Extracting 3D Facial Animation Parameters from Multiview Video Clips

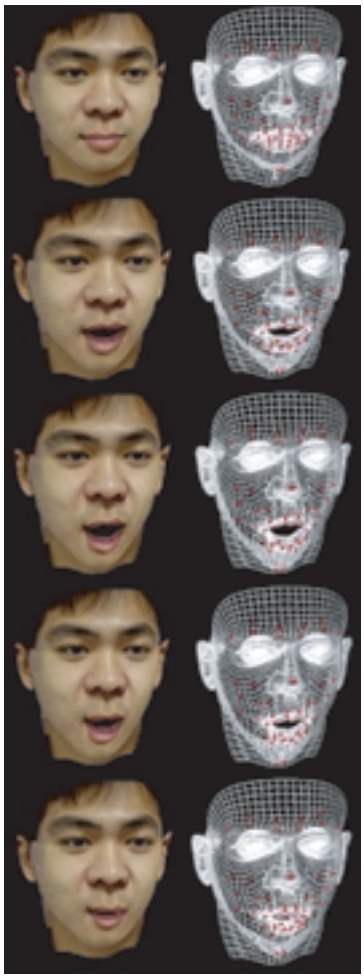
I-Chen Lin, Jeng-Sheng Yeh, and Ming Ouhyoung
National Taiwan University

Our procedure estimates 3D facial and lip motion trajectories from mirror-reflected multiview video clips. Computer simulations reveal that using mirrors yields more accurate 3D position estimation than general-purpose stereovision approaches.

To realistically mimic facial animation, a synthetic face's behaviors must precisely conform to those of a real one. However, facial surface points, being nonlinear and without rigid body properties, have quite complex action relations. During speaking and pronunciation, facial motion trajectories between articulations, called *coarticulation effects*, also prove nonlinear and depend on preceding and succeeding articulations.

Performance-driven facial animation provides a direct and convincing approach to handling delicate human facial variations. This method animates a synthetic face using motion data captured from a performer. In modern computer graphics-based movies such as *Final Fantasy*, *Shrek*, and *Toy Story*, character motion designers used optical or magnetic motion trackers to capture markers' 3D motion trajectories on a performer's face. They can track only a limited number of markers without interference, however, and the dozen or so markers they can place on facial feature points only sparsely cover the whole face area. Therefore, to derive a vivid facial animation, animators must adjust for the uncovered areas. Other approaches, discussed in the "Related Work" sidebar, also present limitations in analyzing and synthesizing facial motion.

To tackle this problem, we propose an accurate and inexpensive procedure that estimates 3D facial motion parameters from mirror-reflected multiview video clips. We place two planar mirrors near a subject's cheeks and use a single camera to simultaneously capture markers' front and side view images. We also propose a novel closed-form linear algorithm to reconstruct 3D positions from real versus mirrored point correspondences in an uncalibrated environment. Figure 1 shows such a reconstruction.



1 The 3D facial motion trajectories estimated with the proposed algorithm for realistic facial animation. The red points in the right column represent the estimated markers' 3D positions, and the left side depicts synthesized facial animation of the pronunciation of "o-u."

0272-1716/02/$17.00 © 2002 IEEE

Related Work

Many methods proposed to approximate human facial motion use physical dynamic systems or mathematical formulations. Terzopoulos and Waters[1] proposed a muscle-based face model with three-layer tissues. Cohen and Massaro suggested that the weights of transition between visemes should be overlapping dominance functions with bases of negative exponential functions.[2] Even though these hypotheses try to parameterize complicated facial motion, they encounter critical problems. For example, what are the parameters' values? How much error will occur when adopting certain parameter values? We can only answer these questions by comparing simulations with measured data from a real human face. However, existing measurement devices such as the optoelectronical motion trackers, though highly accurate, are also quite expensive and pose limitations on marker number and placement.

In 3D facial motion tracking, an optoelectronic system uses optoelectronic cameras to track infrared-emitting photodiodes on a subject's face. Such an instrument suffices for research demanding high accuracy, such as facial biomechanics analysis. However, wires must power each diode, which may interfere with a subject's facial motion.

Video-based systems that apply passive markers avoid this problem. For example, the Vicon series (http://www.vicon.com) uses six to 24 specially designed cameras to capture high-reflectivity markers' motion in a specific spectrum. This costly motion capture system is popular in the computer graphics industry for movies or video games. The protruding spherical markers help with shape analysis, but they don't work well for lip surface motion tracking because people sometimes tuck in or otherwise obstruct lip surfaces.

Most stereovision-based motion-tracking approaches derive from epipolar constraints. This approach first uses corresponding points in images of different viewpoints to estimate the essential matrix. It then decomposes the rotation $\boldsymbol{R}$ and translation $\mathbf{t}$ between cameras from the essential matrix. Finally, it estimates each point's 3D position by intersecting projected vectors from the cameras. Huang and Netravali[3] discussed 3D motion and structure estimation from image sequences.

