Head Pose Estimation Based on Nonlinear Interpolative Mapping

Hwei-Jen Lin, Chen-Wei Chang, and I-Chun Pai
Department of Computer Science and Information Engineering
Tamkang University, Taipei, Taiwan, R.O.C.
hjlin@cs.tku.edu.tw, 696410314@s96.tku.edu.tw, 491193172@s91.tku.edu.tw

Abstract

The performance of face recognition systems depends on conditions being consistent, including lighting, pose and facial expression. To solve the problem produced by pose variation it is suggested to pre-estimate the pose orientation of the given head image before it is recognized. In this paper, we propose a head pose estimation method that is an improvement on the one proposed by N. Hu et al. [1]. The proposed method trains in a supervised manner a nonlinear interpolative mapping function that maps input images to predicted pose angles. This mapping function is a linear combination of some Radial Basis Functions (RBF). The experimental results show that our proposed method has a better performance than the method proposed by Nan Hu et al. in terms of both time efficiency and estimation accuracy.

Keywords: face recognition, head pose estimation, Isomap, Radial Basis Function (RBF), nonlinear interpolative mapping.

1. Introduction

Many businesses and managers of apartment buildings have taken advantage of the rapid development of technology to equip their buildings with access control systems as a security measure for both people and the contents of the building. Security ID cards are commonly used for access control purposes. However, this has some drawbacks. Tokens, ID cards or passwords are easily forgotten and are easy to duplicate. It is therefore important to find an access control system that is more secure and at the same time remains easy to use. In recent years, security systems based on a person's unique characteristics such as fingerprints, palm prints, iris, face, or voice recognition, are becoming the preferred methods. From all these various recognition techniques, face recognition is probably the most user friendly. The final result of using the face recognition technique depends on the quality of the face image, the variation in lighting, shadows, facial expression, pose etc. The first two factors can be alleviated by the appropriate image processing technique but not the rest. In this study, only the issue of the head pose will be addressed. To deal with the problem of pose variation, it is suggested to pre-estimate the angle of the head image before it is to be recognized. Head pose estimations are already widely applied in many webcam applications. In surveillance systems [2], combining the head pose with prior knowledge helps to analyze the intended motion of people. The head pose can also be used to deduce the concentration of people so as to improve human-computer interaction [3]. In addition, the head pose can help in visual communications [4], 3-D face reconstruction [5], facial expression analysis [6] etc. Many head pose estimation techniques have been proposed in the literature [1, 7, 8, 9, 10]. The image of a person's head usually contains a lot of information that is irrelevant for estimating the pose orientation. Some methods such as Principle Component Analysis (PCA) [11], Linear Discriminant Analysis (LDA) [12], Isomap [13], and Local Linear Embedding (LLE) [14] have been proposed to reduce the dimension of the images so as to delete the irrelevant information. Several techniques have been developed for head pose estimation, such as Geometrical face model [15], Support vector machines (SVMs) [16], Neural network [17], and Ada Boost [18]. Y. L. Tian et al. [7] proposed a system to estimate the pose of the by combining the three-dimensional head positions and the posture information. The system can be applied to an active face cataloger to obtain the best view of the face for surveillance. L. Chen et al. [8] used nonlinear interpolation to estimate the angle of the head and needed two directions to utilize the head image. The average head pose estimation error is about 4 degrees for a recognition range of angles ranging from -10° to +10°. Z. Guo et al. [9] used a mask transform and SVM classifier to correctly detect the face, and then analyzed the head pose using the Bilateral-projection Matrix Principle Component Analysis (BMPCA) algorithm. Although they claimed an accuracy rate of 98%, they did not tell how the accuracy rate was defined in their article. A. Lablack et al. [10] presented the use of a template based method to estimate the head pose. Their purpose is to extract the location of the person's gaze in the scene by estimating the head pose. N. Hu et al. [1] presented a

978-1-4244-5228-6/09/$26.00 ©2009 IEEE

scheme to estimate the head pose from either a video sequence or from individual images. Developed from Isomap, they learned a nonlinear embedding space (called a 2-D feature space) for different poses. They proposed a nonlinear interpolation to map new sequences or images into the 2-D feature space. However, evaluating and normalizing the unified embedding space is time-consuming, and the error produced is transmitted to the training process for the interpolative mapping function producing an augmented error. In this paper, we propose a method to estimate the head pose by training a nonlinear interpolation in a supervised manner. This method is an improvement of the method proposed by N. Hu et al..

