & Sons, 1991.

[62] N. Yokoya and M.D. Levine: "Range image segmentation based on differential geometry: A hybrid approach," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, no. 6, pp.643–649, 1989.

[63] F. Young and P. Rheingans, "Visualizing structure in high-dimensional multivariate data," *IBM Journal of Research and Development*, vol. 35, no. 1/2, pp. 97–107, 1991.

Appendix A: Construction of Gabor Wavelets using a σ–based boundary criterion

Our construction rule for Gabor wavelets in the frequency plane bases on the geometric problem illustrated in fig. 22. The basic idea is to locate the single 2D Gaussians in the u,v–plane in a way that their $\sigma_{u,v}$–boundaries touch each other. In this figure, the boundaries are represented by circles for different scales, positions and orientations. We furthermore assume circular envelopes, with $\sigma_u = \sigma_{v.}$. Hence, we have to fill a plane with non–overlapping circles at a given resolution for an orientation staggering $\Delta\Theta$ and starting at a lower frequency bound ω_0. This scheme provides an overlap of the Gabors at their σ–boundaries and yields a compromise between a good representation of the data spectrum on the one side and compactness on the other side. It should be noted that due to the variety of parameters to be tuned, the final layout of Gabor pyramids scatters much, depending on the application.

Fig. 22. Illustration of the fitting procedure of the Gabor functions into the frequency plane for $\Delta\Theta = \pi/4$ for the first 3 iterations. The circles indicate Gaussians at different scale, position and orientation.

Let $m = 0,..,M-1$ count for the iteration steps in the pyramid and $i = 0,..,I-1=\lfloor 2\pi/\Delta\Theta \rfloor - 1$ indicate the angular rotation, the Gabor function $G_{m,i}(u,v)$ in frequency plane can be expressed as in eq. 7. The immediate computation of the convolution products allows us to neglect spatial translations. σ_m denotes the effective width of the envelope:

$$G_{m,i}(u,v) = e^{\frac{-\pi}{\sigma_m^2}\left(\left(u-u_{m,i}\right)^2+\left(v-v_{m,i}\right)^2\right)} \tag{41}$$

The center coordinates $(u_{m,i}, v_{m,i})$ of the Gaussian follow the geometric relationships of fig. 22 and are computed as:

$$u_{m,i} = a^m \omega_0 \cos\left(\frac{2i + (m+1) \bmod 2}{2} \Delta\Theta\right) \tag{42}$$

$$v_{m,i} = a^m \omega_0 \sin\left(\frac{2\ i + (m + 1) \bmod 2}{2} \Delta\Theta\right) \tag{43}$$

The effective width σ_m of the Gaussian is scaled by

$$\sigma_m = a^m \sigma_0 = a^m \omega_0 \sin\left(\frac{\Delta\Theta}{2}\right) \tag{44}$$

The center frequencies are obtained with

$$\omega_m = a^m \omega_0 \tag{45}$$

Following fig. 22 σ_0 relates to ω_0 with:

$$\sigma_0 = \omega_0 \sin\left(\frac{\Delta\Theta}{2}\right) \tag{46}$$

In order to compute the scaling factor a as a function of ω_0 and $\Delta\Theta$, we introduce the following relationships:

$$\left(\sigma_0 + \sigma_1\right)^2 = \sigma_0^2 + \left(\omega_1 - d_x\right)^2 \tag{47}$$

with

$$d_x = \omega_0 \cos\left(\frac{\Delta\Theta}{2}\right) \tag{48}$$

Inserting the upper equations results in a 2nd order polynomial equation for the recursive computation of σ_{m+1} from σ_m, where a represents the scaling factor and m the iteration index. The solutions are:

$$\sigma_{m+1} = \sigma_m \underbrace{\left[\left[\frac{1}{\cos\left(\frac{\Delta\Theta}{2}\right)} + \tan\left(\frac{\Delta\Theta}{2}\right)^2 \right] \pm \sqrt{\left[\frac{1}{\cos\left(\frac{\Delta\Theta}{2}\right)} + \tan\left(\frac{\Delta\Theta}{2}\right)^2 \right]^2 - 1} \right]}_{a} \quad (49)$$

a only depends on the angular increment $\Delta\Theta$.

We finally yield the solution for the recoursive definitions of the wavelet pyramid, through eqns. 41 – 44:

Note, that one distinct additional function G_0 is included in our pyramid, representing the DC–component of the data, explicitly:

$$G_0(u,v) = e^{-\pi(\omega_0 - \sigma_0)^2 (u^2 + v^2)} \quad (50)$$

The overall weighting function $G_{all}(u,v)$ results by superimposing all the bases we defined as:

$$G_{all}(u,v) = e^{-\pi(\omega_0 - \sigma_0)^2 (u^2 + v^2)} + \sum_{i=0}^{\lfloor 2\pi/\Delta\Theta \rfloor - 1} \sum_{m=0}^{M-1} e^{\frac{-\pi}{a^{2m}\sigma_0^2}\left(\left(u - u_{m,i}\right)^2 + \left(v - v_{m,i}\right)^2 \right)} \quad (51)$$

Any straightforward expansion of the data by superimposing the convolution products of all wavelets will render a modified spectrum and does not hold the energy equivalent of Parseval's theorem. Due to the symmetries, it is sufficient to use only one half of the pyramid, for instance with i = 1,..,$\pi/\Delta\Theta$ for data analysis.

This scheme can be extended to tune the localization of the wavelets for specific purposes by keeping the centers of the Gaussians but increasing or decreasing its overlap. In this case the important scaling property and the computation of a are both preserved.

Let τ denote the overlap of the Gaussians in frequency space, we apply the same construction rules for our wavelets and obtain their center coordinates in accordance to eqns. 42–43. The effective width σ'_m however will be computed to:

$$\sigma'_m = \tau\ \sigma_m = \tau\ a^m\ \sigma_0 \tag{52}$$

$0 \leq \tau \leq 1$ shrinks the Gaussian and increases the frequency resolution, whereas $\tau > 1$ blows it up and results in a wavelet, which is localized more accurately in spatial domain. It is clear that τ gives an additional degree of freedom and allows for a further fine tuning. The examples in this paper have been computed with τ=2.