Affine Correspondence Based Head Pose Estimation for a Sequence of Images by Using a 3D Model

Guoyuan Liang, Hongbin Zha, Hong Liu
National Lab on Machine Perception
Peking University, People's Republic of China
{lianggy, zha, liuhong@cis.pku.edu.cn}

Abstract

T*his paper proposes a method of determining human head poses from a sequence of images. The main idea is to use some features in a 3D head model to generate a virtual fronto-parallel projection that satisfies conditions of affine approximation. Then the affine parameters between the virtual projection and input view are calculated. After that, rotation and translation parameters of the head are roughly estimated by a circle-ellipse correspondence technique based on the affine parameters. Finally, an iterative optimization algorithm is utilized further to refine the results. The accuracy is maintained by estimating reliability of the 2D-3D feature correspondences and weighting each factor of the optimization objective function. The system performance is also improved by applying a modified KLT technique to speed up the convergence during the face feature tracking process. Experimental results show that our method can accurately recover head poses in a wide range of head motion.*

1. Introduction

Head gestures play a very important role in people's daily communication. The estimation of head poses can facilitate many human-computer interactive applications, such as face identification, gesture understanding, expression recognition and model-based low bit-rate video coding.

Many approaches have been reported on this topic. Most of them can be classified into two categories: face property-based and model-based. Property-based methods assume there exists certain relationship between the 3D head pose and some properties of the 2D face images and they use a large number of training images to determine the relationship. Face properties may include image intensity, colors [1], gradient or some transformations of the image intensity [2,4,7,11]. Model-base methods commonly assume a model as the approximation of head, and obtain the pose parameters using the 2D-3D feature correspondences. The models may be a geometrical object such as a cylinder [12], some simple geometrical structures [3], or 3D head models obtained by range finders [5,10,13]. Since 3D head models can provide rich geometrical information, making it possible to recover the head pose with fewer features, the use of 3D models becomes more and more popular in recent research.

This paper presents a robust approach using affine correspondences and 3D head models to estimate the head poses. The main purpose of our work is to estimate the head pose parameters relative to the fronto-parallel position for the frames, given a sequence of head images and a 3D head model of a person. At the beginning, we manually select some features both in the model and the first frame of the image sequence, and construct a virtual fronto-parallel project of the features by using the 3D model. Then we improve the well-known KLT method [9] that can reliably track 2D face features in the following frames by optimizing some matching criterion with respect to the small inter-frame displacement. For each frame, we use the same technique described below to estimate the pose parameters. First, the affine parameters between the virtual projection and the input frame are estimated. Then the head pose parameters are roughly estimated by a circle-ellipse correspondence technique. Finally a nonlinear optimization process is utilized to refine the rough results. In order to make the system more robust, we estimate reliability of the 2D-3D feature correspondences and weight each factor of the optimization objective function.

As compared with the affine correspondence technique in previous works, there are three advantages in our method. First, adding an extra rotation, our method can recover all six pose parameters instead of the face plane normal alone. Second, introducing 3D head models into the system, we can construct a virtual fronto-parallel (it means the face plane is parallel to the image plane with the front side to the camera) 2D projection to avoid the acquisition of a real fronto-parallel image as the reference view. Third, We use certain 3D features to define the face plan and make it precisely parallel to the image plane. That can eliminate errors of affine parameters estimated by using the camera-captured "fronto-parallel" view that is not really fronto-parallel. The defined face plane also makes the plane assumption of affine transformation is effective even in a close distance from the camera. Experiments show that our method can work effectively and obtain accurate head pose estimation in a wide range of head motion.