The remainder of this paper is organized as follows: In Section 2, we provide a brief description of the head pose estimation method proposed by N. Hu et al. [1]. This is followed by a detailed introduction of our proposed method. In Section 3 we provide the experimental results and compares our proposed method to the method by N. Hu et al.. Finally, in Section 4 we draw our conclusions and provide suggestions for future work.

2. Head Pose Estimation

This section briefly describes the head pose estimation method proposed by N. Hu et al. [1], and then introduces our proposed method and describes how it improves the method proposed by N. Hu et al. in terms of time efficiency and estimation accuracy.

2.1 Nonlinear Embedding and Mapping

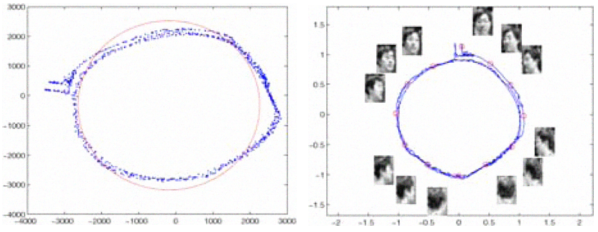
N. Hu et al. proposed a head pose estimation method based on nonlinear embedding and mapping to estimate head pose. First, $Y^P = \{y_1^P, \ldots, y_{n_p}^P\}$ the vector (image) sequence of length n_p in the original measurement space for person P was defined. ISOMAP was used to embed Y^P to a 2-D embedded space to introduce the corresponding coordinates $Z^p = \{z_1^P, \ldots, z_{n_P}^P\}$ for person P. Second, a Direct Ellipses-Specific Fitting proposed by A. Fitzgibbon et al. [19] was used to fit Z^p, resulting in the ellipse with parameters: center, major and minor axes, and orientation. Some linear operations, including rotation, scaling, translation, and reversion, were performed to normalize Z^p into Z^{*p}, so that the angle of each point in Z^{*p} is close to that of the corresponding head pose. The feature space where Z^{*p} belongs to is called a unified embedded space, and each point in Z^{*p} is called the feature point (or vector) of the corresponding image. Fig. 1(a) shows the result of the ellipse (red line) fitted on the sequence sample (blue point). Fig. 1(b) shows one of sequence normalizes into the 2-D feature space [1].

As soon as the feature point for each of the sample images has been evaluated a nonlinear interpolative mapping from the input images to the corresponding coordinates in the feature space can be trained. This nonlinear interpolative mapping is a linear combination of Radial Basis Functions (RBF). The Radial Basis Function $\psi(\bullet)$ is defined in (1), where c and σ_i^2 denote the center and the width of the RBF, respectively. The nonlinear interpolative mapping function is defined in (2), where ω_i's are real coefficients and $\mathbf{c}_i$'s are centers of the basis functions, and $\|\bullet\|$ is the norm in R^d.

$$\psi\left(\|y-c\|\right) = e^{-\frac{(\|y-c\|)^2}{2\sigma^2}} \tag{1}$$

$$f(y) = \omega_0 + \sum_{i=1}^{M} \omega_i \quad \psi\left(\| y - c_i \|\right) \tag{2}$$

All the persons' sequences were combined together, $\Gamma = \bigcup_{i=1}^{K} Y^{P_i} = \{y_1, y_2, \ldots, y_{n_0}\}$, and their corresponding coordinates in the feature space, $\Lambda = \bigcup_{i=1}^{K} Y^{P_i} = \{z_1^*, \ldots, z_{n_0}^*\}$, where n_0 is the total number of input images. For every point in the feature space, the nonlinear interpolative mapping function f was trained.



(a) (b)

Fig. 1. (a). The result of the ellipse (solid line) fitted on the sequence (dotted points); (b). A sequence whose 2-D embedded manifold has been normalized into the unified embedding space.

