IMAVIS1869

ELSEVIER

Image and Vision Computing 00 (2002) 000–000

image AND vision COMPUTING

www.elsevier.com/locate/imavis

3D Face pose estimation and tracking from a monocular camera

Qiang Ji*, Rong Hu

Department of Electrical, Computer and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY 121801, USA

Received 23 May 2000; received in revised form 4 November 2001; accepted 21 January 2002

Abstract

In this paper, we describe a new approach for estimating and tracking three-dimensional (3D) pose of a human face from the face images obtained from a single monocular view with full perspective projection. We assume that the shape of a 3D face can be approximated by an ellipse and that the aspect ratio of 3D face ellipse is given. Given a monocular image of a face, we first perform an ellipse detection to locate the face in the image and the 3D position and orientation of the face are then estimated from the detected image face. The face detection is greatly facilitated by exploring the physiological properties of eyes under a special IR illumination and some geometric constraints. The detected initial face ellipse is then tracked in subsequent frames, allowing to track 3D face pose from frame to frame. Compared with the existing feature-based approaches for face pose estimation, our approach is more robust, since ellipse can be more reliably and robustly detected and tracked. Experimental study using a large number of synthetic and real images demonstrates the accuracy and robustness of the proposed approach. © 2002 Elsevier Science B.V. All rights reserved.

Keywords: Pose estimation; Face tracking; Face detection; Gaze estimation

1. Introduction

Face pose determination represents an important area of research in human computer interaction (HCI). An important problem in HCI is to determine one's focus of attention (inattention or lack of attention) and needs. This can be inferred from the person's head orientation and gaze direction. Both head and gaze directions can be estimated from one's face orientation. Moreover, a person's state of mind and/or level of vigilance can also be deduced from his/her face orientation. For example, tracking one's face orientation through multiple image frames allows us to detect the nodding behavior, which can be used to infer one's fatigue level. In summary, face pose estimation and tracking play an important role in HCI.

Methods for face pose estimation can be classified into two main categories: *model-based* and *face property-based* (*or appearance based*). Model-based approaches assume a three-dimensional (3D) model of the face and typically recover the face pose by first establishing 2–3D features correspondences and then solving for the face pose using the conventional pose estimation techniques. Property-based approaches, on the other hand, assume there exists a unique causal–effect relationship between 3D face pose and certain properties of the facial image. Their goal is to determine the relationship from a large number of training images with known 3D face poses. Image properties may include image intensity (appearance), color, image gradient, or transformation of image intensity, such as image projection onto eigenspace. Image property may also include geometric properties of the images or region of the images.

Several appearance-based methods employ artificial neural networks to estimate face pose. They use neural networks to construct a mapping between 2D face images and 3D face poses. These approaches are based on learning pose model from appearance examples. In Ref. [21], a head orientation recognition system is reported. Although this system is able to identify the face direction in front of laboratory background, it cannot work well for previously unseen persons and backgrounds. In Ref. [12], neural networks are used to estimate the pose of a rigid 3D object. Its estimation is based on a number of 2D snapshots of the object with known poses. A 2D/3D mapping is constructed using training data. The constructed mapping is then used to estimate 3D face pose of a given 2D face image.

Darrell et al. [5] proposed to use eigenspace for face detection and face pose estimation. A separate set of eigenspaces is computed for each face and for each possible pose. A face and its pose are determined by computing the

* Corresponding author. Address: Department of Computer Science, 171 College of Engineering, University of Nevada at Reno, Reno, NV 89557-0148, USA. Tel.: +1-775-784-4613; fax: +1-775-784-1877.
E-mail address: qiangji@cs.unr.edu (Q. Ji).

0262-8856/02/$ - see front matter © 2002 Elsevier Science B.V. All rights reserved.
PII: S0262-8856(02)00024-0

eigenspace projection of the input image onto each eigenspace and selecting the one with the lowest residual error. The pose determination is formulated as MAP estimation problem. Murase and Nayar [17] presented a similar eigenspace method. Given N individuals and M different poses, their method projects each image onto a common eigenspace, forming $N \times M$ eigenspace vector classes, each of which encodes both identity and pose information. The identity and pose of an input face are determined by selecting the eigenvector class, that is, closest to the eigenspace vector representing the input image. The underlying assumption of eigenspace approach is that there is a unique correlation between the 3D pose of a face and its eigenspace projection. The approach proposed by Chen et al. [4] models the head as a combination of skin and hair regions. Geometric properties, such as area, center, and geometric moments of each region are computed. A correlation is then experimentally established between the computed geometric properties and the head pose, based on which the 3D head pose can be estimated for an input head image. Shioyama et al. [20] also utilize color information to estimate pose. Firstly, the skin region and the hair region in an image are extracted by estimation the skin color likeness and the hair color likeness for each pixels with the models named as skin color distribution model and hair color distribution model.

In summary, the property-based methods are simpler. They are, however, less accurate, since many of them require interpolation. Moreover, they usually require a large number of training face images of different people and under different orientations, illuminations, and scales, the reason being the facial pattern changes in various ways due to individuality, scale, and illumination. There is also question of the validity of the underlying assumption that a unique causal-effect relationship exists between certain properties of face image and the 3D face pose. This assumption has not been validated and the relationship may not be unique. Even assume there exists such an unique relationship, the relationship may be complex and a large number of training images are needed to determine it.

Many model-based approaches have been reported in the literature. Most of them model a face with certain facial features. These methods usually start with feature detection, followed by matching 2D/3D corresponding features and determining face pose using the matched features. Among all facial features, the most commonly used ones are eyes and the mouth. Nikolaidis and Pitas [18] determine the pose orientation using the isosceles triangle constructed by eyes and the mouth. In Ref. [22], six facial feature points including pupils, nostrils, and lip corners are used to model a face. In Ref. [13], five feature points containing four eye corners and the tip of the nose are used to model the head. To recover the 3D face pose using five points, they employ projective invariance of cross ratios and the statistical modeling for face structure from anthropometry. Gee and Cipolla [7] proposed a facial model based on the ratios of lengths between some facial features. Assuming weak perspective projection, their approach recovers face pose using the 2D/3D length ratios and symmetry. Ballard and Stockman [1] described a method for recovering 3D face pose using three facial points (one for each eye and the third point is either from mouth or nose). The two eye points are detected first and they are then used to guide the detection of the third point. Given the detected three feature points and assuming the 3D distances between them are known, they then apply an iterative procedure to solve for the 3D face pose. Without taking the assumption of orthogonal projection as in the previous methods, Ho and Huang [11] proposed an analytic solution applied to perspective projection by utilizing four feature points in the image: far corners of the eye and that of the mouth. But, this solution requires priory knowledge of 3D distance between the two far eye corners. A RAndom SAmple Consensus (RANSAC)/alignment method is proposed by Gee and Cipolla [8] for pose calculation. The idea behind is to find a significant group of points, which are all consistent with a particular pose, and reject the remaining points as outliers. Smith and co-workers [19] estimate face pose by constructing a bounding box of head first, then determining rotation using distance between eye and lip feature points and sides of face.

Gee and Cipolla [6] presented another method for face pose estimation and tracking using six facial features. Their approach assumes weak perspective projection and also requires prior knowledge of the distances between eyes and mouth corners. The distances, however, may vary under different facial expressions. Lee and Tsukamato [15] proposed a model-based real-time system for face tracking and face pose determination. Their system uses a generic 3D graphic model of human face. To create more reliable model image, the 3D model is reshaped using a face image with known pose. Synthesized model images at eight different poses are then generated from the reshaped 3D face model. A disparity measure based on intensity correlation between the model images of known 3D poses and an input image is used to determine the 3D pose of the input image. For face detection, the authors proposed a qualitative model of a full face consisting of a combination of image blocks with different lightness and edgeness. Hattori et al. [10] introduced an approach for estimating 3D pose of a human face using both range data and the symmetric property of a face. The approach first used both range data and intensity image to detect 3D positions of eyes and eyebrows, based on which an initial estimation of the parameters of the symmetry plane is obtained. The plane parameters are subsequently improved/refined via a non-linear minimization procedure using range data. The normal of the symmetry plane can then be used to estimate face pose. The limitations of this approach lie at its need of range data and its need to detect and track facial features in 3D. Gordon [9] also advocated 3D face pose estimation from 3D facial features data obtained using structure from motion for video

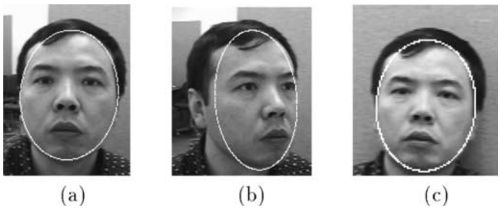


Fig. 1. Face shape distortions due to face orientation (foreshortening) and perspective projection: (a) look front without distortion; (b) look sideway with foreshortening distortion; and (c) look front but far from camera with perspective distortion.

imagery. His approach requires 3D detection and tracking certain facial features.

