Affine Real-Time Face Tracking using Gabor Wavelet Networks

Volker Krüger, Alexander Happe and Gerald Sommer
Computer Science Institute, Christian-Albrechts University Kiel
Preußerstr. 1-9, 24105 Kiel, Germany
Tel: ++49-431-560496, FAX: ++49-431-560481
email: vok@ks.informatik.uni-kiel.de

Abstract

In this article we present a method for visual face tracking that is based on a wavelet representation of a face template. The wavelet representation allows arbitrary affine deformations of the facial image, it allows to generalize from an individual face template to a rather general face template and it allows to adapt the computational needs of the tracking algorithm to the computational resources available. The method presented was implemented on a Linux Pentium 450 MHz and runs off-line with 25 Hz and on-line, using an active camera mount, with 22 Hz. We present experimentals results on off-line tests on several common image sequences including the salesman-sequence as well as on on-line tests.

1 Introduction

This paper addresses the issue of face tracking. Face tracking in real-time (RT) (25 Hz) is of major importance for many applications including Human-Computer-Interfaces (HCI), surveillance applications, teleconferencing or teleteaching. Also applications such as gesture- and gaze detection are often stated applications that heavily depend on precise tracking algorithms. Yet, the issue of face tracking is far from being solved satisfactorily. Many tracking systems use color as a clue for tracking but they track imprecisely and they are not capable of distinguishing between faces and non-faces [1; 5; 8]. Other tracking systems use a previously given template (gray value, active contour, etc), while allowing affine variations of the facial image. These systems track precisely as shown in the excellent work of [3], but the templates are either of individual persons [3] or are computationally expensive [2; 4] and therefore slow so that tracking is not in RT. In [6] a system is presented that is able to track faces independent of face orientation and gesture. The system uses a wavelet jet bunch-graph approach and tracks with less than 1 fps.

In this paper we present an approach for RT face tracking

- that allows arbitrary affine deformations of the facial image in order to compensate for different poses,
- that is robust to homogenious illumination changes,
- that is efficient, fast and surprisingly robust.

For tracking, we need a gray value face template of the person who should be tracked. However, for the actual tracking, we do not use this ordinary gray value template. Instead, we approximate the discrete face template with a linear combination of continuous 2D odd-Gabor wavelet functions. For this, the Gabor wavelets are optimized with respect to their 2D parameters position, scale and orientation. Using this *Gabor wavelet template* for tracking has the following major advantages:

1. The Gabor wavelet template (GWT) is a discretizised version of a continuous function for which continuous derivations can be calculated.
2. The Gabor wavelet template can be deformed arbitraily, i.e. can be continuously translated, rotated, scaled and sheared.
3. It is well known that the precision of a wavelet representation depends on the number of used basis functions. Using a GWT allows the user to decide on using a rather low-frequential general template that works well on different individuals or a rather precise template that works well only on the individual person.
4. The computer power needed for tracking will depend upon the number of wavelets to evaluate. A tracking program may chose the number dynamically with respect to the available computer power.

By exploiting these advantages, we establish real-time tracking of arbitrary faces by optimizing the affine parameters of the entire wavelet representation at each image frame while being able to dynamically adapt the number of used wavelets and with this the computing resources needed.

In section 2 we give a short introduction to Gabor wavelets Networks. In section 3, the tracking algorithm is presented. In section 4 we give experimental results and conclude with final remarks in section 5.

2 Introduction to Gabor Wavelet Networks

In order to find a *Gabor wavelet template* (GWT) that approximates a given face template I, we use a Gabor Wavelet Network (GWN). A GWN is defined as follows:

Definition: Let $\psi_{\mathbf{n}_i}$, $i = 1, \ldots, N$ be a set of odd Gabor wavelets that are given as

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

with $\mathbf{n} = (c_x, c_y, \theta, s_x, s_y)^T$. Let I be an image with DC-value $\mathrm{dc}(I)$. Let $w_i \in \mathbb{R}$ be a set of weights and $\mathbf{n}_i \in \mathbb{R}^5$ be the parameter vectors of the wavelets $\psi_{\mathbf{n}_i}$, chosen such that the energy function

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

is minimized with respect to the weights w_i and the wavelet parameter vectors $\mathbf{n}_i$. The two vectors

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

define then the *Gabor wavelet network* $(\Psi, \mathbf{w})$ for image I.

