Head Pose Determination from One Image Using a Generic Model

Ikuko Shimizu[1,3] Zhengyou Zhang[2,3] Shigeru Akamatsu[3] Koichiro Deguchi[1]

[1] Faculty of Engineering, University of Tokyo, 7-3-1 Hongo, Bynkyo-ku, Tokyo 113, Japan
[2] **INRIA**, 2004 route des Lucioles, BP 93, F-06902 Sophia-Antipolis Cedex, France
[3] **ATR** HIP, 2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-02, Japan
e-mail: `ikuko@meip7.t.u-tokyo.ac.jp`

Abstract

We present a new method for determining the pose of a human head from its 2D image. It does not use any artificial markers put on a face. The basic idea is to use a generic model of a human head, which accounts for variation in shape and facial expression. Particularly, a set of 3D curves are used to model the contours of eyes, lips, and eyebrows. A technique called Iterative Closest Curve matching *(ICC) is proposed, which aims at recovering the pose by iteratively minimizing the distances between the projected model curves and their closest image curves. Because curves contain richer information (such as curvature and length) than points, ICC is both more robust and more efficient than the well-known iterative closest point matching techniques (ICP). Furthermore, the image can be taken by a camera with unknown internal parameters, which can be recovered by our technique thanks to the 3D model. Preliminary experiments show that the proposed technique is promising and that an accurate pose estimate can be obtained from just one image with a generic head model.*

1. Introduction

This paper deals with techniques for estimating the pose of a human head using its 2D image taken by a camera. They are useful for the realization of a new man-machine interface. We present a new method for the accurate estimation of a head pose from only one 2D image using a 3D model of human heads. By a 3D model with characteristic curves, our method does not use any makers on the face and uses an arbitrary camera with unknown parameters to take images.

Several methods have been proposed for head pose estimation which detect facial feature and estimate pose by location of these features using 2D face model[1] or by template matching[3]. Jebara[7] tracked facial features in the sequence of images to generate 3D model of face and estimate pose of face.

We use 3D models of human heads in order to estimate a pose from only one 2D image. There are some difficulties with such 3D models; head shapes are different from one person to another person and, furthermore, facial expressions may vary even for one person. Nevertheless, it is unrealistic to have 3D head models for all persons and for all possible facial expressions. To deal with effectively this problem, we use a *generic* model of the human head, which is applicable to many persons and is able to consider the variety of facial expressions. Such a model is constructed from the results of intensive measurements on the heads of many people. With this 3D generic model, we suppose that an image of a head is the projection of this 3D generic model onto the image plane. Then, the problem is to estimate this transformation, which is composed of the rigid displacement of the head and a perspective projection.

We take a strategy that we define edge curves on the 3D generic model in advance. For edge curves, we use the contours of eyes, lips, eyebrows, and so on. They are caused by discontinuity of the reflectance and appear in the image independent of the head pose in 3D space. (We call these edges *stable edges*.) For each defined edge curve on the generic model, we search its corresponding curves in the image. This is done by first extracting every edge from the image and next using the relaxation method.

After we have established the correspondences between the edges curves on the model and the edges in the image, we are to estimate the head pose. For this purpose, we develop ICC (Iterative Closest Curve) method which minimizes the distance between the curves on the model and the corresponding curves in the image. This ICC method is similar to the ICP (Iterative Closest Point) method [5] [8], which minimizes the distance from points of a 3D model to the corresponding measured points of the object. Because a curve contains much richer information than a point, curve correspondences can be established more robustly and with less ambiguity, and therefore, pose estimation based on curve correspondence is thought to be more accurate than that based on point correspondence.

The ICC method is an iterative algorithm and needs a reasonable initial guess. To obtain it, prior to applying the ICC method, we roughly compute the pose of a head and the camera parameters by using the correspondence of conics fitted to the stable edges. The computation is analytically carried out. Then, a more precise pose are estimated by the ICC method. In this step, in addition to the stable edges, we use *variable edges*, which are pieces of occluding contours of a head, e.g. the contour of the face.

Our method is currently applied for extracted face area from the natural image or the face image with unicolor background. Many techniques have been reported in the literature to extract the face from clustered background.