Proceedings of the Sixth IEEE International Conference on Automatic Face and Gesture Recognition (FGR'04)
0-7695-2122-3/04 $ 20.00 © 2004 IEEE

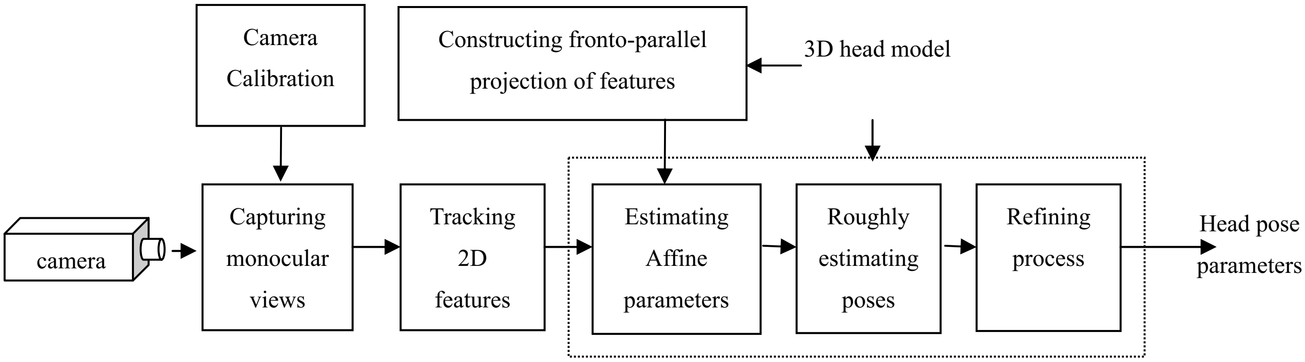

Fig. 1. System work flow diagram.

2. System Outline

Fig.1 shows the block diagram of our system. We have available a 3D head model which is generated with a *FastScan* laser scanner. Since the original data may be noisy, hence we use 3D modeling software *Polyworks* to filter out the noises and fill holes. Head images are captured by a *Mintron 64G-1K* camera and fed into the 2D feature tracking system. It is calibrated beforehand by using a self-calibration method [6]. After the rough estimation and a refining process, the accurate pose parameters are estimated.

We use six feature points including four corners of eyes and two corners of mouth. These points are nearly in the same plane. One extra point on the nose tip is also used to get a unique solution in the process of head pose estimation. These points are chosen here because there is rich texture information around them and hence they are easy to track [9]. The tracker needs to know the initial correspondences of features. We manually select the seven feature points both in the 3D models and the first frame of the image sequence.

3. 3D Head Pose Tracking

3.1 Virtual Fronto-parallel Projection

Affine correspondence depends on the face plane assumption. If we use 3D feature points mentioned above to define a face plane, it's easy to construct a virtual fronto-parallel projection of features. It can be achieved by two steps. First, we transform the feature points from the model coordinate system to a camera coordinate system with the face plane parallel to the image plane. Let $\mathbf{p}_1$ and $\mathbf{p}_4$ denote the outer corners of eyes, $\mathbf{p}_2$ and $\mathbf{p}_3$ the inner corners of eyes, $\mathbf{p}_5$ and $\mathbf{p}_6$ the corners of mouth, as shown in Fig.2 (a). $\mathbf{c}_1$ is the mean location of $\mathbf{p}_1$ $\mathbf{p}_2$$\mathbf{p}_3$ and $\mathbf{p}_4$, and $\mathbf{c}_2$ is the mean location of $\mathbf{p}_5$ and $\mathbf{p}_6$. Here define vectors $\mathbf{r}_1 = \mathbf{p}_3 - \mathbf{c}_1$ and $\mathbf{r}_2 = \mathbf{c}_2 - \mathbf{c}_1$. Then we can define the face plane that passes the point $\boldsymbol{c}_1$ and has $\mathbf{r}_2 \times \mathbf{r}_1$ as its normal. Let's define another point $\boldsymbol{p}_7$ at the vector $\mathbf{r}_2 \times \mathbf{r}_1$ with the distance between $\mathbf{c}_1$ and $\mathbf{p}_7$ being one. If we make $\mathbf{c}_1$, after a transformation, locate at the camera coordinate system's origin with $\mathbf{c}_1$-$\mathbf{p}_7$ as the z-axis direction, then there are three pairs of non-collinear points to determine the transformation parameters using Rodrigues Formula [6].