The strengths of model-based methods include simple implementation, the availability of large amount of pose determination theories, and high accuracy and efficiency if features can be detected accurately. Unlike other stationary rigid bodies, a face is in constant motion in both orientation and position. The major challenge, therefore, facing feature-based approach is to robustly detect and track the required facial features from frame to frame under varying illuminations and different head orientations. Feature-based methods also suffer the same weakness as point-based pose estimation methods, i.e. it is difficult to establish 2D/3D correspondence of these facial points. Moreover, most of feature-based methods assume weak perspective projection. This assumption makes their methods unsuitable for applications, where human faces are close to the camera, such as cases, where a user is before a computer or a machine. Finally, since face is not a complete rigid body, its shape and certain feature measurements may change under different facial expressions. The required facial features may be occluded by beard or glasses. Since the shape of individual faces varies quite a bit, using a general face shape model (either in terms of some ratios or distances) introduces an inherent error into the pose estimation.

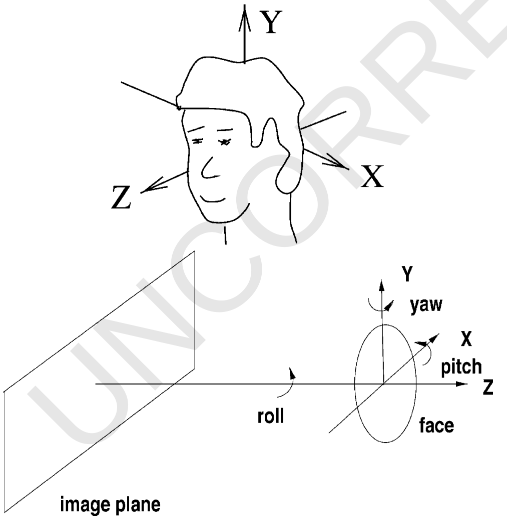


Fig. 2. The definition of three face rotation angles: roll, yaw, and pitch.

In view of these limitations, we propose a novel model-based approach for estimating the 3D orientation and position of a human face with respect to the camera frame from a monocular image of the face. Our approach models the shape of a face with an ellipse. The use of ellipse to model a face has the following advantages: first, human face can be rather accurately modeled with an ellipse. A person's face silhouette is less sensitive to facial expression change and less likely completely occluded by glasses, beard, or other objects. Second, like facial feature points, ellipses are preserved under perspective or projective transformations. Third, unlike feature points, ellipses contain compact global information of the face; they are expected to be more robust to noise than facial feature points. It tolerates partial face occlusion. Fourth, the 2D/3D correspondences can be established much more easily.

According to this approach, the observed elliptical shapes (e.g. faces) appear distorted under different 3D face orientations and positions as shown in Fig. 1. There are two sources contributing to face shape distortion: foreshortening and perspective projection. Face orientation changes introduce foreshortening distortion as shown in Fig. 1(b). Perspective distortion renders a smaller face as the face is far from camera as shown in Fig. 1(c). In practice, the face shape distortion can be a combination of both distortions.

Hence, shape-based face pose estimation amounts to inferring face orientation and position from face shape distortions. This process in general, however, requires two images, since inferring 3D face pose from a single image is ill-posed unless assumptions can be made or multiple images from different view points are utilized. In Ref. [16], a reconstruction technique is introduced to reconstruct a 3D conic and its pose from its two images obtained from two different viewpoints. Since their approach requires at least two images obtained from different orientations, it is not practical to estimate face pose considering the complexity in system setup and the associated computational time.

To overcome the problem that a single ellipse image is not sufficient to recover 3D face pose, we assume that the ratio of the major to minor axes of the 3D face ellipse (hereafter referred to as face aspect ratio) is known or can be obtained on-the-fly. This assumption is not unreasonable. Since the ratio is preserved under full perspective projection in the absence of foreshortening distortion, the face aspect ratio can be obtained from the image of the face, when the face is in front of the camera without any pitch (around horizontal axis) and yaw (around vertical axis) rotations. Given this assumption, we will be able to recover the 3D face pose from a single image. Fig. 2 defines the three possible face rotation angles: roll, yaw, and pitch.

An overview of our algorithm is given in Fig. 3. This paper is organized as follows. In Section 2, we will briefly discuss the special illuminator we built for efficient pupils

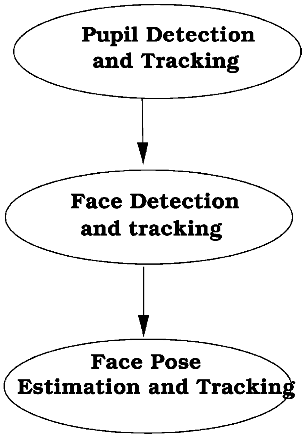

Pupil Detection and Tracking
Face Detection and tracking
Face Pose Estimation and Tracking

Fig. 3. Major components of the proposed system.

detection and tracking. We discuss in Section 3 on how to recover the roll angles from the detected pupils. Our algorithm for face ellipse detection and tracking is briefly covered in Section 4. In Section 5, we describe the mathematical model for estimating face pose from a single monocular view. Section 6 discusses the experimental results. Both simulation and real data are used to validate the proposed technique. The paper ends in Section 7 with a summary and a discussion of future work.

2. Image acquisition and pupil detection

The quality of an image is important for efficient and robust face detection. Specifically, the acquired images should have relatively consistent illumination under different climatic conditions and should, if possible, produce distinguishable features that can facilitate the subsequent face detection and tracking. To this end, the person's face is illuminated using a near-infrared illuminator. The use of infrared illuminator serves three purposes: first it minimizes the impact of different ambient light conditions, therefore ensuring image quality under varying real-world conditions including poor illumination, day, and night. Second, it allows to produce the bright-pupil effect, which will be used to guide face detection and tracking. Third, since the near infrared is barely visible to the user, this will minimize any interference with the user work. To further minimize interference from light sources beyond infrared light and to maintain uniform illumination under different climatic conditions, a narrow bandpass near infrared filter is installed before the camera lens. The IR illuminator produces an image of the subject with pupils much brighter than other part of the face as shown in Fig. 4. The high intensity contrast between the pupil and the rest of images allows easy pupil detection via a simple global thresholding. This is a computationally inexpensive and robust operation and can be implemented in real time. Our study shows that the technique works even for people with glasses and to certain degree even with sun glasses.

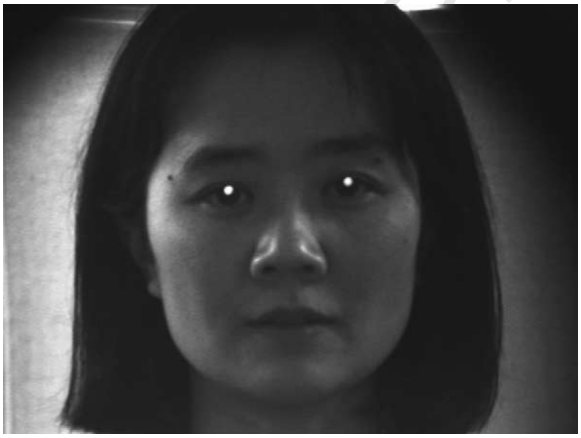


Fig. 4. The bright-pupil effect. The pupils are brighter than the rest part of the face.