2. Notation

The coordinates of a 3D point $\boldsymbol{X} = (X, Y, Z)^t$ in a world coordinate system and its image coordinates $\boldsymbol{x} = (u, v)^t$ are related by

$$\begin{pmatrix} \boldsymbol{x} \\ 1 \end{pmatrix} = \lambda P \begin{pmatrix} \boldsymbol{X} \\ 1 \end{pmatrix}, \quad \text{or simply} \quad \tilde{\boldsymbol{x}} = \lambda P \tilde{\boldsymbol{X}}. \tag{1}$$

where λ is an arbitrary scale factor, P is a 3×4 matrix, called the perspective projection matrix, and $\tilde{\boldsymbol{X}} = (X, Y, Z, 1)^t$ and $\tilde{\boldsymbol{x}} = (u, v, 1)^t$. The matrix P can be decomposed as

$$P = AT. \tag{2}$$

The matrix A maps the coordinates of the 3D point to the image coordinates. The general matrix A can be written as

$$A = \begin{pmatrix} \alpha_u & 0 & u_o & 0 \\ 0 & \alpha_v & v_o & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}. \tag{3}$$

α_u and α_v are the product of the focal length and the horizontal and vertical scale factors, respectively. u_o and v_o are the coordinates of the principal point of the camera, i.e., the intersection between the optical axis and the image plane. For simplicity of computation, both u_o and v_o are assumed to be 0 in our case, because the principal point is usually at the center of the image.

The matrix T denotes the positional relationship between the world coordinate system and the image coordinate system. T can be written as

$$T = \left(\begin{array}{c|c} \boldsymbol{R} & \boldsymbol{t} \\ \hline \mathbf{o} & 1 \end{array} \right). \tag{4}$$

$\boldsymbol{R}$ is a 3×3 rotation matrix and $\boldsymbol{t}$ is a translation vector.

Note that there are eight parameters to be estimated: two camera parameters α_u and α_v, three rotation parameters, and three translation parameters.

We use $\mathcal{C}_k^I (k = 1, \ldots, K)$ to denote the k-th stable curve in the image, and $\mathcal{C}_l^W(P)(l = 1, \ldots, L)$, the l-th stable curve in the model projected by P. Both $\mathcal{C}_k^I$ and $\mathcal{C}_l^W$ are 2D curves.

$\mathcal{C}_o^I$ is used to denote the contour of the face in the image. $\mathcal{C}_o^W(P)$ is the contour of the face projected by P.

$\boldsymbol{x}_i^{I_k}$ is the 2D the point belonging to the k-th curve $\mathcal{C}_k^I$ in the 2D image. $\boldsymbol{X}_j^{W_l}$ is the 3D point belonging to the l-th curve of the 3D model. $\boldsymbol{x}_j^{W_l}(P)$ is the 2D point belonging to the l-th curve $\mathcal{C}_l^W(P)$ projected by P. $\boldsymbol{X}_j^{W_l}$ and $\boldsymbol{x}_j^{W_l}(P)$ are related by

$$\boldsymbol{x}_j^{W_l}(P) = \begin{pmatrix} a_1/a_3 \\ a_2/a_3 \end{pmatrix} \quad \text{with} \quad \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} = P \boldsymbol{X}_j^{W_l}. \tag{5}$$

3. Generic Model of a Human Head

We use the generic model of the human head which is able to take account of shape differences between individuals and the changes of the facial expression. This section explains this generic model.

3.1. Construction of the Generic Model

We represent the deformation of the 3D shape of a human head (i.e., shape differences and the changes of the facial expression) by the mean $\boldsymbol{X}$ and the variance $V[\boldsymbol{X}]$ of each point on the face. These variables are calculated from the results of measuring heads of many people. To do so, we need a method for sampling points consistently for all faces. That is, we need to know which point on a face corresponds to a point on another face. Many methods have been proposed for such a purpose and we can use them; we use the resampling method [4] developed in our laboratory. This method uses several feature points (such as the corners of the eyes, the vertex of the nose, and so on) as reference points. Using these reference points, the shape of a face is segmented into several regions and further each region is resampled. We choose the sample points using this method.

3.2. Edge Extraction in the Model

As mentioned earlier, we use two types of edges: stable edges and variable edges. For stable edges, we extract them beforehand from the 2D image taken at the same time as the acquisition of the 3D data of a head. They are the contours of the eyes, lips, and eyebrows. We obtain their corresponding curves on the head by back-projecting them onto the 3D model. For variable edges, which are occluding contours and depend on the head pose and camera parameters, we extract them whenever these parameters change. Figure 1 shows an example of images of the generic model with stable and variable edges. It shows that the stable edges (i.e., the eyes and lips) do not change under the change of the pose, and the variable edge (i.e., the contour of the face) changes whenever the pose changes.