2.2 Nonlinear Interpolative Mapping – Supervised Learning

In the method proposed by N. Hu et al., evaluating and normalizing the unified embedding space is time-consuming, and the error it produced would be transmitted to the training process for the interpolative mapping function to produce an augmented error. Excluding the process of rotating and reversing the normalized ellipse so that the points corresponding to the left-pose head images is placed at a specific position of

the ellipse, respectively, this method is almost unsupervised. Due to the ease of sample collection, we were wondering if the method could be improved by skipping the embedding process using Isomap and replacing the training process with a supervised training scheme. This was convinced by the experimental results. The data we used were carefully collected. To collect a sequence of head images, a person is sitting on a rotating chair, a video is placed in the front of the person, and the chair is rotating by 360° at a certain constant speed while the video is taking the image. In the following subsections, we describe how to efficiently crop the head images from the sequence and how to estimate its orientation angle, and finally how to train the mapping function.

2.2.1 Uniformly Selecting Images and Estimating Pose Angles

Given a sequence of N head images of pose angles from 0° to 360°, numbered 1, 2, …, N. Suppose that A_0(= 1), A_1, A_2, and A_3 are the numbers for the images with pose angles 0°, 90°, 180°, and 270°, respectively. We divide the image sequence into four subsequences numbered $A_0 \sim A_1$-1, $A_1 \sim A_2$ -1, $A_2 \sim A_3$ -1, and $A_3 \sim$ N, including images with pose angles in the four quadrant, respectively. As described in (3), we uniformly select q images from each subsequence to obtain $m = 4q$ sample images, namely $s_1, s_2, \ldots, s_m$. The pose angle θ_j for each sample image s_j can be estimated as shown in (4).

$$s_{r*q+i} = A_r + round(\frac{A_{r+1} - A_r}{q} * (i - 1)), \quad r = 0 \sim 3,\ i = 1 \sim q \tag{3}$$

$$\theta_{r*q+i} = r * \pi / 2 + \frac{\pi}{2q} * (i - 1), \quad r = 0 \sim 3,\ i = 1 \sim q \tag{4}$$

2.2.2 Cropping Images in a Sequence

For efficiently cropping the images in all the sequences, we randomly select sequences and for each image sequence P, we crop the four images with pose angles 0°, 90°, 180°, and 270° (the front, left, back, and right pose images), into rectangular regions R_1^P, R_2^P, R_3^P, and R_4^P using horizontal projection and vertical projection. Fig. 2 shows the result of the cropped image of R_1^P for an individual.

For the four cropped regions R_1^P, R_2^P, R_3^P, and R_4^P, let (X^P_{i1}, X^P_{i2}) and (Y^P_{i1}, Y^P_{i2}) denote the intervals the region R_i^P across in the x-axis and the y-axis, respectively, and then denote $R_i^P = [(X^P_{i1}, X^P_{i2}), (Y^P_{i1}, Y^P_{i2})]$.We then evaluate the minimum rectangular region $R^P = [(X^P_{min}, X^P_{max}), (Y^P_{min}, Y^P_{max})]$, that covers the 4 regions as follows (see Fig. 3):

$$X^P_{min} = \min(X^P_{11}, X^P_{21}, X^P_{31}, X^P_{41})$$
$$X^P_{max} = \max(X^P_{12}, X^P_{22}, X^P_{32}, X^P_{42})$$
$$Y^P_{min} = \min(Y^P_{11}, Y^P_{21}, Y^P_{31}, Y^P_{41})$$
$$Y^P_{max} = \max(Y^P_{12}, Y^P_{22}, Y^P_{32}, Y^P_{42})$$

Evaluate the off-set ratios of the region R_1^P to the region R^P as follows:

$$\triangle X^P_{left} = (X^P_{11} - X^P_{min}) / (X^P_{12} - X^P_{11})$$
$$\triangle X^P_{right} = (X^P_{max} - X^P_{12}) / (X^P_{12} - X^P_{11})$$
$$\triangle Y^P_{down} = (Y^P_{11} - Y^P_{min}) / (Y^P_{12} - Y^P_{11})$$
$$\triangle Y^P_{up} = (Y^P_{max} - Y^P_{12}) / (Y^P_{12} - Y^P_{11}).$$