To continuously monitor the user, it is important to track the pupils from frame to frame in real time. This can be done by performing a pupil detection in each frame. This brute force method, however, will significantly slow down the speed of pupil detection, making real-time pupil tracking impossible, since it needs to search the entire image for each frame. This can be done more efficiently by using the location of the pupil in previous frames to predict the location of the face in future frames based on Kalman filtering [3], assuming that the person's pupil will not undergo significant locational change in two consecutive frames. This assumption is realistic since sudden face position change is not expected under normal condition. The prediction scheme can significantly reduce search area, making pupil detection in real time possible. Fig. 5 shows a sequence of images with detected pupils on faces of different orientations. More detailed discussion on pupils detection and tracking may be found in Ref. [14].

3. Roll determination

Roll angle is defined as the angle resulted from face rotation about the optical axis (z axis) as defined in Fig. 2. Given the locations of two pupils, roll recovery is straightforward. From Fig. 6, we see immediately that face roll is

$$\gamma = \arctan\frac{p_{1_y} - p_{2_y}}{p_{1_x} - p_{2_x}}$$

where p_1 and p_2 are the image coordinates of the two detected pupils. The corresponding 3D pupils are P_1 and P_2.

4. Face detection and tracking

Face detection and tracking are concerned with determining the location of the face at each frame. Accurate image

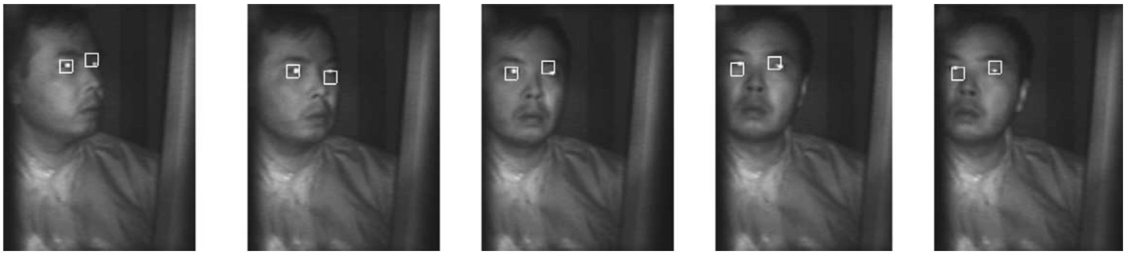


Fig. 5. The pupil tracking algorithm tracks the pupils as the head rotates. The white squares indicate the positions of the detected pupils at different face orientations.

face detection at different orientations is critical to the success of the proposed face pose estimation algorithm. Unlike the conventional face detection methods, which are either essentially head detection or detection of face, when it faces to the camera, we are only interested in the detection of frontal part of the face. Detection of frontal face is difficult, since the image face may have significantly different shapes under different face orientations. The existing approaches for automatic face detection based on color, texture, facial features, or motion cues may not work well, since they cannot distinguish frontal part of the face from the sides of the face. The conventional face detection based on ellipse fitting assumes that the occluding boundaries of the face are approximated well by an ellipse and further that the face boundaries fall along intensity gradients. The basic approach is error prone, because the occluding boundaries of the face do not correspond to the same physical location on the face surface in different views of the face, particularly along the left and right sides of the face. The reliance on intensity gradients is also problematic, because it depends heavily on lighting conditions and on the existence of contrast between hair and skin and between face and background. As a result, an ellipse-based face pose tracker will need additional strong constraints to be useful. In our application, there exist a number of constraints, which can significantly constrain the ellipse-based face detection and lead to convergence to correct face ellipse under different face orientations. Specifically, the environment under which our system will work may be summarized as

- relatively simple background
- limited area of face movement
- single individual face
- infrared illumination.

In particular, the use of infrared LEDs as the illumination source allows us to employ a different strategy for face detection. Contrary to conventional approaches, where face detection precedes facial features detection, the proposed approach first searches for certain easily identifiable facial features (e.g. eyes and nostril) and then uses the locations of these features to locate face. Specifically, the IR illumination allows us to detect pupils first. The detected pupils are then efficiently tracked in real time from frame to frame [14]. The locations of the tracked pupils can, in turn, be utilized to guide face detection and tracking. The face detection and tracking algorithm consists of three steps:

- detect an approximate initial location of the face based on the locations of the detected pupils
- perform a constrained sum of gradients maximization procedure to search for the exact face location
- the detected face is subsequently tracked using Kalman filtering.

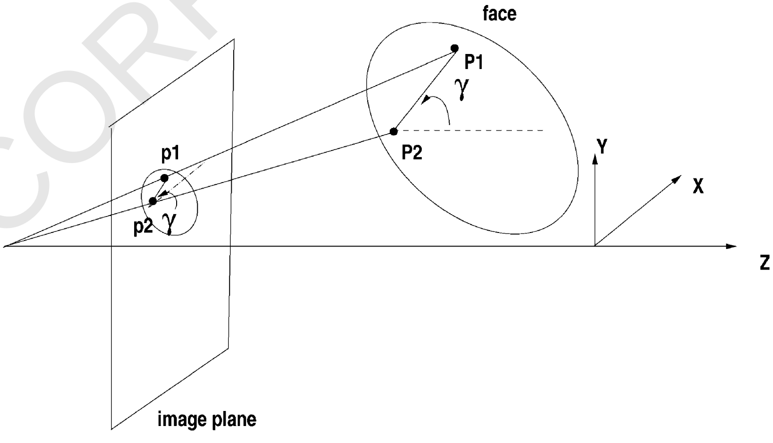


Fig. 6. The determination of roll angle from the detected pupils.

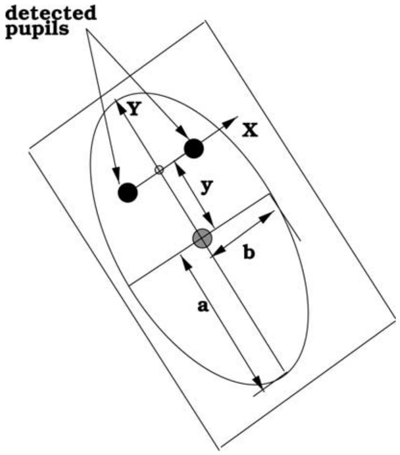


Fig. 7. Face ellipse parameterization with the pupils locations.

Of the three steps, step 2 is critical. In Section 4.1, we will discuss step 2 in more detail.

4.1. Face detection and tracking

The first step gives an approximate location of the face based on the positions of the detected pupils. In the second step, we apply a variant of the technique proposed by Birchfield [2] to determine the exact position and size of the best ellipse by maximizing the normalized sum of image gradients around the perimeter of the face subject to certain constraints.

Birchfield's technique is aimed at detecting head instead of face. To detect face, we put constraints on the gradient maximization procedure to force the detected ellipse corresponding to the frontal part of the face. The constraints we exploit include size, location, symmetry, and shape. Specifically, we assume the image of the frontal face can be approximated by an ellipse. Given the detected pupils, we draw a line segment between the detected pupils as shown in Fig. 7. The distance between the detected pupils and their locations are used to constrain the size and location of the image face ellipse. The image ellipse should minimally and symmetrically include the two detected pupils. Specifically, since the subject is about 3 ft away from the camera and the frontal face can be approximated as a planar, we can presume weak perspective projection. Therefore, the 3D face symmetry that the two pupils should symmetrically locate within the 3D face ellipse continues to hold in 2D image, i.e. the major axis of the face ellipse should be orthogonal to and pass through the center of line segment connecting the two pupils as shown in Fig. 7.

Given the roll angle determined, the image face ellipse orientation is known. We can therefore select the major axis as the y axis of the ellipse coordinate system as shown in Fig. 7. Hence, the image face ellipse can be characterized by three parameters (y, a, b), where y is the y coordinate of the ellipse center (center x coordinate is 0) and a and b are the lengths of the major and minor semi-axes of the ellipse, respectively.