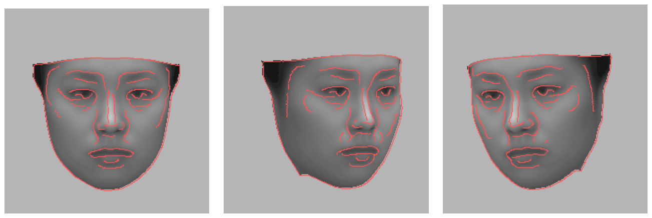

Figure 1. A generic model of a head. In all poses, the stable edges such as the eyes and lips do not change. The variable edges change because they are occluding contours.

4. Definition of the Distance Between Curves

Here we define *distance* between curves. It is a basis of the ICC method which we will present in the later section, and it is also used for finding corresponding curves.

The squared distance between a 2D curve in the image and a projection of a curve on the 3D model is defined by

$$\begin{aligned} & d(\mathcal{C}_k^I, \mathcal{C}_l^W(P)) \\ &= \frac{1}{N_k^I} \sum_{\boldsymbol{x}_i^{I_k} \in \mathcal{C}_k^I} \left(\min_{\boldsymbol{x}_j^{W_l} \in \mathcal{C}_l^W(P)} d_m(\boldsymbol{x}_i^{I_k} - \boldsymbol{x}_j^{W_l}(P)) \right), \end{aligned} \tag{6}$$

where N_k^I is the number of points in $\mathcal{C}_k^I$ and $d_m(\boldsymbol{x}_i^{I_k} - \boldsymbol{x}_j^{W_l}(P))$ is the squared Mahalanobis distance:

$$\begin{aligned} & d_m(\boldsymbol{x}_i^{I_k} - \boldsymbol{x}_j^{W_l}(P)) \\ &= (\boldsymbol{x}_i^{I_k} - \boldsymbol{x}_j^{W_l}(P))^t \boldsymbol{M}_{ij}^{kl} (\boldsymbol{x}_i^{I_k} - \boldsymbol{x}_j^{W_l}(P)), \end{aligned} \tag{7}$$

$$\boldsymbol{M}_{ij}^{kl} = \left(\left(\frac{\partial \boldsymbol{x}_j^{W_l}(P)}{\partial \boldsymbol{X}_j^{W_l}} \right) V[\boldsymbol{X}_j^{W_l}] \left(\frac{\partial \boldsymbol{x}_j^{W_l}(P)}{\partial \boldsymbol{X}_j^{W_l}} \right)^t \right)^{-1} \tag{8}$$

It is possible to give another definition of the distance between curves. Our definition is based on the following assumptions:

- When, for edges on the 3D model, the corresponding edges in the image are found, the projected model curve contains the image curve.
- The generic model is sampled at a higher resolution than the image.
- The variance $V[\boldsymbol{X}]$ of each point can be different and unisotropic.

5. Finding Corresponding Curves by Relaxation

In this section, we explain the method for finding correspondence between 3D model curves and 2D image curves. This is done by matching 2D image curves $\mathcal{C}_k^I$ and model curves $\mathcal{C}_l^W(P_o)$ projected by P_o. P_o is an arbitrary projection.

We assume that all of the eyes and lips are seen in an image. Therefore, the edges of a 2D image are expected to include stable edges. However, they also include noisy edges caused by illumination, measurement error and so on. Consequently, there are some correspondence ambiguities. We use the relaxation techniques to resolve such the correspondence ambiguities.

First, we find candidates for corresponding curves using the similarity of the curvature. Curvature of the curve is not preserved under projection. However, because we assume the pose estimate P_o is reasonable, curvature of the same curve might be similar. After finding candidates, we resolve ambiguities by relaxation method.

5.1. Finding Candidates for Corresponding Curves

Both the image edges and the projected model edges are segmented into equi-curvature curves. Candidates for corresponding pairs are found by evaluating the similarity of curvature.

The similarity of curvature $s(k,l)$ is defined as

$$s(k,l) = 1.0/(1.0 + |c(\mathcal{C}_k^I) - c(\mathcal{C}_l^W(P_o))|), \tag{9}$$

where $c(\mathcal{C})$ is the curvature of curve $\mathcal{C}$.

$s(k,l)$ has the following properties:(i) when two curves have exactly the same curvature, $s(k,l)$ equals 1, and (ii) as the difference of the curvature between two curves becomes larger, $s(k,l)$ becomes smaller.