In other words, a Gabor wavelet network for an image I is defined to be an N-dimensional vector of weights w_i and an N-dimensional vector of Gabor wavelets $\psi_{\mathbf{n}_i}$, where the the weights w_i and the parameter vectors $\mathbf{n}_i$ are chosen such that the weighted sum of Gabor wavelets $\psi_{\mathbf{n}_i}$ approximates the discrete gray value image I optimally. The *discrete* GWT $\hat{I}$ of image I is according to eq. (2) given by

$$\hat{I} = \text{Ш}\Big(\textstyle\sum_i w_i \psi_{\mathbf{n}_i} + dc(I)\Big), \quad (3)$$

where $\text{Ш}(x)$ is the usually obmitted sampling function. An

Figure 1. A GWN with $N = 52$ wavelets is optimized for image I (very left) according to eq. (2). The second image (from left) shows the corresponding GWT $\hat{I}$. The third image shows the GWT $\hat{I}_{16}$, made up of just the 16 largest wavelets, the fourth image shows their positions.

example can be seen in fig. 1: $N = 52$ wavelets are distributed over the inner face region of the very left image I by the minimization formula (2). The corresponding GWT $\hat{I}$ is shown in the second image (from left). The third image shows the GWT $\hat{I}_{16}$, made up of the 16 largest wavelets and the forth image shows their positions.

Minimizing equation (2) is crucial, because finding a global minimum is an inefficient task. In order to find a GWN $(\Psi, \mathbf{w})$ for a discrete gray value image I, we use the Levenberg-Marquard gradient descent method [7]. The Levenberg-Marquard method optimizes the parameters of each single Gabor wavelet with respect to the energy function (2). This method might get stuck in local minima and a careful selection of the initial parameters is therefore important. We use prior knowledge about significant image features to allow a task oriented optimization.

3 Affine Face Tracking in Image Sequences

In the preceding section we have given an introduction to wavelet networks. Now, we are going to describe in the next subsection 3.1 how a GWT, taken as a face template, can be used to precisely locate the face independently of perspective deformations and illumination. In subsection 3.2 we will extend this approach to allow a repeated localization of the face in image sequences and we will show that the localization is speeded up considerably because image changes are small from frame to frame.

3.1 Re-parameterizing a Gabor Wavelet Network in Single Images

In this subsection we will demonstrate how a GWT can be used to precisely localize a face. For this, we first assume that the face is given by the GWN of its gray value template I and by the corresponding GWT $\hat{I}$. We further assume that the approximate position, scale and orientation of the searched face in image J is known (for a justification, see below). We then re-parameterize (translate, rotate, dilate and sheare) the GWN so that its wavelets are finally positioned on the same facial features in the new test image J as they were in the original gray value template I. An example for this can be seen in fig. 1, where in the very right image the original positions of the first 16 Gabor wavelets are marked and in fig. 2, where in new images the positions of the first 16 Gabor wavelet of the *re-parameterized* GWN are marked. In fig. 2, the corresponding re-parameterized GWT can be seen.

Image distortions of a planar object that is viewed under orthographic projection is described by six parameters: translation c_x, c_y, rotation θ, dilation s_x, s_y and s_{xy}. For larger distances a face can be well understood as a planar object. What we therefore need to do for a correct re-parameterization of the GWT is to find the correct affine parameters $c_x, c_y, \theta, s_x, s_y, s_{xy}$.

Re-parameterizing a GWN, i.e. finding the correct parameters, is established by using a *superwavelet* [9].

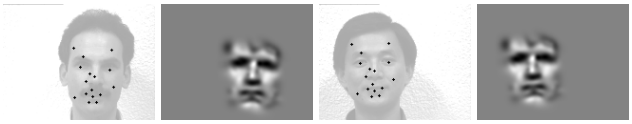

Figure 2. The images show the positions of each of the first 16 wavelets after re-parameterization of the GWN (left) and the corresponding GWT (right). The re-parameterized GWTs show the same orientation, position and size as the ones they were repositioned on.

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

$$\Psi_{\mathbf{n}}(\mathbf{x}) = \sum_i w_i \psi_{\mathbf{n}_i}(\mathbf{S}\mathbf{R}(\mathbf{x} - \mathbf{c})) \,, \qquad (4)$$

where the parameters of vector $\mathbf{n}$ of the GSW Ψ define the dilation matrix $\mathbf{S}$, the rotation matrix $\mathbf{R}$ and the translation vector $\mathbf{c}$. To reflect the re-parameterization, the corresponding GWT is denoted by $\hat{I}^n$.