Given the three parameters to estimate, the image ellipse can be detected as the one that minimizes the sum of gradient magnitudes projected along the directions orthogonal to the ellipse around the perimeter of the ellipse. Mathematically, this can be formulated as follows:

$$\epsilon^2 = \frac{1}{N} \sum_{i=1}^{N} |n(i)\cdot g(i)|^2 \qquad (1)$$

where $n(i)$ is the unit vector normal to the ellipse at pixel i, $g(i)$ is the intensity gradient at pixel i, and $(\cdot)$ denotes dot product. Therefore, the best face ellipse e^* is

$$e^* = \arg \max_{e \in E} \epsilon^2$$

where search space E is a set of all possible ellipses produced by varying the three parameters (y, a, b) within some ranges centered at the parameters for the face ellipse as predicted by the Kalman filtering or simply those in the previous frame. Surprisingly, during tracking, the ellipse parameters changes from frame to next frame are rather small. The parameters ranges are therefore small, leading to rather quick detection of face ellipse.

Compared with the existing face detection methods, ours takes advantage of the following factors

- use the detected pupil locations to guide the detection of face

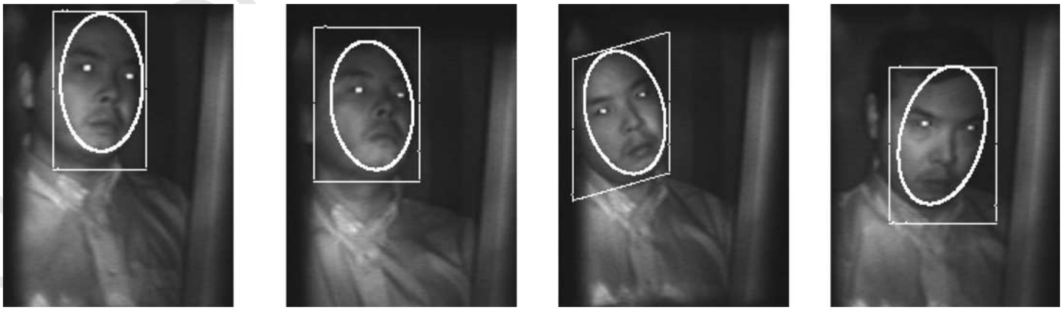


Fig. 8. Sample results of detected faces at different face orientations. Note the bright-pupil effects. The large white rectangles give approximate face locations determined based on the detected pupil locations. The white ellipses represent the accurate face locations resulted through a constrained gradient maximization.

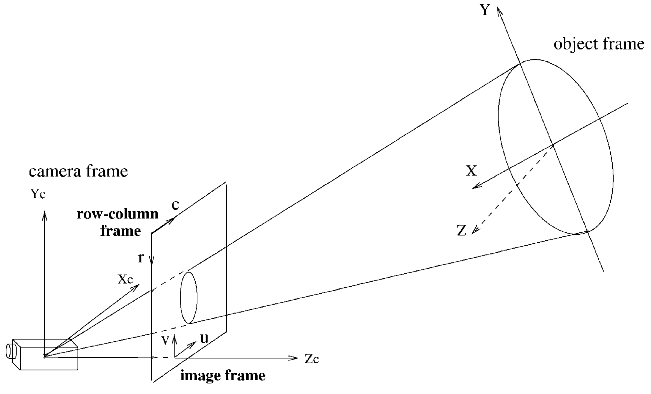


Fig. 9. Camera perspective projection model.

- assume weak perspective projection, so that the symmetric property of the 3D face can be used in 2D image
- use intensity gradients.

Experiments, however, shown our face detection and tracking method may fail, if the face is in oblique angle with respect to the camera. This is caused by the loss of pupils due to occlusion and the invalidity of the weak perspective projection assumption.

Finally, assuming small change (smooth motion) in size, shape, and location between two faces in two consecutive frames, we can efficiently track the face from frame to frame via Kalman filtering. The face tracking works well, if the face can be detected accurately in the first frame. It does not always work well, depending on illumination conditions (light interference from sources other than the IR light) and face orientations. Sometimes, we have to manually give an initial position of the face. Fig. 8 gives a sequence of image frames with detected face ellipses superimposed on the original images.

5. Theories for 3D face pose estimation

5.1. Notations

Consider a single image of a face in general pose. Let X–Y–Z represent the object coordinate frame. The origin of the object frame is located at the center of the 3D face ellipse, and the X and Y axes are along the major and minor axes of the ellipse. The Z axis is perpendicular to the 3D ellipse plane as shown in Fig. 9. Let X_c–Y_c–Z_c be the camera coordinate frame, with its origin located at the camera optical center. Assume X_c and Y_c axes are aligned along the horizontal and vertical directions in the image, respectively, and Z_c along the optical axis of the camera perpendicular to the image plane. Let u–v be the image coordinate frame centered at the principle point, with u and v axes parallel to X_c and Y_c, respectively. The row–column coordinate frame (pixels) is located at the upper left corner of the image, with row axis pointing downwards and column axis points from left to right. Fig. 9 gives different coordinate frames assignment. Let $\mathbf{X}_m = [X\ Y\ Z]^t$ be the coordinates of a 3D point in object frame, and $\mathbf{U} = (u\ v)^T$ be the coordinates of the corresponding image point in the image frame. Let $\mathbf{X}_c = [X_c\ Y_c\ Z_c]^T$ be the coordinates of $\mathbf{X}_m$ in the camera frame, and $\mathbf{P} = [c\ r]^T$ be the corresponding coordinates of $\mathbf{U}$ in the row–column frame in pixels.

5.2. Projection equation of points

Between camera frame and object frame, there exists the rigid body transformation

$$\mathbf{X}_c = \mathbf{R}\mathbf{X}_m + \mathbf{T} \tag{2}$$

where $\mathbf{R}$ is a rotation matrix and $\mathbf{T}$ is a translation vector. They characterize the relative orientation and position of object frame to the camera frame. In the object frame, the coordinates of a point on the 3D ellipse may be represented as $(X, Y, 0)^t$. Denote the ith column of the rotation matrix $\mathbf{R}$ by r_i, thus from Eq. (1) we have

$$\begin{pmatrix} X_c \\ Y_c \\ Z_c \end{pmatrix} = [\,\mathbf{r}_1 \quad \mathbf{r}_2 \quad \mathbf{r}_3 \quad \mathbf{T}\,] \begin{pmatrix} X \\ Y \\ 0 \\ 1 \end{pmatrix} = [\,\mathbf{r}_1 \quad \mathbf{r}_2 \quad \mathbf{T}\,] \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} \tag{3}$$

Between image frame and camera frame, we have:

$$\begin{pmatrix} u \\ v \\ f \end{pmatrix} = \lambda \begin{pmatrix} X_c \\ Y_c \\ Z_c \end{pmatrix} \tag{4}$$

where λ is a scalar and f is the camera focal length. Between row–column frame and image frame, we have

$$\begin{pmatrix} c \\ r \end{pmatrix} = \begin{pmatrix} s_x & 0 \\ 0 & s_y \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} + \begin{pmatrix} u_0 \\ v_0 \end{pmatrix} \tag{5}$$

where s_x, s_y are scale factor (pixels/mm) in image u, v axis, u_0, v_0 are the coordinates of the principle point in pixels relative to the row–column frame. Eq. (5) can be equivalently rewritten

$$\begin{pmatrix} c \\ r \\ 1 \end{pmatrix} = \begin{pmatrix} s_x & 0 & u_0/f \\ 0 & s_y & v_0/f \\ 0 & 0 & 1/f \end{pmatrix} \begin{pmatrix} u \\ v \\ f \end{pmatrix} = \frac{1}{f} \begin{pmatrix} s_x f & 0 & u_0 \\ 0 & s_y f & v_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} u \\ v \\ f \end{pmatrix} \tag{6}$$

Denote

$$\mathbf{W} = \begin{pmatrix} s_x f & 0 & u_0 \\ 0 & s_y f & v_0 \\ 0 & 0 & 1 \end{pmatrix}$$