Then evaluate the maximum off-set ratios as follows:

$$\triangle X_{left} = \max_{1 \le P \le K} \triangle X^P_{left}$$
$$\triangle X_{right} = \max_{1 \le P \le K} \triangle X^P_{right}$$
$$\triangle Y_{down} = \max_{1 \le P \le K} \triangle Y^P_{left}$$
$$\triangle Y_{up} = \max_{1 \le P \le K} \triangle Y^P_{right}$$

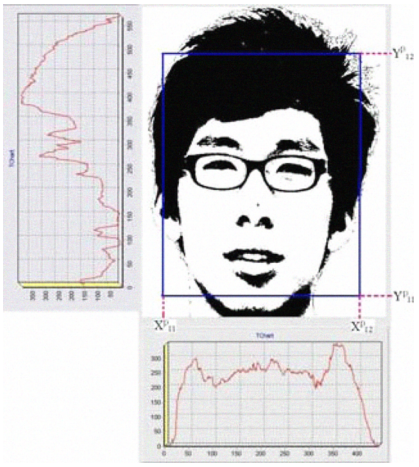


Fig. 2. The graph on the left: the horizontal projection; the graph at the bottom: the vertical projection; rectangular region (blue line): the cropped region R_1^P for the front face image for person P.

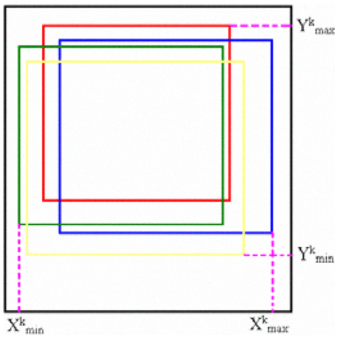


Fig. 3. Red, blue, yellow, and green frames are cropped regions, $R^P_1, R^P_2, R^P_3, R^P_4$, and R^P, respectively.

With the maximum off-set ratios, to crop the images in each sequence P', we need only determine the cropping region $R^{P'}_1 = [(X^{p'}_{11}, X^{p'}_{12}), (Y^{p'}_{11}, Y^{p'}_{12})]$ by x- and y-projection for the image of front pose (angle 0°) and then simply determine the cropping region $R^{P'} = [(X^{p'}_1, X^{p'}_2), (Y^{p'}_1, Y^{p'}_2)]$, as shown in Fig. 4, for each of the remaining image with the maximum off-set ratios as follows. Finally, each cropped image is normalized into a fixed size of 35 × 40.

$$X^{p'}_1 = X^{p'}_{11} - \triangle X_{left}(X^{p'}_{12} - X^{p'}_{11}),$$
$$X^{p'}_2 = X^{p'}_{12} + \triangle X_{right}(X^{p'}_{12} - X^{p'}_{11}),$$
$$Y^{p'}_1 = Y^{p'}_{11} - \triangle Y_{down}(Y^{p'}_{12} - Y^{p'}_{11}),$$
$$Y^{p'}_2 = Y^{p'}_{12} + \triangle Y_{up}(Y^{p'}_{12} - Y^{p'}_{11}).$$

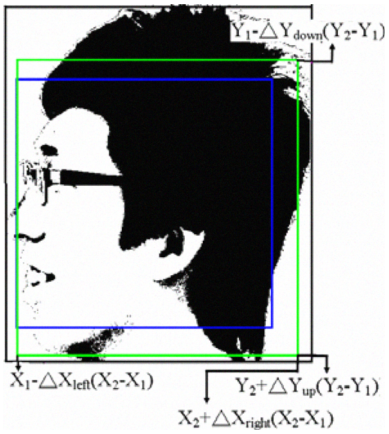


Fig. 4. The green frame is the final cropped frame.

2.2.3 Training the Mapping Function

Let $U^P = \{u^P_1, ..., u^P_m\}$ denote a sequence of m head images for person P and $\Theta^P = \{\theta^P_1, ..., \theta^P_m\}$ be the pre-estimated corresponding pose angles, P = 1, 2, …, K. We then estimate the feature points $X^P = \{x^P_1, ..., x^P_m\}$ as $x^P_j = (\cos\theta^P_j, \sin\theta^P_j)$. We combine all the training sample sequences together, $Q = \bigcup_{P=1}^{K} U^P = \{u_1, ..., u_n\} \subseteq R^d$ and their corresponding coordinates in the feature space R^2. $T = \bigcup_{P=1}^{K} X^P = \{x_1, ..., x_n\}$, where $n = Km$ is the total number of training samples. We use the combined sample set and the corresponding feature point set to train a nonlinear interpolative mapping function defined in (2). To determine the center c_i and variance σ_i^2 for each of the M Radial Basis Functions (RBF), we cluster the sample images into M groups (clusters), and evaluate the mean c_i and width σ_i^2 for each group, as defined in (5) and (6), where v_{ij} is the jth sample in the cluster i and h_i is the number of elements in cluster i.