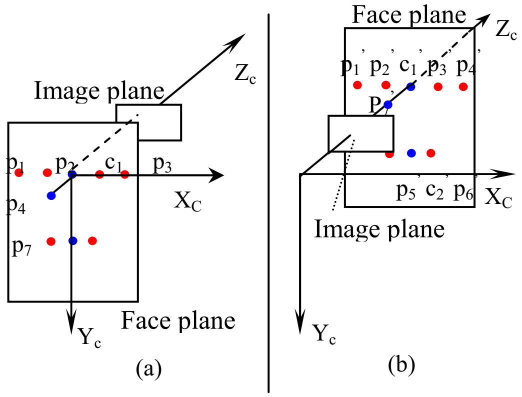

Fig. 2. Construction of virtual fronto-parallel projection.

Second, we translate the model to construct the virtual projection. If the camera is modeled as a pinhole one, then the coordinates of a 3D point $\mathbf{X} = (X \quad Y \quad Z)^T$ in the world coordinate system and its 2D projection coordinates $\mathbf{x} = (u \quad v)^T$ are related by

$$\begin{pmatrix} \mathbf{x} \\ 1 \end{pmatrix} = \lambda \mathbf{P} \begin{pmatrix} \mathbf{X} \\ 1 \end{pmatrix} \quad , \tag{1}$$

where is a scale factor and $\mathbf{P}$ is the 3×4 perspective projection matrix.

By choosing an arbitrary positive value t_z along the z-axis, as shown in Fig.2 (b), we can obtain the coordinates of the projected features using Eq. (1). Later, we will show the value of t_z does not affect the final estimation of the head pose parameters.

3.2 2D Feature tracking

We use a modified KLT algorithm to track the 2D features in the image sequence. In essence, the KLT method [9] assumes a linear approximation of the intensity in neighboring regions of the features. It will cause some error especially at the high-curvature points in the intensity function and need an iterative process to make the results converge. In order to reduce the number of iteration and improve the accuracy, we use a modified FFT-based KLT technique.

Assume the residue error function in the template window is

$$\varepsilon = \int_W [i_t(x,y) - \frac{\partial i_t(x,y)}{\partial x} d_x - \frac{\partial i_t(x,y)}{\partial y} d_y - i_{t+1}(x,y)]^2 dxdy \text{ , } \quad (2)$$

where $i_t(x,y)$ and $i_{t+1}(x,y)$ are image intensity in the window at time t and t+1. d_x and d_y are the motion parameters. The total error here equals to the energy of error function in the template window. After the FFT, according to Parseval Theorem, we have

$$\begin{aligned}\varepsilon &= \int_W [i_t(x,y) - \frac{\partial i_t(x,y)}{\partial x} d_x - \frac{\partial i_t(x,y)}{\partial y} d_y - i_{t+1}(x,y)]^2 dxdy \\ &= \int_W \left\| I_t(\omega_x,\omega_y) - I_{t,x}(\omega_x,\omega_y) d_x - I_{t,y}(\omega_x,\omega_y) d_y - I_{t+1}(\omega_x,\omega_y) \right\|^2 d\omega_x d\omega_y\end{aligned} \quad ,(3)$$

where $I_t(\omega_x,\omega_y), I_{t+1}(\omega_x,\omega_y), I_{t,x}(\omega_x,\omega_y), I_{t,y}(\omega_x,\omega_y)$ are the FFT results for $i_t(x,y) \bullet i_{t+1}(x,y)$ $\partial i_t(x,y)/\partial x, \partial i_t(x,y)/\partial y$, respectively. In fact, most of the high-curvature points correspond to the high frequency part of the image. If we perform optimization calculation using the low frequency component, it is expected to speed up the convergence while producing more accurate solution of the motion parameters. Because FFT has computational complexity depending on the window size, as a tradeoff, we choose 8 ×8 FFT windows in our method.