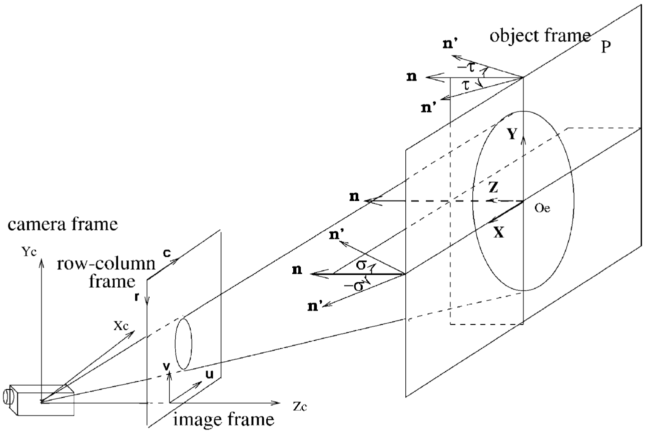


Fig. 10. Face ellipse rotation angles: pitch (σ) and yaw (τ).

where **W** is called the camera intrinsic matrix. Assume we are dealing with a calibrated camera and **W** is therefore known. Substituting Eq. (4) to Eq. (6) yields

$$\lambda\begin{pmatrix} c \\ r \\ 1 \end{pmatrix} = \mathbf{W}\begin{pmatrix} X_\mathrm{c} \\ Y_\mathrm{c} \\ Z_\mathrm{c} \end{pmatrix} \tag{7}$$

Substituting Eq. (3) to Eq. (7), we have

$$\lambda\begin{pmatrix} c \\ r \\ 1 \end{pmatrix} = \mathbf{W}[\,\mathbf{r}_1 \quad \mathbf{r}_2 \quad \mathbf{T}\,]\begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} \tag{8}$$

Denote $\mathbf{M} = [\mathbf{r}_1\ \mathbf{r}_2\ \mathbf{T}]$, **M** is called the camera extrinsic matrix and Eq. (8) is rewritten

$$\lambda\begin{pmatrix} c \\ r \\ 1 \end{pmatrix} = \mathbf{WM}\begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} \tag{9}$$

Eq. (9) is the projection equation that characterizes the relation between an image ellipse point and the corresponding 3D ellipse point.

5.3. Projection equation of ellipses

Let **Q** be a 3 × 3 matrix representing the 3D ellipse in object frame, **A** be a 3 × 3 matrix for the image ellipse, the equations of the image ellipse and the 3D ellipse are

$$\begin{pmatrix} c \\ r \\ 1 \end{pmatrix}^{\mathrm{T}} \mathbf{A}\begin{pmatrix} c \\ r \\ 1 \end{pmatrix} = 0 \tag{10}$$

$$\begin{pmatrix} X \\ Y \\ 1 \end{pmatrix}^{\mathrm{T}} \mathbf{Q}\begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} = 0 \tag{11}$$

Substituting Eq. (9) to Eq. (10) leads to

$$\lambda\left(\mathbf{WM}\begin{pmatrix} X \\ Y \\ 1 \end{pmatrix}\right)^{\mathrm{T}} \mathbf{A}\left(\mathbf{WM}\begin{pmatrix} X \\ Y \\ 1 \end{pmatrix}\right) = 0 \tag{12}$$

Eq. (12) can be rewritten

$$\begin{pmatrix} X \\ Y \\ 1 \end{pmatrix}^{\mathrm{T}} \lambda\mathbf{M}^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\mathbf{AWM}\begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} = 0 \tag{13}$$

From Eqs. (11) and (13), we have

$$\mathbf{Q} = \lambda\mathbf{M}^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\mathbf{AWM} \tag{14}$$

here, λ is an unknown scale factor. Denote $\mathbf{B} = \mathbf{W}^{\mathrm{T}}\mathbf{AW}$, this yields

$$\mathbf{Q} = \lambda\mathbf{M}^{\mathrm{T}}\mathbf{BM} \tag{15}$$

Let the length of major and minor axis of the 3D ellipse be a and b, respectively, then based on the location of the object frame, **Q** may be parameterized as

$$\mathbf{Q} = \begin{pmatrix} 1/a^2 & 0 & 0 \\ 0 & 1/b^2 & 0 \\ 0 & 0 & -1 \end{pmatrix}$$

therefore Eq. (15) can be rewritten

$$\lambda\begin{pmatrix} 1/a^2 & 0 & 0 \\ 0 & 1/b^2 & 0 \\ 0 & 0 & -1 \end{pmatrix} = \begin{pmatrix} \mathbf{r}_1^{\mathrm{T}}\mathbf{Br}_1 & \mathbf{r}_1^{\mathrm{T}}\mathbf{Br}_2 & \mathbf{r}_1^{\mathrm{T}}\mathbf{BT} \\ \mathbf{r}_2^{\mathrm{T}}\mathbf{Br}_1 & \mathbf{r}_2^{\mathrm{T}}\mathbf{Br}_2 & \mathbf{r}_2^{\mathrm{T}}\mathbf{BT} \\ \mathbf{T}^{\mathrm{T}}\mathbf{Br}_1 & \mathbf{T}^{\mathrm{T}}\mathbf{Br}_2 & \mathbf{T}^{\mathrm{T}}\mathbf{BT} \end{pmatrix} \tag{16}$$

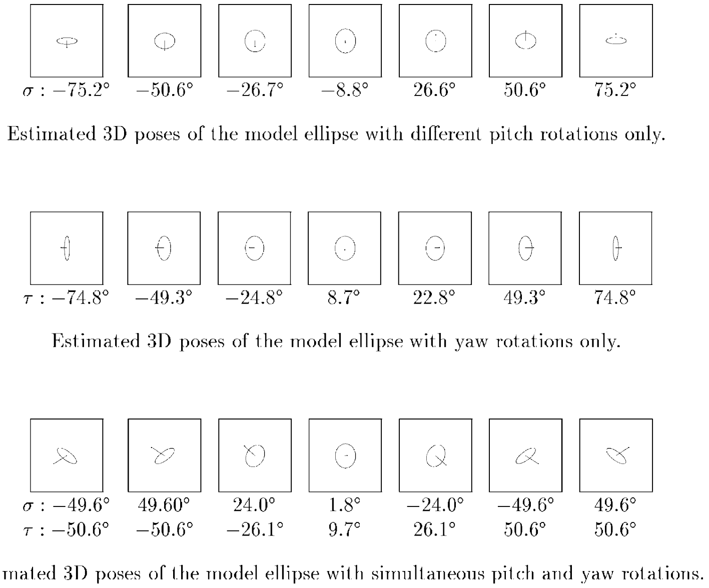
Note that **B** is known for a calibrated camera, Eq. (16) is the basic geometric constraint of a 3D ellipse and its projection. In Eq. (16), since the matrix is symmetric, there are six constraints. However, there are a total of nine unknowns: three rotation angles, three translation variables, the ellipse major axis length a, the minor axis length b, and the scale factor λ. To solve for these unknowns, additional information is necessary.

5.4. Face pose characterization

We will characterize the 3D face pose by a rotation matrix **R**, which specifies the relative orientation of the face normal relative to the camera frame, and a translation vector **T**, which specifies the position of the 3D ellipse (its center) relative to the camera frame.

5.4.1. The rotation matrix *R*

In general, a face rotation may be characterized by three Euler angles roll, pitch, and yaw as shown in Fig. 2. Since the roll angle has been independently determined as discussed in Section 3, we can now assume that face



Estimated 3D poses of the model ellipse with different pitch rotations only.

Estimated 3D poses of the model ellipse with yaw rotations only.

Estimated 3D poses of the model ellipse with simultaneous pitch and yaw rotations.

Fig. 11. Pose estimation for the synthetic ellipses. The line in the ellipse represents the estimated ellipse normal. The number(s) below each image is the estimated yaw or/and pitch angles.

orientation (face normal) can be characterized by two angles: pitch and yaw.