$$c_i = \frac{1}{h_i}\sum_{j=1}^{h_i} v_{ij}, \; i = 1, 2, \ldots, M \tag{5}$$

$$\sigma_i^2 = \frac{1}{h_i}\sum_{j=1}^{h_i} (v_{ij} - c_i)^2, \; i = 1, 2, \ldots, M \tag{6}$$

Let $f(u_i) = (f_1(u_i), f_2(u_i)) = x_i = (x_{i1}, x_{i2})$, as defined in (7). To determine the weights $\omega_{\alpha j}$, we merely have to solve the linear system given in (8), where the matrices Ω, Ψ, and X are given in (9)~(11), and $\psi_{ij} = \psi(\|u_i - c_j\|)$.

$$f_\alpha(u_i) = \sum_{j=0}^{M} \omega_{\alpha j} \bullet \psi(\|u_i - c_j\|) = x_{i\alpha}, i = 1, ..., n, \alpha = 1, 2 \tag{7}$$

$$\Omega\Psi = X \tag{8}$$

$$\Omega = \begin{pmatrix} \omega_{10} & \cdots & \omega_{1M} \\ \omega_{20} & \cdots & \omega_{2M} \end{pmatrix} \tag{9}$$

$$\Psi = \begin{pmatrix} \psi_{10} & \cdots & \psi_{n0} \\ \vdots & \psi_{ij} & \vdots \\ \psi_{1M} & \cdots & \psi_{nM} \end{pmatrix} \tag{10}$$

$$X = \begin{pmatrix} x_{11} & \cdots & x_{n1} \\ x_{12} & \cdots & x_{n2} \end{pmatrix} \tag{11}$$

The linear system given in (8) can be solved by the least squares approximation method. The least squares

solution for Ω is then given by $\Omega = X\Psi^{\Delta}$, where $\Psi^{\Delta} = \Psi^{T}(\Psi\Psi^{T})^{-1}$ is the pseudo inverse of Ψ. In the test phrase, for a given test image $\hat{u}$, we evaluate its mapping point $(\hat{x}_1, \hat{x}_2)$, where $x_\alpha = f_\alpha(\hat{u})$, α=1,2, and then estimate the pose angle as follows.

$$\theta = \begin{cases} \arctan(|x_2/x_1|) & \text{if } x_1 > 0 \text{ and } x_2 \geq 0 \\ \pi + \arctan(|x_2/x_1|) & \text{if } x_1 < 0 \text{ and } x_2 \geq 0 \\ 2\pi + \text{rctan}(|x_2/x_1|) & \text{if } x_1 > 0 \text{ and } x_2 < 0 \\ \pi + \arctan(|x_2/x_1|) & \text{if } x_1 < 0 \text{ and } x_2 < 0 \\ \pi/2 & \text{if } x_1 = 0 \text{ and } x_2 > 0 \\ 3\pi/2 & \text{if } x_1 = 0 \text{ and } x_2 < 0 \end{cases} \quad (12)$$

3. Experimental Results

We used image sequences of 19 (K = 19) persons for training. The image sequence for each individual consists of 800~1000 frames of 720 × 480 pixels. m = 400 images were uniformly extracted from each sequence. Fig. 5 shows 24 images selected from a sequence of head pose images for a person. A leave-one - out cross-validation is used to estimate the generalization ability of our proposed method.

Eight (M = 8) RBFs are used to form the nonlinear interpolative mapping function. The numbers of test images with errors in a variety of ranges for our method and the method proposed by N. Hu et al. are compared in Table 1, in which the range of pose angles considered is from 0° to 360°. We also trained the head images with pose angles in different recognition ranges and compare average errors produced by the two methods in Table. 2. Fig. 6 shows the distribution of the angles estimated by the two methods, compared with the predicted angles.

