Learning Feature Transformations to Recognize Faces Rotated in Depth

Thomas Maurer*, Christoph von der Malsburg*+

*Ruhr-Universität Bochum
Institut für Neuroinformatik
D-44780 Bochum, Germany
E-mail: maurer@neuroinformatik.ruhr-uni-bochum.de
Phone: +49-234-700-7996

+University of Southern California
Dept. of Computer Science and Section for Neurobiology
Los Angeles, CA 90089, USA

Abstract

We present a method for recognizing objects (faces) on the basis of just one stored view, in spite of rotation in depth. The method is not based on the construction of a three-dimensional model for the object. Our recognition results represent a significant improvement over a previous system developed in our laboratory. We achieve this with the help of a simple assumption about the transformation of local feature vectors with rotation in depth. The parameters of this transformation are learned on training examples.

1 Introduction

It is known from biology that objects are seen under two aspects [UM]: the ‘Where’ and the ‘What’ aspect. ‘Where’ means localization in space and perspective, ‘What’ means classification and identification. In technical systems the last task often turns out to be the more difficult one which needs the highest accuracy (e.g., it is less difficult to find a face in an image than to recognize it). Thus, visual features are needed which are invariant under perspective changes or can be transformed. Invariant features seem attractive, avoiding computational effort like mathematical transformations. But invariance always means loss of information because a feature detector is insensitive to the input change to which it is invariant. To compensate, many different detectors with different invariances have to be combined. This seems biologically plausible but is suited better for parallel than sequential hardware. Here, we have chosen to use localized visual features which can be transformed according to perspective changes in a homogenous style. We will describe our transformation method and apply it to the recognition of faces rotated in depth. The paper has three parts: The following section is a very short description of our face recognition system without feature transformation. The second

---

1Supported by grants from the German Federal Ministry for Science and Technology (413-5839-01 IN 101 B9) and from the US Army Research Lab (01/93/K-0109).
2 Description of the face recognition system

As basic visual feature we use a local image descriptor in the form of a vector called ‘jet’ [LVB]. Each component of a jet is the filter response as the result of the convolution of a Gabor wavelet of specific frequency and orientation with the image, extracted at a definite point \( x \). We are employing Gabor wavelets of 5 different frequencies and 8 different orientations for a total of 40 complex components. Such a jet describes the area surrounding \( x \). Many jets taken at different positions form a graph which describes an object in the image. To compare jets and graphs, similarity functions are defined: The normalized dot product of two jets yields their similarity, the sum over jet similarities being the similarity of graphs of identical geometry. But distortions of the grid can also be taken into account to compute the graph similarity [LVB]. Features and similarities defined in this way are robust against changes in illumination and contrast. In order to create an appropriate face graph for a gallery of 50-80 frontal faces of equal size, we place the nodes by hand, e.g., at the center of the eyes or at the tip of the nose. There are typically 40-50 nodes forming a face graph. Such a gallery contains the general face knowledge for frontal faces. For a new frontal face of the same size these nodes of the graph can be found automatically [WFK], even if this person is not in the gallery. After finding the face graph it can be compared with others. In order to achieve a similar scale for new images, this procedure of automatically finding the nodes is repeated twice: in the first stage only few nodes are used to find the face in the image, but graphs of different sizes are tried and that one with the largest similarity compared to the stored gfk is used to cut out the face. In the second stage this zoomed image is treated as new input and the automatic finding of the nodes is repeated with a finer grid and a larger gfk to place the nodes with high accuracy. The whole procedure can be repeated in the same manner for face images of other poses, say profile or half profile (faces rotated by approximately 45 degrees in depth). Faces of the same pose can be compared without problems, but comparing face images of different poses remains difficult. The graphs of all poses can be defined in such a way that they share many nodes, i.e., there are always eye nodes, nose nodes etc., and graph converters can be constructed which allow comparing graphs of different poses on the subset of common nodes. However, as long as the jets, the visual features extracted from a single 2D-view, are not transformed, results are not quite satisfactory [WFK].