3.3 Affine Parameter Estimation

Under the condition of the face plane assumption, we can use a single global affine transformation to approximate the position change between projected features in the virtual fronto-parallel projection and those in the input view. Let $(\mathbf{p}_1, \mathbf{p}_2, \ \ \mathbf{p}_i, \ \ ,\mathbf{p}_N)$ denote the N feature points in the virtual projection, and $(\mathbf{p}'_1, \mathbf{p}'_2, \ \ \mathbf{p}'_i, \ \ ,\mathbf{p}'_N)$ denote the corresponding features in the input view obtained by 2D feature tracking. The affine relation between them can be written as

$$\mathbf{p}'_i = \mathbf{A}\mathbf{p}_i + \mathbf{b} \text{ ,} \quad (4)$$

where **A** is the linear component matrix and ***b*** the translation vector. The relation between the N pairs of features can be expressed as

$$\mathbf{Km} = \mathbf{u} \text{ ,} \quad (5)$$

where **m** is the affine parameter vector defined by

$$\mathbf{m} = \begin{pmatrix} A_{11} & A_{12} & b_1 & A_{21} & A_{22} & b_2 \end{pmatrix}^T \text{,}$$

and

$$\mathbf{K} = \begin{pmatrix} p_{1x} & p_{1y} & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & p_{1x} & p_{1y} & 1 \\ & & & & & \\ p_{Nx} & p_{Ny} & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & p_{Nx} & p_{Ny} & 1 \end{pmatrix} \text{ , } u = \begin{pmatrix} p'_{1x} \\ p'_{1y} \\ \\ p'_{Nx} \\ p'_{Ny} \end{pmatrix} . \quad (6)$$

We solve **m** in (5) by a linear least squares method. The estimated affine parameter vector is then given by

$$\tilde{\mathbf{m}} = (\mathbf{K}^T\mathbf{K})^{-1}\mathbf{K}^T\mathbf{u} \text{ .} \quad (7)$$

3.3 Rough Head Pose Estimation

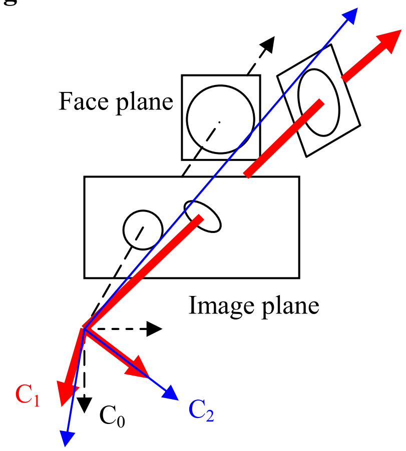

Fig.3. Pose estimation using the circle-ellipse correspondence technique.

Yao described an algorithm for the head pose estimation in [14] based on a circle-ellipse correspondence technique. It formed a cone whose intersection with the image plane was given by an ellipse equation obtained from the affine transformation. Then the camera coordinate system was rotated twice to make the intersection of the cone with a plane become a circle again. After that, the z-axis in the new camera coordinate system is consistent with the face plane normal, as shown in Fig.3. Here $\mathbf{C}_0$ denotes the original camera coordinate system, $\mathbf{C}_1$ the coordinate system after the first rotation, and $\mathbf{C}_2$ that after the second rotation.

Obviously the face plane normal can not determine the head pose uniquely because the face pose may vary on the face plane. Therefore we need a third rotation around the z-axis in the face plane.

Fig.4 illustrates the determination of the third rotation angle. Here we define an identification point **N**(0,1,f) at the unit notional circle in the image plane, which is superposed with the projected point **m**(0,1) of a certain point **M** in the face plane. When the face pose changes as shown in Fig.4(b), **M** is shifted to $\mathbf{M}'$.The projected point $\mathbf{m}'$ of $\mathbf{M}'$ can be determined using the affine parameters estimated from above steps.