Given this characterization and assuming the object frame initially coincide with the camera frame, **R** can be obtained from three successive Euler rotations, i.e. rotating the object frame around Y_c axis by 180°, around X_c by σ°, and finally around Y_c axis again by τ°. σ and τ lie in the range $[-90°, 90°]$ as shown in Fig. 10.

Thus, **R** can be expressed as follows,

$$\mathbf{R} = \mathbf{R}_1^{\mathrm{T}}\mathbf{R}_\sigma^{\mathrm{T}}\mathbf{R}_\tau^{\mathrm{T}} = \begin{pmatrix} \cos\pi & 0 & -\sin\pi \\ 0 & 1 & 0 \\ \sin\pi & 0 & \cos\pi \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\sigma & \sin\sigma \\ 0 & -\sin\sigma & \cos\sigma \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} \cos\tau & 0 & -\sin\tau \\ 0 & 1 & 0 \\ \sin\tau & 0 & \cos\tau \end{pmatrix}^{\mathrm{T}} = \begin{pmatrix} -\cos\tau & 0 & -\sin\tau \\ \sin\sigma\sin\tau & \cos\sigma & -\sin\sigma\cos\tau \\ \cos\sigma\sin\tau & -\sin\sigma & -\cos\sigma\cos\tau \end{pmatrix} \quad (17)$$

5.4.2. The constraint equations

Let c be the ratio between the major and minor axis of the 3D ellipse, i.e. $c = a^2/b^2$, from Eq. (16), the first 2×2 submatrix should yield

$$\begin{pmatrix} \mathbf{r}_1^{\mathrm{T}}\mathbf{B}\mathbf{r}_1 & \mathbf{r}_1^{\mathrm{T}}\mathbf{B}\mathbf{r}_2 \\ \mathbf{r}_2^{\mathrm{T}}\mathbf{B}\mathbf{r}_1 & \mathbf{r}_2^{\mathrm{T}}\mathbf{B}\mathbf{r}_2 \end{pmatrix} = \lambda \begin{pmatrix} 1/a^2 & 0 \\ 0 & 1/b^2 \end{pmatrix} \quad (18)$$

Thus, two constraint equations are obtained

$$\mathbf{r}_1^{\mathrm{T}}\mathbf{B}\mathbf{r}_2 = 0; \qquad \mathbf{r}_1^{\mathrm{T}}\mathbf{B}\mathbf{r}_1 - c\mathbf{r}_2^{\mathrm{T}}\mathbf{B}\mathbf{r}_2 = 0 \quad (19)$$

where r_1 and r_2 may be obtained from Eq. (17) as

$$r_1 = \begin{pmatrix} -\cos\tau \\ \sin\sigma\sin\tau \\ \cos\sigma\sin\tau \end{pmatrix}; \qquad r_2 = \begin{pmatrix} 0 \\ \cos\tau \\ -\sin\tau \end{pmatrix}$$

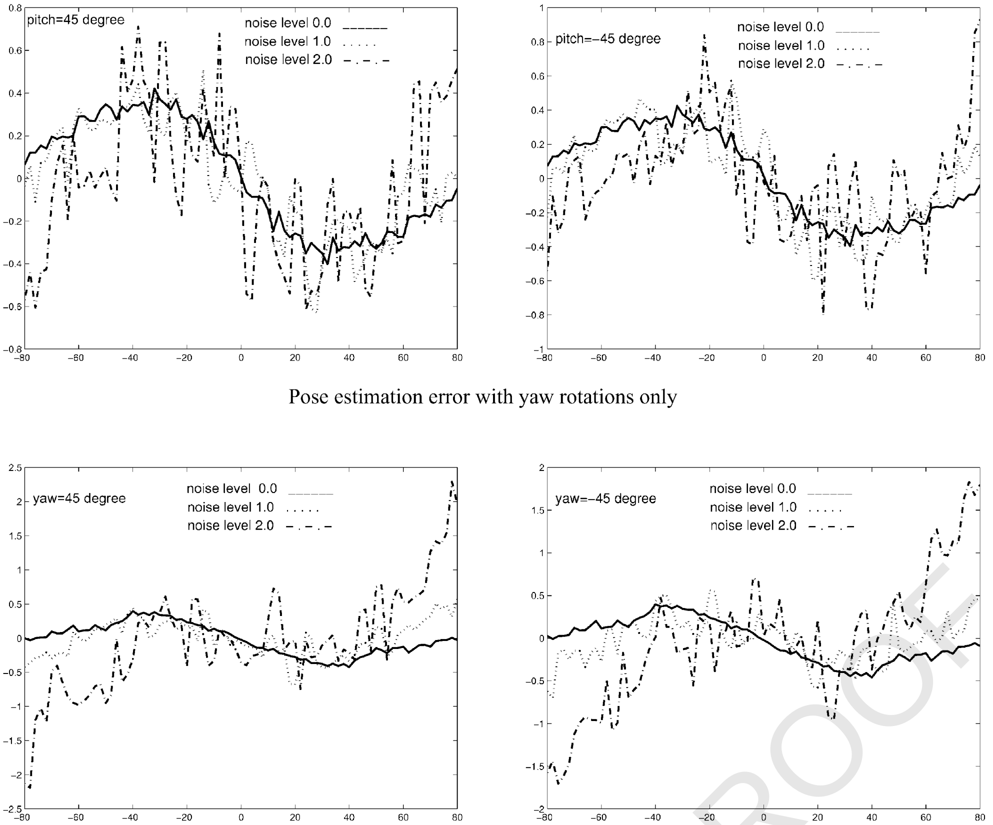
It is apparent that we can solve for pitch and yaw from Eq. (19), since there are two equations for two unknowns. The solution is, however, iterative. The two angles can be solved using Newton's method. Our experiments indicated that the non-linear solution is not very sensitive to initial estimates. The initial estimates of 0° for both pitch and yaw seem to be sufficient for the non-linear solution to converge correctly and quickly. We are currently exploring the possibility of obtaining the initial estimates from a closed form solution by assuming weak perspective projection. Moreover, given r_1 and r_2, the translation **T** can be calculated up to a scale factor as follows. From (1, 3) and (2, 3) elements of the matrices on both sides of Eq. (16), we have

$$\begin{pmatrix} \mathbf{r}_1^{\mathrm{T}}\mathbf{B}\mathbf{T} \\ \mathbf{r}_2^{\mathrm{T}}\mathbf{B}\mathbf{T} \end{pmatrix} = \lambda \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad (20)$$

Let $\mathbf{T} = (\mathbf{t}_x\ \mathbf{t}_y\ \mathbf{t}_z)^{\mathrm{T}}$, $\mathbf{t}_x/\mathbf{t}_z$, $\mathbf{t}_y/\mathbf{t}_z$ can be solved for by the above two equations analytically.

6. Experimental results

The proposed algorithm has been tested extensively using synthetic ellipse data, actual ellipse data, and human face image data. We summarize the results for each experiment as follows.



Pose estimation error with yaw rotations only

Pose estimation errors with pitch rotation only

Fig. 12. Pitch and yaw estimation error for a synthetic ellipse.