If the value of $s(k,l)$ is higher than the threshold, the pair of curves $(\mathcal{C}_k^I, \mathcal{C}_l^W(P_o))$ is selected as the candidate pair.

5.2. Calculating the Strength of Match

If $((\mathcal{C}_k^I, \mathcal{C}_l^W(P_o)))$ is a correct pair, many of the rest of the model curves $\mathcal{C}_{k_m}^W$ have corresponding curve $\mathcal{C}_{l_n}^I$ such that the position of $\mathcal{C}_{l_n}^I$ relative to $\mathcal{C}_{k_m}^W$ is similar to that of $\mathcal{C}_l^W(P_o)$ relative to $\mathcal{C}_k^I$. We define the strength of match S_M for pair $((\mathcal{C}_k^I, \mathcal{C}_l^W(P_o)))$ in a way similar to the one for point pair used in [10].

5.3. Updating corresponding pairs of curves

The strategy we use for updating corresponding pairs is called the "some-winners-take-all" strategy[10]. Consider the corresponding pairs having the highest strength of match for both of the image and the model. These pairs are called potential matches and denoted by $\{P_i\}$. For $\{P_i\}$, two tables T_{SM} and T_{UA} are constructed.

T_{SM} saves the matching strength of each $\{P_i\}$ which is sorted in decreasing order. T_{UA} saves the value of U_A. U_A describes unambiguity and is defined as

$$U_A = 1 - S_M(2)/S_M(1) \tag{10}$$

where $S_M(1)$ is the S_M of $\{P_i\}$ and $S_M(2)$ is the S_M of the second best candidate in the pairs which include the curve forming $\{P_i\}$. T_{UA} is also sorted in decreasing order.

The pairs are selected as "correct" matches if they are among the first $q(> 50)$ percent of pairs in T_{SM} and the first q percent of pairs of T_{UA}. Using this method, the pairs which are matched well and unambiguous are selected.

6. Rough Estimation of a Head Pose

In this section, we explain the method for roughly estimating the head pose and camera parameters which are used as the initial guess in the refinement process.

To roughly estimate the head pose and the camera parameters, co-planar conics are used. Because the eyes and mouth are approximately on a single plane, the 3D stable edges of the model, such as the edges of the eyes and lips, are projected to that plane.

We use the intersections and bi-tangent lines of the co-planar conics because they are preserved under projection[2]. When using pairs of co-planar conics, at least one pair of co-planar conics is needed to determine all the parameters. But still remain two possibilities in our case: the correct one and the upside-down one. Therefore, we use three pairs of conics: left eye and right eye, left eye and lips, right eye and lips.

6.1. Projection to the Face Plane

Edge points of eyes and lips are almost on one plane, called the face plane.

Consider a coordinate system in which the face plane coincides with $z = 0$. We call such a coordinate system the plane coordinate system.

The 3D coordinates of a point projected to the face plane $\boldsymbol{X}$ in the world coordinate system and the coordinates of the point $(x_p, y_p, 0)^t$ in the plane coordinate system are related by

$$\tilde{\boldsymbol{X}} = \left(\begin{array}{c|c} \boldsymbol{R}_p & \boldsymbol{t}_p \\ \hline \mathbf{o} & 1 \end{array} \right) \begin{pmatrix} x_p \\ y_p \\ 0 \\ 1 \end{pmatrix} = T_p \begin{pmatrix} x_p \\ y_p \\ 0 \\ 1 \end{pmatrix}, \tag{11}$$

where T_p means the positional relationship between the world coordinate system and the plane coordinate system.

From equations (1) and (11), we have

$$\tilde{\boldsymbol{x}} \;=\; \lambda H \begin{pmatrix} x_p \\ y_p \\ 1 \end{pmatrix}, \qquad (12)$$

where H is a 3×3 matrix, given by

$$H = \begin{pmatrix} \alpha_u r_{11} & \alpha_u r_{12} & \alpha_u t_1 \\ \alpha_v r_{21} & \alpha_v r_{22} & \alpha_v t_2 \\ r_{31} & r_{32} & t_3 \end{pmatrix}, \qquad (13)$$

where r_{ij} is the (i, j)-th component of $\boldsymbol{R}' = \boldsymbol{R}\boldsymbol{R}_p$, and t_i is the i-th component of $\boldsymbol{t}' = \boldsymbol{R}\boldsymbol{t}_p + \boldsymbol{t}$.