3 Feature Transformation

For simplicity we assume the local object shape near a single node to be flat. The orientation of this area, i.e., its normal vector, has somehow to be determined. Once this normal vector and the geometric transformation of the object (e.g., rotation angle of the face) are known, the jet at this node can be transformed analytically. One could argue that this assumption of flat areas is grossly violated at nodes near the tip of the nose or the eyes. However, many nodes in a face graph can be approximately described this way and even if the area is not locally flat the system tries to find an effective area with an effective normal vector to describe the transformation as accurately as possible. This will become clearer below. Another difficulty is that with increasing rotation angle between two views of a face, fewer nodes can be used for comparing the pictures because of reduced overlap of the graphs. This reduction in mutual information will make face recognition impossible beyond a certain rotation angle, as is borne out by psychophysical experiments [KBC]. The only exception may be the comparison of left and right profile views, which are similar due to facial symmetry.
3.1 Geometric transformation of a single jet

At first we will show how to transform a single jet describing the surroundings at a single node on the surface of an object. Therefore, we assume the surface to be locally flat, as discussed above. Then the image coordinates $x = (x_1, x_2)$ with respect to the coordinate center at corresponding points (see figure 1) transform according to $x' = A \ x$ with $A$ in general being determined by the scale factor and the rotation angles in plane and in depth. We will restrict ourselves to rotation in depth only. If $\phi$ and $\theta$ are the horizontal and vertical normal angles—see figure 2—the matrix $A$ is easily determined with the help of the transformed unit vectors:

$$
A = \begin{bmatrix}
\cos \phi & \sin \theta \sin \phi \\
0 & \cos \theta 
\end{bmatrix}
$$

If the planar object does not lie in the camera plane initially but at $(\phi_1, \theta_1)$, then it can be imagined being transformed from the camera plane, resulting in

$$
x' = A(\phi_2, \theta_2) \ A^{-1}(\phi_1, \theta_1) \ x.
$$

This means that, if we know the orientation or normal vector of the flat piece of surface in the two images we can compute the corresponding image coordinate’s transformation matrix $A$. Looking again at figure 1, we now turn to the question: how does the jet located at the origin of the coordinate system transform from the left image to the right? We will not restrict ourselves to rotation in depth here but solve the general problem with arbitrary $A$. The jet component $j_k$ with wave vector $k$ was defined as

Figure 1: How do the image coordinates transform in a coordinate system whose origin is shifted to the corresponding object point?

Figure 2: A flat object initially lies in the $xy$-plane, is rotated around the $y$-axis by angle $\phi$ and around the $x'$-axis by angle $\theta$. If the camera is placed at a positive $z$-coordinate, how do image coordinates transform if the flat object is rotated from the unprimed to the primed coordinates?
\[
\psi_k(\mathbf{x}) = \frac{k^2}{\sigma^2} e^{-\frac{k^2}{2\sigma^2}} \left\{ e^{i\mathbf{k} \cdot \mathbf{x}} - e^{-\frac{k^2}{2}} \right\}.
\]

The integral can be taken to infinity because of the finite support of \(\psi_k(\mathbf{x})\). The corresponding jet component in the other image containing the transformed planar object is

\[
j'_k = \int d\mathbf{x} I(\mathbf{A}^{-1} \mathbf{x}) \psi_k(\mathbf{x}) \quad \Rightarrow \quad j'_k = \int d\mathbf{x} I(\mathbf{x}) \psi_k(\mathbf{A} \mathbf{x}) \det(\mathbf{A}).
\]

We make the Ansatz

\[
\psi_k(\mathbf{A} \mathbf{x}) \det(\mathbf{A}) \approx \sum_{k'} \alpha_{kk'}(\mathbf{A}) \psi_{k'}(\mathbf{x}),
\]

although this can only be approximated. The accuracy of this Ansatz will increase with sampling density in \(k\)-space. Multiplying eq. (6) with \(\psi_{k'}(\mathbf{x})\) and integrating leads to a system of linear equations determining the \(\alpha_{kk'}(\mathbf{A})\). All integrals can be solved analytically; details can be found in [MauVdM95]. Once the \(\alpha_{kk'}(\mathbf{A})\) are determined, the jet can be transformed according to

\[
j' = C^T \mathbf{j}.
\]

To summarize, given the orientations or normal vectors of a locally flat piece of surface seen under two different perspectives, we can compute the transformation matrix \(C\) to transform the jet—our local visual feature vector—from one perspective into the other.