4. Conclusions and Future Work

We have proposed a supervised learning strategy to train a nonlinear mapping function that maps head images to a 2D feature space. With the mapping function we may estimate the pose angles for head images. The proposed method has been verified to superior to that proposed by N. Hu et al. in terms of both the time efficiency and estimation accuracy. The estimation accuracy of our method could be raised by improving the sample images with the use of an automatically rotating chair. The current research focuses on only the yaw rotation of heads. In the future, we will consider the pitch rotation of heads as well. We will also investigate some dimension reduction techniques such as PCA or LDA to rid the useless information from head images to further improve the performance. In addition, we will SVM to classify the mapping feature points to develop a head pose classifier.

Table 1. No. of test images with errors in a variety of ranges

Methods / *Ranges of error θ*	*Our method*	*N. Hu et al.*
0° ≤ 10°	*1326*	*998*
10° < θ ≤ 20°	*1074*	*838*
20° < θ ≤ 30°	*703*	*647*
30° < θ ≤ 40°	*336*	*429*
40° < θ ≤ 50°	*187*	*269*
50° < θ ≤ 60°	*106*	*171*
60° < θ ≤ 70°	*27*	*119*
70° < θ ≤ 80°	*24*	*72*
80° < θ ≤ 90°	*9*	*60*
θ > 90°	*8*	*197*

Table 2. Average errors over different recognition ranges of pose angles.

Methods / *Ranges of angles*	*Our method*	*N. Hu et al.*
0° ~ 360°	*18.5°*	*30°*
-90° ~ 90°	*10.21°*	*27.47°*
-60° ~ 60°	*8.25°*	*22.85°*

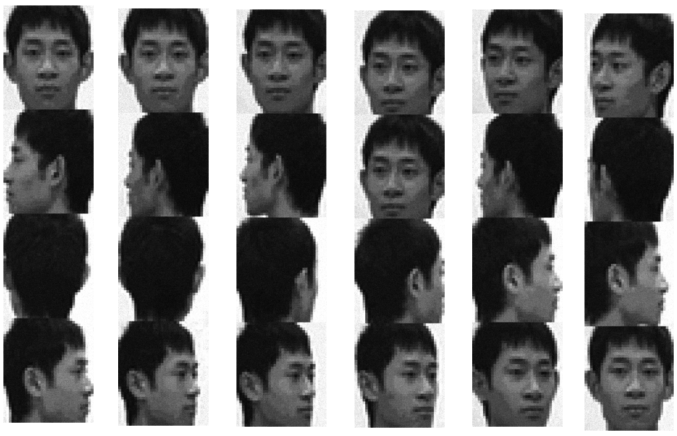


Fig. 5. 24 images selected from a sequence of head pose images for a person.

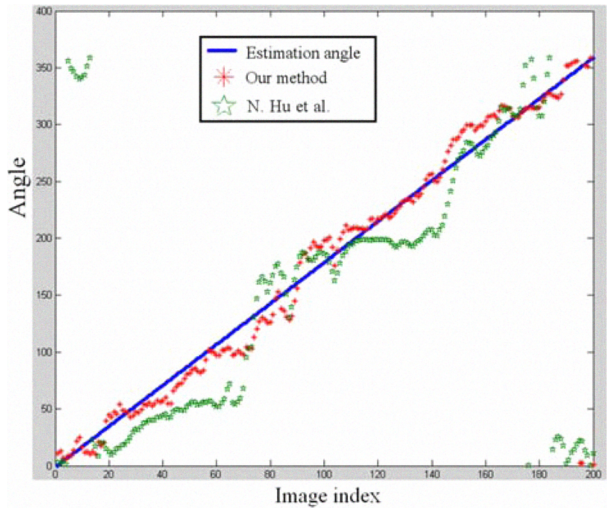


Fig. 6. Angles estimated by two methods. blue line: the predicted angles; red stars: angles estimated by our method; green stars: angles estimated by N. Hu et al..

5. References

[1] Nan Hu, Weimin Huang, and Surendra Ranganath, "Head Pose Estimation by Non-linear Embedded and Mapping," in *Proc. of IEEE International Conference on Image Processing* Vol. 2, pp. 342-345, September 2005.