6.2. Intersection and bi-tangent of co-planar conics

A conic in a 2D space is a set of points $\boldsymbol{x}$ that satisfy

$$\tilde{\boldsymbol{x}}^t \boldsymbol{Q} \tilde{\boldsymbol{x}} = 0, \qquad (14)$$

where $\boldsymbol{Q}$ is a 3×3 symmetric matrix. We fit a conic to the edge points of right eye, left eye, and lips by gradient weighted least square fitting described in [9].

The intersection $\tilde{\boldsymbol{m}}$ for two conics $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$ satisfies the following simultaneous equations:

$$\tilde{\boldsymbol{m}}^t \boldsymbol{Q}_1 \tilde{\boldsymbol{m}} = 0, \text{ and } \tilde{\boldsymbol{m}}^t \boldsymbol{Q}_2 \tilde{\boldsymbol{m}} = 0. \qquad (15)$$

Denoting bi-tangent line for two conics $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$ as $\tilde{\boldsymbol{l}}^t \tilde{\boldsymbol{x}} = 0, \tilde{\boldsymbol{l}}$ satisfies the following simultaneous equations[2]:

$$\tilde{\boldsymbol{l}}^t \boldsymbol{Q}_1^{-1} \tilde{\boldsymbol{l}} = 0, \text{ and } \tilde{\boldsymbol{l}}^t \boldsymbol{Q}_2^{-1} \tilde{\boldsymbol{l}} = 0. \qquad (16)$$

$\boldsymbol{m}$ and $\boldsymbol{l}$ are obtained by solving quartic equations analytically.

6.3. Combinations of the Correspondence

There are no real intersections for these pairs of conics. Therefore, the solutions of the quartic equation are two complex conjugate pairs. In complex cases, there are eight possibilities to correspond four points of the image to four points of the model because conjugate pairs project to conjugate pairs under real projection[2].

On the other hand, because all of the bi-tangent lines are real in this case, there are only four possibilities of correspondence.

Therefore, there are 32 possible combinations for each pair of conics. When we use three pairs of conics, the number of the all possible pairs are $32^3 (= 32768)$.

We reduce the number of combinations. Because there are two possibilities are remaining in our case for only one pair of conics (the true one and the up-side-down one), we select two combinations for each pair of conics. Then using these combinations for three pairs of conics, all possible values of H is calculated by the linear least squares described in appendix A. The number of possible values of H is much reduced to $32 + 32 + 32 + 2^3 (= 104)$.

We select the best one among all possible values of H by evaluating H. The method for evaluation is descried in appendix B.

From equation (13), unknown parameters are obtained using every components of H (see appendix C). This is the initial guess for refinement process.

7. Refinement of the Head Pose by ICC (Iterative Closest Curve) Method

In this section, we explain the method for refinement of the head pose and camera parameters using the initial guess obtained by the method described in the previous section. We employ the ICC method which minimizes the distance between corresponding curves.

We use the correspondence of two types of edges in this process: stable ones and variable ones.

The correspondence of stable edge curves have been established by the method described in section 5.

Variable edges, e.g. the contour of the face, of the generic model should be extracted whenever the parameters are updated because this curve varies whenever the parameters change. However, the correspondence of the contour of the face is known.

Once the correspondence of the curves are established, the squared Mahalanobis distance of corresponding curves is minimized.

We minimize the value of the function J:

$$\begin{aligned} J &= \sum_l d(\mathcal{C}_k^I, \mathcal{C}_l^W(k)(P)) + d(\mathcal{C}_o^I, \mathcal{C}_o^W(P)) \qquad (17) \\ &= \sum_l \frac{1}{N_k^I} \sum_{\boldsymbol{x}_i^{I_k} \in \mathcal{C}_k^I} \left(\min_{\boldsymbol{x}_j^{W_l} \in \mathcal{C}_l^W(P)} d_m(\boldsymbol{x}_i^{I_k} - \boldsymbol{x}_j^{W_l}(P)) \right) \\ &\quad + \frac{1}{N_o^I} \sum_{\boldsymbol{x}_i^{I_o k} \in \mathcal{C}_o^I} \left(\min_{\boldsymbol{x}_j^{W_o} \in \mathcal{C}_o^W(P)} d_m(\boldsymbol{x}_i^{I_o} - \boldsymbol{x}_j^{W_o}(P)) \right). \end{aligned} \qquad (18)$$

We minimize the value of J to find P by iterating these two steps:

- For each image point $\boldsymbol{x}_i^{I_k}$ of each corresponding curve pairs $(\mathcal{C}_k^I, \mathcal{C}_l^W(P))$, the point $\boldsymbol{x}_j^{W_l}$ which minimize $d_m(\boldsymbol{x}_i^{I_k} - \boldsymbol{x}_j^{W_l})$ are found.