Assume the unit notional circle plane rotates with the camera coordinate system. After the second rotation, **N** is shifted to $\mathbf{N_2}$. Then we translate the notional circle to make it center at the point **O** where the z-axis of $\mathbf{C}_1$ intersects with the image plane. (For the purpose of clarity, the notional circle center is placed at a distance from point **O** in Fig. 4(b)). In order to make the face pose in the notional circle plane consistent with that in the real face plane, point $\mathbf{N_2}$ should be rotated about the z-axis by a rotation angle to $\mathbf{N_3}$. $\mathbf{N_3}$ must be in the plane defined by **O**, $\mathbf{m}'$ and the COP (center of projection). Its projection $\mathbf{n_3}$, $\mathbf{m}'$ and **O** should locate at the same line. There are two solutions to. The angle that makes the distance between $\mathbf{m}'$ and $\mathbf{n_3}$ smaller is the correct solution.

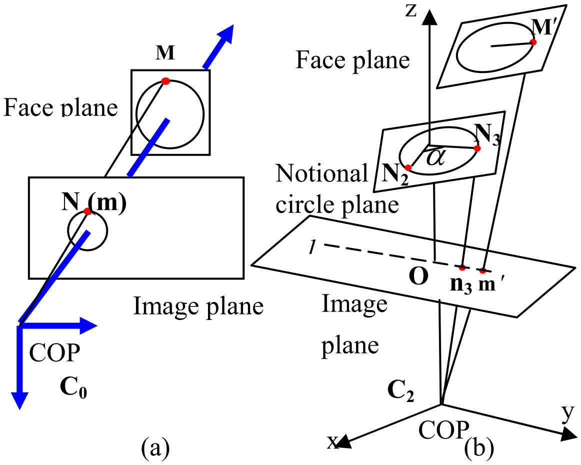

Fig. 4. Estimating the third rotation angle.

The first rotation matrix was determined by the eigenvectors of the diagonal matrix given in [14], and the second rotation angle was determined by

$$\cos^2\beta = (\lambda_2 - \lambda_3)/(\lambda_1 - \lambda_3)\,,$$

where $\lambda_1, \lambda_2, \lambda_3$ are the eigenvlaues of the diagonal matrix. Let **M** denote the diagonal matrix calculated by assumed translation $\mathrm{t_z}$ mentioned in section 3.1. If the real translation is $\hat{\mathrm{t}}_\mathrm{z} = k\mathrm{t_z}$, we can derive that the new diagonal matrix is k^2**M.** The eigenvalues of the matrix k^2**M** are k^2 times those of **M,** and the eigenvectors will not change at all. It means that $\hat{\mathrm{t}}_\mathrm{z}$ will not affect the estimation of the three rotation matrices.

Once the final rotation matrix is calculated, the translation parameters in Eq. (1) can be solved by a linear least squares method. Therefore, we recover all six head pose parameters.

3.4. Refinement of Head Poses

Because human face is roughly approximated by a plane, the above head pose estimation can not be very accurate. Therefore, we use the head pose parameters estimated from last section as the initial rough guess for an iterative optimization process that minimize the distance between the projected features and tracked features in the input view.

The objective function of the optimization is

$$D = \sum_{i=1}^{N} \omega_i [\left(x_i'(\mathbf{X},\mathbf{P}) - x_i\right)^2 + \left(y_i'(\mathbf{X},\mathbf{P}) - y_i\right)^2]\,, \qquad (8)$$

where x'_i(**X, P**) and y'_i(**X, P**) are the coordinates of a projected feature point **X** in the world coordinate system, **P** the projection matrix, x_i and y_i the coordinates of the corresponding feature points in the input view. $_i$ is the weight representing reliability of the 2D-3D feature correspondences, which will be discussed in the next section. The problem of optimization can be solved by the Levenburg-Marquet method [8].

Non-linear optimization with constrains of the rotation matrix is complicated. Therefore, using the Rodrigues Formula, we can find a three-dimensional vector **q** whose direction is consistent with the rotation axis, and the norm of **q** is equal to the rotation angle. Since three elements of **q** are independent, it is easy to implement the optimization process.