[2] M. Doi and Y. Aoki, "Real-time Video Surveillance System Using Omni-directional Image Sensor and Controllable Camera," in *Proc. of SPIE, Real-Time Imaging VII*, Vol. 5012, pp. 1-9, April 2003.

[3] U. Weidenbacher, G. Layher, P. Bayerl, and H. Neumann, "Detection of head Pose and Gaze Direction for Human-computer Interaction," *Perception and Interactive Technologies*, Springer, Berlin, Heidelberg, Vol. 4021/2006, pp. 9-19, June 2006.

[4] B. Yip and J. Jin, "Pose Determination and Viewpoint Determination of Human Head in Video Conferencing Based on Head Movement," in *Proc. of 10th International Conference on Multimedia Model.*, Brisbane, Australia, pp. 130–135, January 2004.

[5] B. Yip and J. Jin, "3d Reconstruction of A Human Face with Monocular Camera Based on Head Movement," in *Proc. of Pan-Sydney Area Workshop VIP*, Darlinghurst, Australia, pp. 99-103, 2003.

[6] C. Chien, Y. Chang, and Y. Chen, "Facial Expression Analysis Under Various Head Poses," in *Proc. of 3rd IEEE Pacific Rim Conf. Multimedia*, Taiwan, Vol. 2532/2002, pp. 199-212, December 2002.

[7] Ying-Li Tian, Lisa Brown, and Jonathan Connell, "Absolute Head Pose Estimation from Overhead Wide-angle Cameras," in *Proc. of IEEE International Workshop on Analysis and Modeling of Faces and Gestures*, pp. 92-99, October 2003.

[8] Longbin Chen, Lei Zhang, Yuxiao Hu, Mingjing Li, and Hongjiang Zhang, "Head Pose Estimation Using Fisher Manifold Learning," *IEEE International Workshop on Analysis and Modeling of Faces and Gestures*, pp. 203-207, October 2003.

[9] Zhibo Guo, Huajun Liu, Qiong Wang, and Jingyu Yang, "A Fast Algorithm Face Detection and Head Pose Estimation for Driver Assistant System," in *Proc. of the 8th International Conference on Signal Processing*, Vol. 3, pp. 16-20, 2006.

[10] Adel Lablack, Zhongfei Zhang, and Chabane Djeraba, "Supervised Learning for Head Pose Estimation Using SVD and Gabor Wavelets," in *Proc. of 10th IEEE International Symposium on Multimedia*, pp. 592-596, December 2008.

[11] Wenyu Sun and Qiuqi Ruan, "Two-Dimension PCA for Facial Expression Recognition," in *Proc. of the 8th International Conference on Signal Processing*, Vol. 3, November 2007.

[12] Aleix M. MartõÂnez and Avinash C. Kak, "PCA versus LDA," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 23, No. 2, pp. 228-233, February 2001.

[13] J. Tenenbaum, V. de Silva, and J. Langford, "A Global Geometric Framework for Nonlinear Dimensionality Reduction," *Science*, 290(5500), pp. 2319-2323, December 2000.

[14] S. Roweis and L. Saul, "Nonlinear Dimensionality Reduction by Locally Linear Embedding," *Science*, 290(5500), pp. 2323-2326, December 2000.

[15] S. H. Jeng and H. Y. Liao, "An Efficient Approach for Facial Feature Detection Using Geometrical Face Model," in *Proc. of IEEE International Conference on Pattern Recognition*, pp. 426-430, 1996.

[16] C. A. Waring and X. W. Liu, "Face Detection Using Spectral Histograms and SVMs," *IEEE Trans. Systems, Man and Cybernetics*, 35(3), pp. 467-476, 2005.

[17] H. A. Rowley, S. Baluja, and T. Kanade, "Neural Network based Face Detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 20(1), pp. 23-38, 1998.

[18] P. Viola and M. Jones, "Rapid Object Detection Using A Boosted Cascade of Simple Features," in *Proc. of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 511-518, 2001.

[19] Andrew Fitzgibbon, Maurizio Pilu, and Robert B. Fisher, "Direct Least Square Fitting of Ellipses," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 21, No. 5, pp. 476-480, May 1999.