- P is updated to minimize J by Levenverg-Marquart algorithm.

P include the head pose and camera parameters in equation 2. We directly estimate eight parameters, i.e., three rotation parameters, three translation parameters, and two camera parameters, instead of each component of P.

7.1. Non-linear Minimization of the Distance between Curves

From equation (2), P is decomposed into a perspective projection and the rigid displacement.

Non-linear minimization with constraints of the rotation matrix is complicated. Therefore, we rewrite the rigid displacement part by using a 3D vector $\boldsymbol{q}$ as

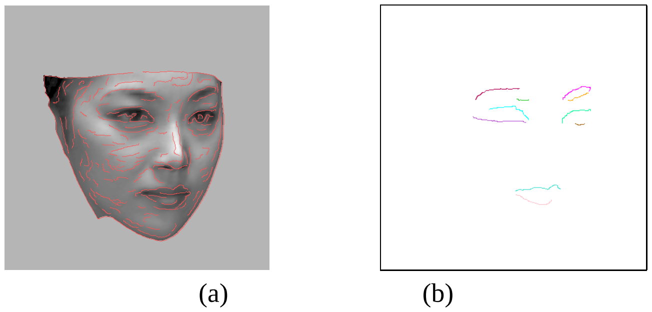

Figure 2. (a) Extracted edges in images of one woman's face and (b) edge curves of the eyes, lips, and eyebrows extracted by the correspondence between the model and the image.

$$
\begin{aligned}
T\tilde{\boldsymbol{X}} &= \boldsymbol{R}\boldsymbol{X} + \boldsymbol{t} && (19)\\
&= \boldsymbol{X} + \frac{2}{1+\boldsymbol{q}^t\boldsymbol{q}}(\boldsymbol{q}\times\boldsymbol{X} - (\boldsymbol{q}\times\boldsymbol{X})\times\boldsymbol{q}) + \boldsymbol{t}. && (20)
\end{aligned}
$$

The direction of $\boldsymbol{q}$ is equal to the rotation axis and the norm of $\boldsymbol{q}$ is equal to tan $\frac{\theta}{2}$ where θ is the rotation angle. Using this equation, because the three component of $\boldsymbol{q}$ are independent, the minimization becomes much simpler.

8. Experimental Result

We show in this section some preliminary result with the proposed technique.

Figure 1 shows the model edges constructed from 36 women's head measurements. All of these are with no facial expressions. Figure 2(a) shows the edges of an image of one woman. These edges are extracted by the method described in [6].

Figure 2(b) shows the extracted stable edge curves, i.e., the contour of the eyes, lips, and eyebrows. These edges are extracted by establishing the correspondence between model edges and image edges described in section 5.

Figure 3(a) shows the co-planar conics fitted to contours of the eyes and lips in the image showed in figure 2(a). Figure 3(a) shows the result of rough estimation. The conics of the model are plotted in red and the conics of the image are plotted in black.

The head pose and camera parameters of the image shown in Fig. 2(a) was estimated. Figure 4 shows the projection of the generic model by the estimated parameters. The pose of the head shown in Fig. 2(a) and that of Fig. 4 are almost the same.

9. Conclusion

Head pose determination is very important for many applications such as human-computer interface and video conferencing. In this paper, we have proposed a new method for estimating accurately a head pose from only one image. To deal with shape variation of heads among individuals and different facial expressions, we use a generic 3D model of the human head, which was built through statistical analysis of range data of many heads. In particular, we use a set of 3D curves to model the contours of eyes, lips, and eyebrows.

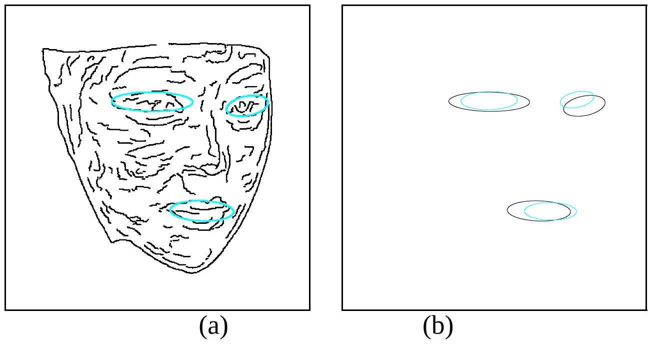

Figure 3. Edges and conic of the eyes and lips and the result of rough estimation using conics. (a) Edges of a woman's face and co-planar conics. (b) The results of rough estimation using conics of (a). The conics of the image are plotted in black and the projection of model conics are plotted in red.