3.5. Enhancement of Robustness

Because of the inevitable errors in the process of tracking and initial localization of features, some 2D-3D features correspondences are not reliable. We apply a robust technique to weight factors of the objective function of optimization in Eq.(8). Since three non-linear pairs of 2-3D feature correspondences can determine an estimate of the head pose, for all combinations, the robust standard deviation is

$$\sigma = \Phi \cdot \min \underset{i=1 \quad N}{median} \sqrt{\left(x_i'(\mathbf{X},\mathbf{P}) - x_i\right)^2 + \left(y_i'(\mathbf{X},\mathbf{P}) - y_i\right)^2}\,. \; (9)$$

where $\Phi = 1.4826$, and it is the correcting term that makes the median equal to the standard deviation of Gauss distribution. Then the weight $_i$ for each feature correspondence is:

$$\omega_i = \begin{cases} 1 & d_i \le c \cdot \sigma \\ 0 & d_i > c \cdot \sigma \end{cases}, \qquad (10)$$

where $d_i = \sqrt{\left(x_i'(\mathbf{X},\mathbf{P}) - x_i\right)^2 + \left(y_i'(\mathbf{X},\mathbf{P}) - y_i\right)^2}$ is the distance between the projected and tracked features in the input view. $c \in [2.5, 3.5]$ is a scalar that represents the strictness of judgment on outliers.

4. Experimental Results

We tested our method in two sequences of head images for different persons. On the top row of Fig.5, some selected frames of the sequences are shown respectively. Rough head pose estimations are in the middle row, and the refined results in the bottom row. □is the horizontal rotation angle and □is the angle between the face normal and the horizontal plane. Obviously the refined pose visually is quite close to what we see in the images.

In order to examine the accuracy of our method, we used several individual views of a bust that is rotated by known angles on a turntable. As illustrated in Fig.6,□and □ below the input images are the known angles. The maximum error ratio of the refined estimations of □to the total rotation angle 50°is 2.44%.

Top of Fig.7 shows the curves of tracking errors of feature points $\mathbf{p}_1$, $\mathbf{p}_2$, $\mathbf{p}_4$ and $\mathbf{p}_5$ in sequence 2. The thick curves represent results using modified KLT method and the thin curves using the KLT method. As expected, the modified KLT technique can produce more accurate results. Bottom is the iteration number of convergence of $\mathbf{p}_4$ in sequence 2. It can be seen that in most frames the modified KLT method can reduce the numbers of iterative computation. Similar results appear with other feature points. Experimental results show that our method can accurately estimate the head poses that ranged from -50°to 50°.

5. Conclusions

We have described an approach for estimating head poses for an image sequence by using affine correspondence and a 3D head model. By adding an extra rotation, our method can recover six pose parameters. By using 3D head model, a virtual fronto-parallel projection is also constructed to avoid the acquisition of a real fronto-parallel head image as the reference image. A non-linear iterative process of optimization and a modified KLT 2D feature tracking technique are further applied to make the results more accurate.

Currently, our method relies on the 3D head model of the person to be tracked. A generic model can be used in future work. In addition, in order to deal with the perturbation of face expression or non-rigid motion of faces, we will consider dynamic deformation in face surfaces.

Acknowledgement

This work was sponsored in part by the National Science Foundation of China under grant no. 60175025.

References

[1] Q. Chen, H. Wu, T. Fukumoto and M. Yachida, “3D Head Pose Estimation without Feature Tracking”, *Proc. Third IEEE Int. Conf. on Automatic Face and Gesture Recognition,* pp. 88-93,1998

[2] T. Darrel, B. Moghaddam and A.P. Pentland, “Active Face Tracking and Pose Estimation in an Interactive Room”, *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pp.67-72, 1996

[3] S.Y. Ho and H. Huang, “An Analytic Solution for the Pose Determination of Human Faces from a Monocular Image”, *Pattern Recognition Letters,* Vol.19, pp.1045-1054, 1998.

[4] T. Hogg, D. Rees and H. Talhami, “Three-dimensional pose from two-dimensional images: a novel approach using synergetic networks”, *Proc. IEEE Int. Conf. on Neural Networks,* Vol.2, pp.1140-1144, 1995.