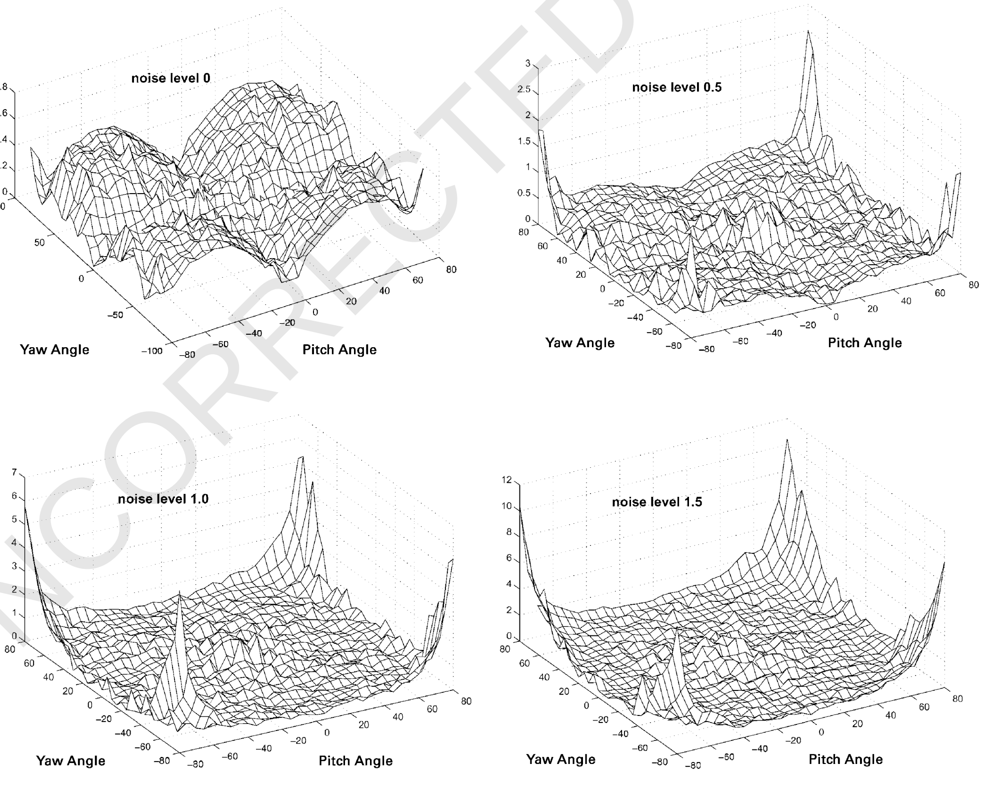


Fig. 13. 3D Plot of the pose estimation errors versus simultaneous yaw and pitch rotations for the synthetic ellipse under different noise levels.

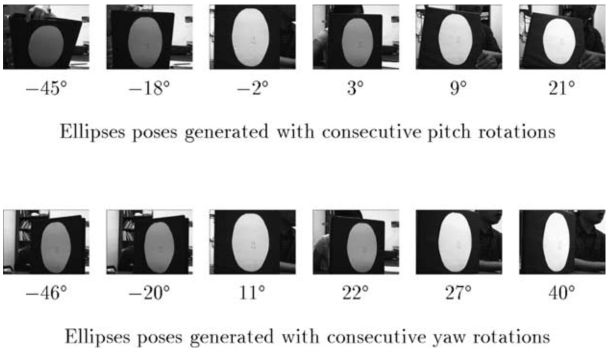

−45°
−18°
−2°
3°
9°
21°
Ellipses poses generated with consecutive pitch rotations
−46°
−20°
11°
22°
27°
40°
Ellipses poses generated with consecutive yaw rotations

Fig. 14. The estimated orientations for an actual ellipse from its image. The number below each image represents the estimated pitch or yaw angle.

6.1. Experiment results with synthetic ellipse data

It is difficult to obtain the ground-truth for the rotation angles of a face. To evaluate our algorithm, we used artificial images of a computer-generated ellipse viewed from different orientations to evaluate our pose estimation technique and to study its sensitivity to image noise. Specifically, we generate a 3D object model ellipse with different lengths of major and minor axes. About 1153 artificial images of the model ellipse were generated using a full perspective projection at different view distances and with different combinations of pitch and yaw angles. Specifically, the images of the 3D model ellipse were generated first with σ and τ varying over the range (−90°,90°) with 2° interval separately and then with both σ and τ varying over the range (−90°,90°) simultaneously with 5° interval. The proposed algorithm was used to estimate 3D model ellipse orientation from each image. Experimental results are summarized in Fig. 11. The directions of arrows in the images are the estimated normals of the model ellipse. The numbers shown below each image are the estimated orientations.

To study the sensitivity of the algorithm to image noise, the imaged ellipse locations were corrupted by zero-mean Gaussian noise with standard deviations 0.5, 1.0, 1.5, 2.0 pixels, respectively. The model ellipse poses were then re-estimated from the perturbed images. The estimated orientations were compared with the ground-truth orientations. Fig. 12 shows the estimated pose errors for separate pitch and yaw rotation under different noise levels. Fig. 13 gives the 3D plot of the pose errors with simultaneous pitch and yaw rotations under different noise levels. It is clear from the two figures that the proposed algorithm is fairly accurate and stable. As we can see from Fig. 12, without Gaussian noise, the maximum estimation error is less than 0.5°. When noise is increased to 2 pixels, the maximum estimation error is about 2°, which demonstrates that the algorithm is fairly robust to i.i.d Gaussian noise. Fig. 13 shows when yaw or pitch is close to −90° or 90°, the estimation errors increase considerably with the noise level. This is due to the fact that the ratio is extremely small and any perturbation with the image can lead to a significant increase with the estimated orientations.

6.2. Actual ellipse data

In order to further investigate the performance of the proposed algorithm, we applied the algorithm to estimate the pose of an actual ellipse from its images generated with different pitches and yaws using a calibrated camera. The results are summarized in Fig. 14. The proposed algorithm was used to compute the poses of the model ellipses from their images. The computed rotation angle for each image is shown below each image in Fig. 14. The estimated results are consistent with the perceived orientations.

6.3. Human face image data

The ultimate test of our algorithm is to study its performance with human faces. For this purpose, two image sequences of a male and a female with different pitches and yaws were captured. An ellipse detection is then performed on each face image to detect face. The detected ellipse is used to estimate the pose of the human face. These results are shown in Figs. 15 and 16. The computed two

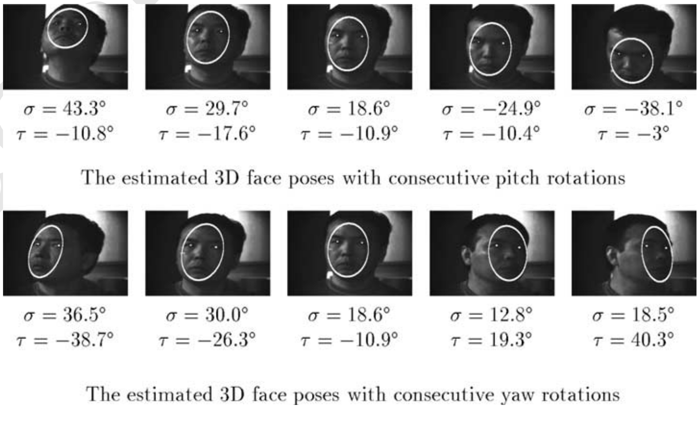


Fig. 15. The estimation of 3D face pose of a male subject. The estimated pitch and yaw angles are below each image. A video demo of our algorithms (for both pupil detection and face pose estimation) may be found at http://www.ecse.rpi.edu/Homepages/demos.html.

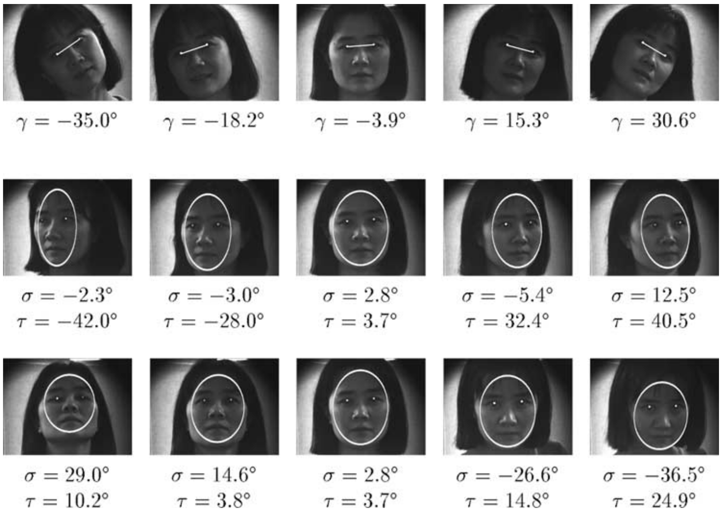

$\gamma = -35.0°$
$\gamma = -18.2°$
$\gamma = -3.9°$
$\gamma = 15.3°$
$\gamma = 30.6°$
$\sigma = -2.3°$ $\tau = -42.0°$
$\sigma = -3.0°$ $\tau = -28.0°$
$\sigma = 2.8°$ $\tau = 3.7°$
$\sigma = -5.4°$ $\tau = 32.4°$
$\sigma = 12.5°$ $\tau = 40.5°$
$\sigma = 29.0°$ $\tau = 10.2°$
$\sigma = 14.6°$ $\tau = 3.8°$
$\sigma = 2.8°$ $\tau = 3.7°$
$\sigma = -26.6°$ $\tau = 14.8°$
$\sigma = -36.5°$ $\tau = 24.9°$