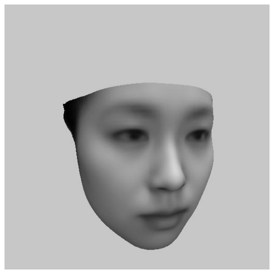

Figure 4. The result of the head pose estimation using ICC.

We have proposed the iterative closest curve matching (ICC) method which estimates directly the pose by iteratively minimizing the squared Mahalanobis distance between the projected model curves and the corresponding curves in the image. The curve correspondence is established by the relaxation technique. Because a curve contains much richer information than a point, curve correspondences can be established more robustly and with less ambiguity, and therefore, pose estimation based the ICC is believed to be more accurate than that based on the well-known ICP.

Furthermore, our technique does not assume that the internal parameters of a camera is known. This provides more flexibility in practice because an uncalibrated camera can be used. The unknown parameters are recovered by our technique thanks to the generic 3D model.

Preliminary experimental results show that (i) accurate head pose can be estimated by our method using the generic model and (ii) this generic model can deal with the shape difference between individuals. The accuracy of the pose estimation depends tightly on whether image curves can be successfully extracted. More experiments need to be carried out for different facial expressions and for cluttered background.

We believe that the ICC method is useful not only for 3D-2D pose estimation but also for 2D-2D or 3D-3D pose estimation.

Acknowledgment:

We thank K.Isono for his help in the presentation of experimental data.

References

[1] A.Lanitis, C.J.Taylor and T.F.Cootes. Automatic Interpretation and Coding of Face Images Using Flexible Models. *IEEE Trans. PAMI*, 19(7):743–756, 1997.

[2] C.A.Rothwell, A.Zisserman, C.I.Marinos, D.A.Forsyth and J.L.Mundy. Relative Motion and Pose from Arbitrary Plane Curves. *IVC*, 10(4):250–262, 1992.

[3] D.J.Beymer. Face Recognition Under Varying Pose. In *CVPR94*, pages 756–761, 1994.

[4] K.Isono and S.Akamatsu. A Representation for 3D Faces with Better Feature Correspondence for Image Generation using PCA. Technical Report HIP96-17, IEICE, 1996.

[5] P.J.Besl and N.D.McKay. A Method for Registration 3-D Shapes. *IEEE Trans. PAMI*, 14(2):239–256, 1992.

[6] R.Deriche. Using Canny's Criteria to Derive a Recursively Implemented Optimal Edge Detector. *IJCV*, 1(2):167–187, 1987.

[7] T.S.Jebara and A.Pentland. Parametrized Structure from Motion for 3D Adaptive Feedback Tracking of Faces. In *CVPR97*, pages 144–150, 1997.

[8] Z.Zhang. Iterative Point Matching for Registration of Free-Form Curves and Surfaces. *IJCV*, 13(2):119–152, 1994.

[9] Z.Zhang. Parameter Estimation Techniques: A Tutorial with Application to Conic Fitting. *IVC*, 15:59–76, 1997.

[10] Z.Zhang, R.Deriche, O.Faugeras and Q.T.Luong. A Robust Technique for Matching Two Uncaribrated Images Through the Recovery of the Unknown Epipolar Geometry. *AI Journal*, 78:87–119, 1995.

A. Linear Estimation of H

Assume the image point (x, y) and the object point (x_p, y_p) are the corresponding pair. We rewrite the components of H as

$$H = \begin{pmatrix} a & b & c \\ d & e & f \\ g & h & 1 \end{pmatrix}. \tag{21}$$

By eliminating λ in equation (12), we get

$$ax_p + by_p + c - gx_px - hy_px \quad = \quad x, \tag{22}$$

and

$$dx_p + ey_p + f - gx_py - hy_py \quad = \quad y. \tag{23}$$

From equation (22) and equation (23), the components of H are calculated by the linear least square algorithm.

B. Eliminating Ambiguous Solutions for H

We select the best correspondence combination which minimizes the criterion function.