[5] R. Lopez and T.S. Huang, “3D Head Pose Computation from 2D Images: Template versus Features”, *Proc. IEEE Int. Conf. on Image Processing,* Vol 2, pp.599 -602, 1995.

[6] S. Ma and Z. Zhang, *Computer Vision*, Science Press, pp. 97-98, 1998.(in Chinese)

[7] H. Murase and S.K. Nayar, “Visual Learning and Recognition of 3D Objects from Appearance”, *Int. Journal of Computer Vision.*, Vol.14,No.1, pp.5-24,1995.

[8] W. H. Press, S. A. Teukolsky, W. T. Vetterling and B. P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing*, Cambridge University Press, Cambridge, 1998.

[9] J. Shi and C. Tomasi, “Good Features to Track”, *Proc. IEEE. Int. Conf. on Computer Vision and Pattern Recognition*, pp. 593 –600, 1994.

[10] I. Shimizu, Z. Zhang, S. Akamatsu and K. Deguchi, “Head Pose Determination from One Image Using a Generic Model”, *Proc. IEEE Int. Conf. on Automatic Face and Gesture Recognition,* pp.100-105, 1998.

[11] Y. Wei, L. Fradet and T. Tan, “Head pose estimation using Gabor eigenspace modeling”, *Proc. Int. Conf. on Image Processing*, Vol. 1, pp. 22-25, 2002.

[12] J. Xiao, T. Kanade, and J.F. Cohn “Robust Full-Motion Recovery of Head by Dynamic Templates and Re-registration Techniques”, *Proc. IEEE. Intl. Conf. on Automatic Face and Gesture Recognition*, pp. 593-600,2002.

[13] R. Yang and Z. Zhang, “Model-based Head Pose Tracking with Stereo Vision”, *Proc. IEEE Int. Conf. on Automatic Face and Gesture Recognition,* pp. 255-260, 2002.

[14] P. Yao, G. Evans and A. Calway, “Using Affine Correspondence to Estimate 3-D Facial Pose”, *Proc. IEEE Int. Conf. on Image Processing*, Vol. 3, pp. 919-922, 2001.