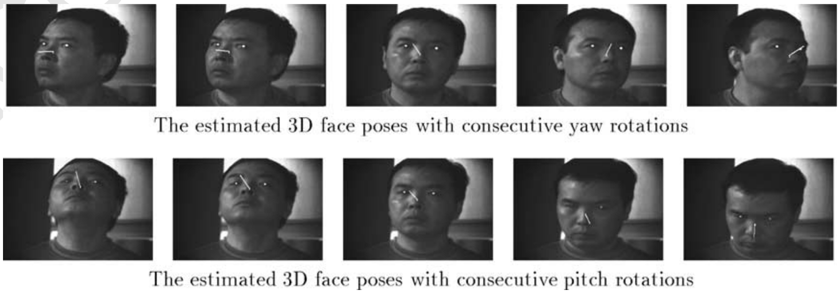
Fig. 16. The estimation of 3D face pose with consecutive roll rotations (top), yaw rotations (middle) and pitch rotations (bottom). The angles below each image are the estimated roll, yaw and pitch angles.

rotation angles of face pose are shown below each image. To further study the performance of our algorithm, we apply it to a sequence of image frames with a successive change in face orientation. Fig. 17 gives sample output with the estimated face orientation represented by the direction of a line superimposed on original images. Through visual inspection of both sets of images, we can conclude that the estimated face poses are in good agreement with actual face orientations.

7. Conclusions

In this paper, we describe a new approach for estimating and tracking 3D pose of a human face from a single monocular view of the face. The main contributions of the proposed approach include: (1) introduction of a constrained frontal face detection method based on ellipse fitting. The detected pupils are used to constrain the detection of the face ellipse; (2) introduction of a new 3D face pose determination technique from a single image of the face with full perspective projection. The algorithm achieves good experimental results. From synthetic data, without noise, the estimation errors are less than 0.5°, even if Gaussian noise with standard deviation 2.0 pixels is added, the estimate error is only 2.5°. For actual ellipse data and real-human face data, the estimated pose results are in good agreement with their perceived orientations. The experimental results demonstrated the proposed facial model is reasonable and that the proposed approach is reliable and stable.

We realize the importance of face ellipse detection for the proposed algorithm to work. A few factors that enable the described face ellipse detection to work reasonably include the knowledge of locations of two pupils, the use of motion for tracking, the knowledge of face aspect ratio, the weak perspective projection assumption, and finally the relatively uniform IR illumination. The IR illumination produces the bright-pupil effect, which significantly simplifies the face detection process. The face detection, however, does fail occasionally, especially when the face is largely slanted or tilted. Sometimes, we have to manually identify the initial face for the subsequent face tracking to work. Moreover, the face detection and tracking algorithm is also slow. As part of future work, we will focus on improving both the robustness and efficiency of the face detection algorithm. Specifically, we will study the use of human facial anthropometric statistics (e.g. ratio between distance of two eyes to the distance


The estimated 3D face poses with consecutive yaw rotations

The estimated 3D face poses with consecutive pitch rotations

Fig. 17. The estimation of 3D face pose of a male subject. The estimated face orientation is represented by the direction of a line superimposed on each image.

between eyes and the top of forehead) to help more accurately locate the image face ellipse. We will also experiment with a constrained version of the active parametric contour technique (e.g. SNAKE) to determine the exact face location and to track face.

References

[1] P. Ballard, G.C. Stockman, Controlling a computer via facial aspect, IEEE Transactions on Systems, Man, and Cybernetics 25 (4) (1995) 669–677.

[2] S. Birchfield, Elliptical head tracking using intensity gradients and color histograms, IEEE Conference on Computer Vision and Pattern Recognition (1998) 232–237.

[3] A. Blake, R. Curwen, A. Zisserman, A framework for spatio-temporal control in the tracking of visual contours, International Journal of Computer Vision 11 (2) (1993) 127–145.

[4] Q. Chen, H. Wu, T. Shioyama, T. Shimada, A robust algorithm for 3D head pose estimation, IEEE International Conference on Multimedia Computing and Systems (1999) 697–702.

[5] T. Darrell, B. Moghaddam, A.P. Pentland, Active face tracking and pose estimation in an interactive room, IEEE Computer Society Conference on Computer Vision and Pattern Recognition (1996) 67–72.

[6] A. Gee, R. Cipolla, Fast visual tracking by temporal consensus, Image and Vision Computing 14 (1996) 105–114.

[7] A.H. Gee, R. Cipolla, Determining the gaze of faces in images, Image and Vision Computing 12 (10) (1994) 639–647.

[8] A.H. Gee, R. Cipolla, Fast visual tracking by temporal consensus, Image and Vision Computing 14 (2) (1996) 105–114.

[9] G.G. Gordon, 3D Pose estimation of the face from video, NATO ASI Series, Series F, Computer and System Sciences 163 (1998) 433–445.

[10] K. Hattori, S. Matsumori, Y. Sato, Estimating pose of human face based on symmetry plane using range and intensity images, International Conference on Pattern Recognition 14 (2) (1998) 1183–1187.

[11] S.Y. Ho, H.L. Huang, An analytic solution for the pose determination of human faces from a monocular image, Pattern Recognition Letters 19 (1998) 1045–1054.

[12] T. Hogg, D. Rees, H. Talhami, Three-dimensional pose from two-dimensional images: a novel approach using synergetic networks, IEEE International Conference on Neural Networks 2 (1995) 1140–1144.

[13] A.T. Horprasert, Y. Yacoob, L.S. Davis, Computing 3D head orientation from a monocular image sequence, Proceedings of SPIE—The International Society for Optical Engineering 25th AIPR Workshop: Emerging Applications of Computer Vision 2962 (1996) 244–252.

[14] J. Qiang, Y. Xiaojie, Real time visual cues extraction for monitoring driver vigilance. International Workshop on Computer Vision Systems, Vancouver, Canada, 2001.

[15] C.W. Lee, A. Tsukamato, A visual interaction system using real-time face tracking, The 28th Asilomar Conference on Signals, Systems and Computers 2 (1994) 1282–1286.

[16] S.D. Ma, Conics-based stereo, motion estimation, and pose determination, International Journal of Computer Vision 10 (1) (1993) 7–25.

[17] H. Murase, S.K. Nayar, Visual learning and recognition of 3D objects from appearance, International Journal of Computer Vision 14 (1) (1995).

[18] A. Nikolaidis, I. Pitas, Facial feature extraction and determination of pose, http://www.citeseer.nj.nec.com/cs, pp. 1–6, 3333.

[19] M. Shah, P. Smith, N. da Vitoria Lobo, Monitoring head/eye motion for driver aleatness with one camera, http://www.citeseer.nj.nec.com/cs, pp. 1–7, 3333.

[20] T. Shioyama, Q. Chen, H. Wu, T. Shimada, 3D Head pose estimation using color information, Proceedings of IEEE International Conference on Multimedia Computing and Systems I (1999) 697–702.

[21] R. Rae, H.J. Ritter, Recognition of human head orientation based on artificial neural networks, IEEE Transactions on Neural Networks 9 (2) (1998) 257–265.

[22] R. Stiefelhagen, J. Yang, A. Waibel, A model-based gaze tracking system, IEEE International Joint Symposia on Intelligence and Systems (1996) 304–310.