A Gabor superwavelet $\Psi_{\mathbf{n}}$ is again a wavelet that has the typical wavelet parameters dilation s_x, s_y, translation c_x, c_y and rotation θ. Therefore, we can handle it in the same way as we handled each single Gabor wavelet in the previous section, and we may optimize its parameters $\mathbf{n}$ with respect to the energy function E that, in this case, reads:

$$E = \min_{\mathbf{n}} \|I - \Psi_{\mathbf{n}}\|_2^2 \qquad (5)$$

Even though eq. (4) looks similar to the definition of a GWT in eq. (3) we want to point out that eq. (4) refers to a continuous wavelet function, whereas a GWT is a discrete gray value image that is derived from a GWN, or GSW, respectively.

The degrees of freedom of a wavelet only allow translation, dilation and rotation. But it is straight forward to include also shearing and thus allow any affine deformation of $\Psi_{\mathbf{n}}$. For this, we enhance the parameter vector $\mathbf{n} \in \mathbb{R}^6$ to a six dimensional vector

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

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

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

we become able to deform the GSW $\Psi_{\mathbf{n}}$ affinely.

In order to minimize (5) and to find the optimal parameter vector $\mathbf{n} \in \mathbb{R}^6$ we may elegantly use the same Levenberg-Marquard algorithm as in the preceding section. In several experiments we have found that the initialization that has to be supplied to the gradient decent method may be within the range of approximately ± 10 px in position, $\pm 20\%$ in scale and $\pm 10°$ in orientation (see below for further comments). An example of the optimization process can be seen in fig. 3: Shown are the initial values of $\mathbf{n}$, the values after 2 and 4 optimization cycles and the final values after 8 cycles, each marked with the white square. The square refers to the inner face region. Its center position marks the center position of the corresponding GSW. The GSW used in fig. 3 is derived from the person in fig. 1. It uses the first 16 wavelets only and its GWT looks like $\hat{I}_{16}$ of fig. 1.

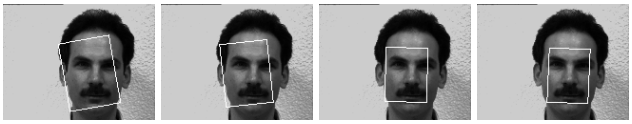

Figure 3. The images show the 1st, the 2th, the 4th and the 8th (final) step of the gradient descent method optimizing the parameters of a GSW. The top left image shows the initial values with 10 px. off from the true position, rotated by $10°$ and scaled by 20%. The bottom right image shows the final result. As GWT, $\hat{I}_{16}$ of figure 1 was used.

3.2 Re-parameterizing GWNs in Image Sequences for Affine Face Tracking

The technique of re-parameterizing a GSW with respect to the energy function (5), as it was explained in the preceding subsection, can also be applied to image sequences. This enables us to track affinely. For this, (5) may be rewritten to

$$E = \min_{\mathbf{n}_t} \|J_t - \Psi_{\mathbf{n}_t}\|_2^2 \,. \qquad (6)$$

so that for each frame J_t at time step t the GSW $\Psi_{\mathbf{n}_t}$ is optimized with respect to the energy function (6). As initial values for the optimization the parameters $\mathbf{n}_{t-1}$ from the preceding frame are used. These initial values were in our experiments always good enough that the optimization procedure always converged quickly (see section 4).

Initial values $\mathbf{n}_0$ for the very first frame I_0 are derived from the color blob information. A color blob is given by its mean value and its standard deviation. The mean value gives a clue about the position and a first clue about the scale and the orientation can be calculated from the standard deviation matrix. For the test sequence of fig. 4, we have chosen $\mathbf{n}_0$ by hand because the sequences is a gray scale sequence.

The number of wavelets that make up the GSW can be adapted: The maximum number is given by the number

N of wavelets in the GWN, but we are free to use less wavelets. Each wavelet of the GSW has to be evaluated during the re-parameterization process, so that using less wavelets results in a respective speedup. Techniques for affine motion prediction have not yet been incorporated into the tracker. Such techniques should result in a significant speedup.

4 Experiments

For a GWT of 40×40 pixels we have found in several experiments that the initialization values of $\mathbf{n}$ may vary from the correct values by approx. ± 10 px in x and y direction, by approx. 20% in scale and by approx. $\pm 10°$ in rotation (see fig. 3).