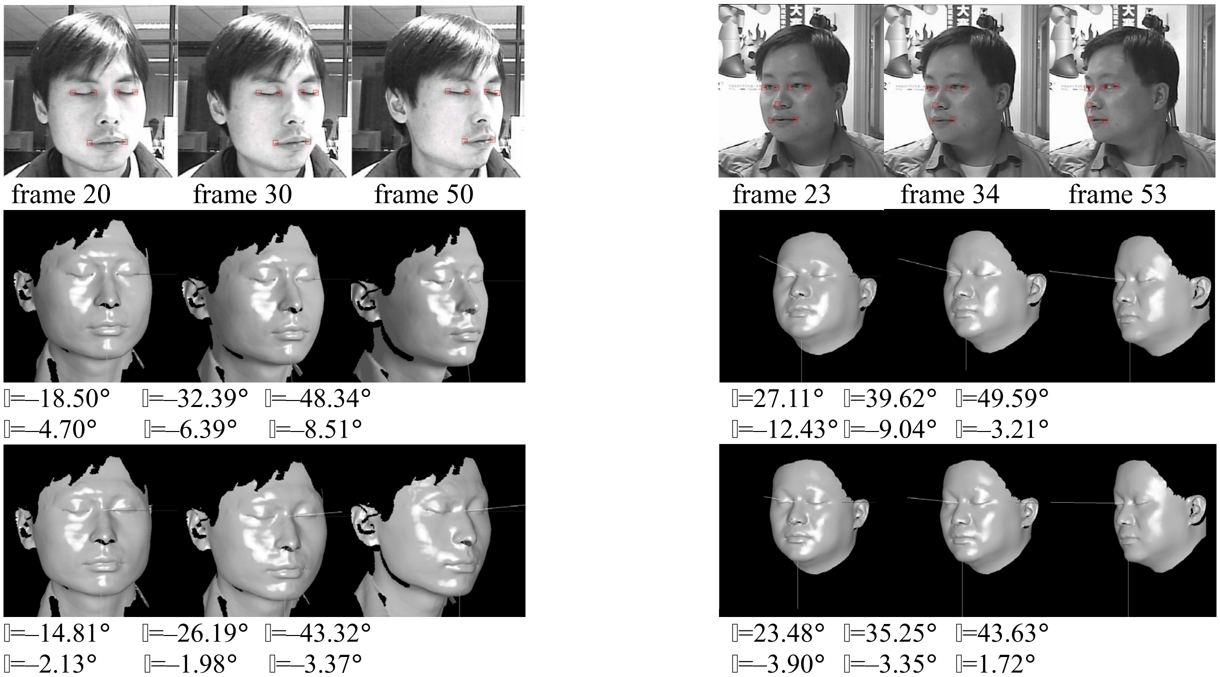

Fig. 5. Experiments on two real image sequences.Top row is some input frames, middle row is the rough estimation and bottom row is the refined results.

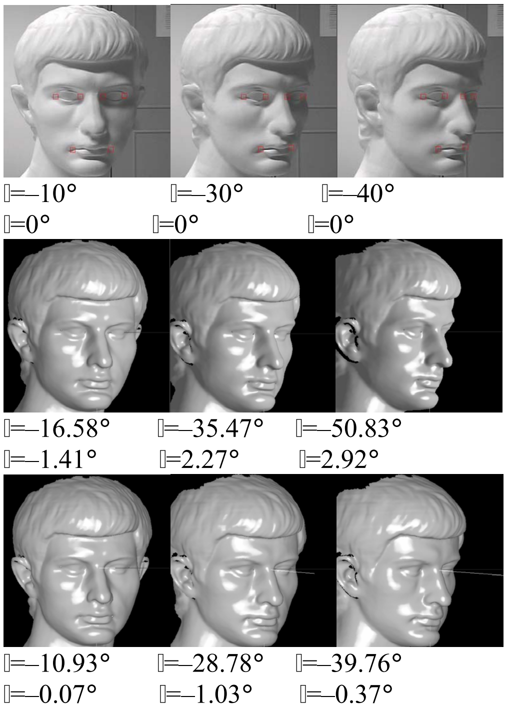

Fig. 6. Results of a bust with measured rotation angles.

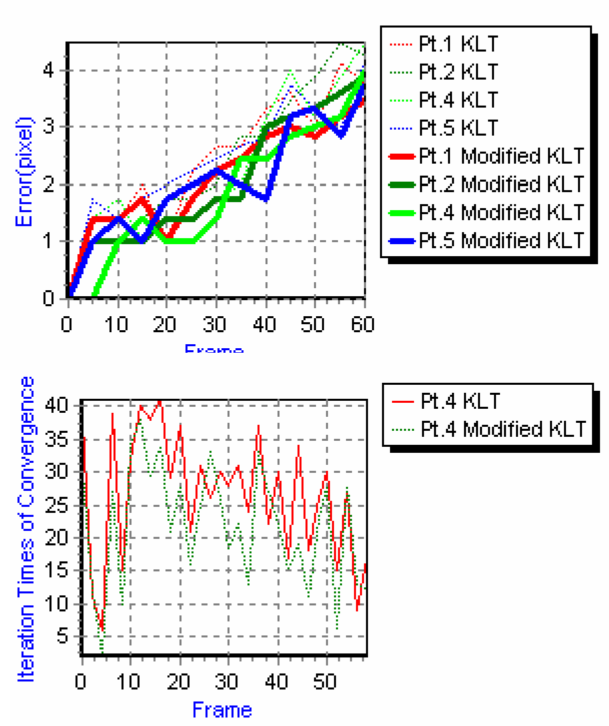

Fig. 7. Comparison of Modified KLT and KLT tracking.
Top is the tracking errors of feature points p_1,p_2,p_4,p_5.
Bottom is the iteration number of convergence of point p_4.