If the H is correct, conic on the face plane $\boldsymbol{Q}^P$ and the image conic $\boldsymbol{Q}^I$ satisfy the following equation:

$$\boldsymbol{Q}^P = \lambda^2 H^t \boldsymbol{Q}^I H. \tag{24}$$

Using this relation, the criterion function $e(H)$ is defined as[2]:

$$\begin{aligned} e(H) = {} & (I_{ab3}(\boldsymbol{Q}^{P\prime}_{m1}, \boldsymbol{Q}^P_1) - 3)^2 + (I_{ab4}(\boldsymbol{Q}^{P\prime}_{m1}, \boldsymbol{Q}^P_1) - 3)^2 \\ & + (I_{ab3}(\boldsymbol{Q}^{P\prime}_{m2}, \boldsymbol{Q}^P_2) - 3)^2 + (I_{ab4}(\boldsymbol{Q}^{P\prime}_{m2}, \boldsymbol{Q}^P_2) - 3)^2 \end{aligned} \tag{25}$$

where

$$\boldsymbol{Q}^{P\prime}_{mi} = H^t_m \boldsymbol{Q}^I_i H_m, \tag{26}$$

and

$$I_{ab3}(A, B) \quad = \quad \text{trace}\left[\left((1/\det A)A\right)^{-1}(1/\det B)B\right], \tag{27}$$

$$I_{ab4}(A, B) \quad = \quad \text{trace}\left[\left((1/\det B)B\right)^{-1}(1/\det A)A\right] \tag{28}$$

C. Decomposition of H

From equation (13), the head pose and camera parameters are determined using every components of H.

Because $\boldsymbol{R}$ is a rotation matrix, we have

$$r_{11}^2 + r_{21}^2 + r_{31}^2 \quad = \quad 1, \tag{29}$$

$$r_{12}^2 + r_{22}^2 + r_{32}^2 \quad = \quad 1, \tag{30}$$

$$r_{11}r_{12} + r_{21}r_{22} + r_{31}r_{32} \quad = \quad 0. \tag{31}$$

We use h_{ij} to denotes the (i, j)-th component of H. From equations 13 and 31, we have

$$h_{11}h_{12}/\alpha_u^2 + h_{21}h_{22}/\alpha_v^2 + h_{31}h_{32} = 0. \tag{32}$$

From equations 29 and 30, we also have

$$\lambda^2\left(h_{11}^2/\alpha_u^2 + h_{21}^2/\alpha_v^2 + h_{31}^2\right) \quad = \quad 1, \tag{33}$$

$$\lambda^2\left(h_{12}^2/\alpha_u^2 + h_{22}^2/\alpha_v^2 + h_{32}^2\right) \quad = \quad 1. \tag{34}$$

Then, by eliminating λ^2, we have

$$(h_{11}^2 - h_{12}^2)/\alpha_u^2 + (h_{21}^2 - h_{22}^2)/\alpha_v^2 + h_{31}^2 - h_{32}^2 = 0. \tag{35}$$

Let $\beta_u = \frac{1}{\alpha_u}$ and $\beta_v = \frac{1}{\alpha_v}$. From equation (32) and (35), we have

$$\beta_u \quad = \quad \frac{-h_{31}h_{32}(h_{21}^2 - h_{22}^2) + h_{21}h_{22}(h_{31}^2 - h_{32}^2)}{d} \tag{36}$$

$$\beta_v \quad = \quad \frac{-h_{31}h_{32}(h_{11}^2 - h_{12}^2) + h_{11}h_{12}(h_{31}^2 - h_{32}^2)}{d} \tag{37}$$

where

$$d = h_{11}h_{12}(h_{21}^2 - h_{12}^2) - h_{21}h_{22}(h_{11}^2 - h_{12}^2). \tag{38}$$

Once α_u and α_v are estimated, we can compute λ using equation (33) or (34). All of the pose parameters are given by

$$r_{11} = \lambda h_{11}/\alpha_u, r_{21} = \lambda h_{21}/\alpha_v, r_{31} = \lambda h_{31}, \tag{39}$$

$$r_{12} = \lambda h_{12}/\alpha_u, r_{22} = \lambda h_{22}/\alpha_v, r_{32} = \lambda h_{32}, \tag{40}$$

$$t_1 = \lambda h_{13}/\alpha_u, t_2 = \lambda h_{23}/\alpha_v, t_3 = \lambda h_{33}. \tag{41}$$

$r_{i3}(i = 1, \ldots, 3)$ can be easily computed using the orthogonality of the rotation matrix.