## 4 Applying feature transformation to face recognition

### 4.1 Learning the transformation parameters

To learn the transformation between two poses we need a training gallery of 50-80 persons with an image of each of the two poses: for instance, one frontal view and one half profile view with an average head rotation angle of approximately 45°. On these faces the flexible grids are placed automatically as described in section 2, determining the corresponding points as left and right eyes, corners of the mouth, lobes of the ears etc. To determine the initially unknown normal vectors at each node, we simply test all possible orientations (in steps of 5° horizontally and vertically). For every normal angle pair we rotate the jet at this node by 45° for all half faces and compare to the corresponding jet in the frontal view. We search for the normal angles which maximize the recognition performance \(E\) at this node, i.e., which result in high feature similarity \(s_{ii}\) for identical persons and low similarities \(s_{ij}\) for different persons:

\[
E = \frac{1}{N^2} \sum_i \sum_j (s_{ii} - s_{ij})
\]

This procedure is repeated for all nodes of the flexible grids visible in both views; the others are discarded. Thereby we get an average set of effective normal vectors which determine—the transformation of the jets between the two poses. Let us remark here that we have only two free parameters (the two normal angles) for 50-80 data points (the jets of all persons at this node). As a consequence, there will be no generalization problem, and this was completely confirmed by our tests. In addition, the time needed for learning is quite reasonable: it takes 20-30 min. on a Sun Sparc-20. The time to rotate face jets after the normals have been learned is negligible compared to the time needed to find the face in a new image and the nodes in the face
Figure 3: Typical half profile and frontal faces. To learn the effective normal angles for the left eye, the left eyes of all half poses are compared with the left eyes of all frontal poses.

which takes 15-20 sec., most of this time is needed for the two wavelet-transformations which have not been optimized yet.

4.2 Recognition Results

To test our method we used two databases: the first one was prepared by the US Army Research Laboratory (ARL), the second one is our Bochum face gallery. The Bochum gallery contains 110 persons with three views: 0°, 15°, and 30°. We will only compare the 30° pictures to the 0° pictures here. The ARL database is more difficult: the head rotation angles are much larger, up to 80°. We tried to discard the extreme ones and used an average rotation angle of 45°, but there are still enough faces with larger rotation angle. Figure 3 shows typical images from the ARL database.

Out of 160 ARL pose pairs we randomly selected 70 for training and 90 for testing. After learning the normal vectors on the training set we got the following results on the test set when comparing half profile faces to the frontal gallery:

<table>
<thead>
<tr>
<th>ARL (45°)</th>
<th>without rotation</th>
<th>rotating h to f</th>
<th>rotating f to h</th>
</tr>
</thead>
<tbody>
<tr>
<td>correct recognized (rank 1)</td>
<td>36%</td>
<td>50%</td>
<td>53%</td>
</tr>
<tr>
<td>among best 5% (ranks 1–4)</td>
<td>56%</td>
<td>73%</td>
<td>73%</td>
</tr>
</tbody>
</table>

For the Bochum gallery with 110 persons we got:

<table>
<thead>
<tr>
<th>Bochum (30°)</th>
<th>without rotation</th>
<th>rotating h to f</th>
</tr>
</thead>
<tbody>
<tr>
<td>correct recognized (rank 1)</td>
<td>88%</td>
<td>96%</td>
</tr>
</tbody>
</table>

Obviously the ARL database is much harder for this task than the Bochum gallery. Because we had very high recognition rates on the ARL database by comparing galleries of only frontal views (98% on 300 persons), we conclude that this is mainly due to the larger head rotation angles.
5 Conclusions

The method presented here is clearly able to improve recognition performance on face galleries rotated in depth. On the other hand, for the large head rotation angles recognition rates are still far away from being near 100%, in contrast to frontal face recognition. But we now think that this is a more general problem: Comparing frontal to profile views does not seem to make much sense, the frontal view doesn’t contain information about the back of the head or how crooked the nose is. It is also difficult for human beings to compare frontal to profile images—especially if they are not familiar with the person. On the other hand, if frontal and profile views are known, it should be no problem to recognize this person with our method independent of the head rotation angle, and with high performance. This will be our next project.

References