We have further tested the positioning procedure on the Yale face database. This database consists of 15 different individuals, showing eights different facial expressions, the faces are approximately all of the same size. The GWT $\hat{I}_1 6$ in fig. 1, that was made up by the first 16 Gabor wavelets of the corresponding GWN can be considered as a rather general face template. Using this GWT, the positioning procedure converged correctly on 13 individuals (independent of expression) by just giving the approximate image center as initial values. This shows two things:

1. It shows, that the reposition algorithm is quite stable with respect to its initial values.
2. It shows, that the wavelet net template is not fixed to one individual and that it is sufficiently general.

For face tracking, using color blob information as initial values for $\mathbf{n}$ seems to be precise enough. We have tested the face tracker within our active camera mount as well as on several sequences, including the salesman sequence (fig 4).

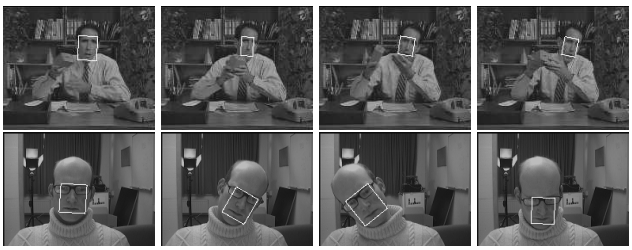

Figure 4. The images show from left to right frame 11, frame 50, frame 120 and frame 137 (top) and some active camera images (bottom).

Off-line tracking is done with 25 Hz, on-line tracking is done with 22 Hz (the difference is due to the frame grabbing). During tracking, since we track with 22-25 Hz, successive frames are sufficiently similar so that the gradient descent method never needed more than two cycles for each time step until reaching the minimum. The GWT we used in all our experiments was made up by the largest 10 wavelets of the corresponding GWN that was trained on the respective person. Experiments were carried out on a 450 MHz Linux Pentium.

5 Conclusion

As the major contribution of this work we presented a novel approach for real-time face tracking where tracking is done with a Gabor wavelet template. The GWT has the advantage that it can be arbitraily translated, rotated, scaled and sheared. This is because the GWT is given by a discrete linear combination of *continuous* Gabor wavelets whose weights and wavelets are given by its corresponding GWN or GSW, respectively. A further great advantage that comes with the continuity is that we can use a fast gradient decent method to estimate the affine parameters. A next great advantage of the GWT is that it can be made up of different numbers of Gabor wavelets of the corresponding GWN which means that the GWT can describe the face of a template I in almost any desired precision. This allows an application of the GWT also to different individuals.

We have exploited all these advantages and have designed a tracking system that is able to work in real time (22-25 Hz), that is able to cope with perspective deformations and that is able, by changing the number of used wavelets, to track either only a special individual or almost any person.

References

[1] S. Birchfield. Elliptical head tracking using intensity gradients and color histograms. In *IEEE Conf. Computer Vision and Pattern Recognition, CVPR*, pages 232–237, 1998.

[2] F. d.l. Torre, S. Gong, and S. McKenna. View-based adaptive affine tracking. In *Proc. Fifth European Conference on Computer Vision*, volume 1, pages 828–824, Freiburg, Germany, June 1-5, 1998.

[3] G. Hager and P. Belhumeur. Efficient region tracking with parametric models of geometry and illumination. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 20(10):1025–1039, 1998.

[4] M. Isard and A. Blake. Condensation – conditional density propagation for visual tracking. *International Journal of Computer Vision*, 1998.

[5] V. Krüger, R. Herpers, K. Daniilidis, and G. Sommer. Teleconferencing using an attentive camera system. In *Int. Conf. on Audio- and Video-based Biometric Person Authentication*, pages 142–147, 1999.

[6] T. Maurer and C. v.d. Malsburg. Tracking and learning graphs on image sequences of faces. In *Int. Conf. on Automatic Face- and Gesture-Recognition*, pages 176–181, Killington, Vermont, USA, Oct. 14-16, 1996.

[7] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling. *Numerical Recipes, The Art of Scientific Computing*. Cambridge University Press, Cambridge, UK, 1986.

[8] Y. Raja, J. McKenna, and S. Gong. Tracking and segmenting people in varying lighting conditions using color. In *Int. Conf. on Automatic Face- and Gesture-Recognition*, pages 228–233, Nara, Japan, April 14-16, 1998.

[9] H. Szu, B. Telfer, and S. Kadambe. Neural network adaptive wavelets for signal representation and classification. *Optical Engineering*, 31(9):1907–1961, 1992.