In addition to capturing stereo videos with multiple cameras, Patterson et al.[4] proposed using mirrors to acquire multiple views for facial motion recording. They simplified the 3D reconstruction problem and assumed a plumb camera and vertical mirrors. Basu et al.[5] used a mirrored view to capture lip motion. They regarded the mirrored view as a flipped image of a virtual camera and applied a general-purpose stereovision approach to estimate 3D lip motion. Our algorithm proves simpler yet more accurate because it conveniently uses mirrored objects' symmetrical properties.

Gluckman and Nayar also researched mirrors and configurations for stereo sensors and developed an epipolar-constraint-based calibration approach.[6,7] Guenter et al.[8] and Pighin et al.[9] demonstrate impressive results of research on performance-driven 3D facial animation.

References

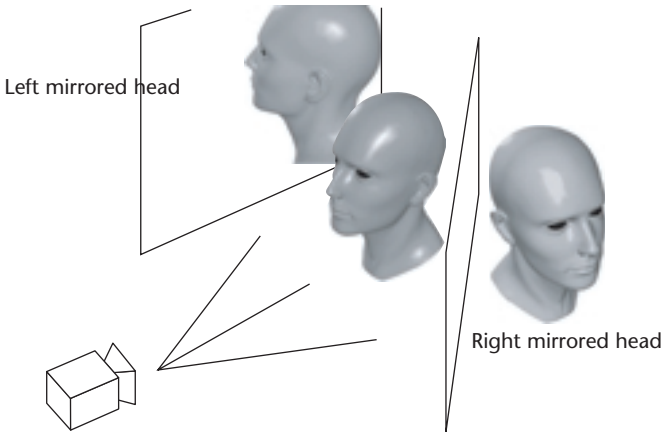
1. D. Terzopoulos and K. Waters, "Physically Based Facial Modeling, Analysis and Animation," *J. Visualization and Computer Animation*, vol. 1, no. 4, Mar. 1990, pp. 73-80.
2. M.M. Cohen and D.W. Massaro, "Modeling Co-articulation in Synthetic Visual Speech," *Models and Techniques in Computer Animation*, Springer-Verlag, Heidelberg, 1993, pp. 139-156.
3. T.S. Huang and A.N. Netravali, "Motion and Structure from Feature Correspondences: A Review," *Proc. IEEE*, vol. 82, no. 2, Feb. 1994, pp. 252-268.
4. E.C. Patterson, P.C. Litwinowicz, and N. Greene, "Facial Animation by Spatial Mapping," *Proc. Computer Animation 91*, N.M. Thalmann and D. Thalmann, eds., Springer-Verlag, Heidelberg, 1991, pp. 31-44.
5. S. Basu and A. Pentland, "A Three-Dimensional Model of Human Lip Motions Trained from Video," *Proc. IEEE Non-Rigid and Articulated Motion Workshop at CVPR 97*, IEEE Press, Piscataway, N.J., 1997, pp. 46-53.
6. J. Gluckman and S.K. Nayar, "Planar Catadioptric Stereo: Geometry and Calibration," *Proc. IEEE Conf. Computer Vision and Pattern Recognition* (CVPR 99), vol. 1, IEEE Press, Piscataway, N.J., 1999.
7. J. Gluckman and S.K. Nayar, "Rectified Catadioptric Stereo Sensors," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 2, 2002, pp. 224-236.
8. B. Guenter et al., "Making Face," *Proc. Computer Graphics* (SIGGRAPH 98), ACM Press, New York, 1998, pp. 55-66.
9. F. Pighin et al., "Synthesizing Realistic Facial Expressions from Photographs," *Proc. Computer Graphics* (SIGGRAPH 98), ACM Press, New York, 1998, pp. 75-84.

Our computer simulations reveal that exploiting mirrors' various reflective properties yields a more robust, more accurate, and simpler 3D position estimation approach than general-purpose stereovision methods that use a linear approach or maximum-likelihood optimization. Our experiments showed a root mean square (RMS) error of less than 2 mm in 3D space with only 20-point correspondences. For semiautomatic 3D motion tracking, we use an adaptive Kalman predictor and filter to improve stability and infer the occluded markers' position. Our approach tracks more than 50 markers on a subject's face and lips from 30-frame-per-second video clips. We've applied the facial motion parameters estimated from the proposed method to our facial animation system.

3D motion tracking

Our face synthesis system separates a face into 11 regions. Assuming each region is a smoothly deformable surface, we find 50 points on a face—10 for lip contours, 12 for lip surfaces, 10 for the mouth, 8 for cheeks, and 10 for the forehead—where the variations most closely represent controlling surface deformation.

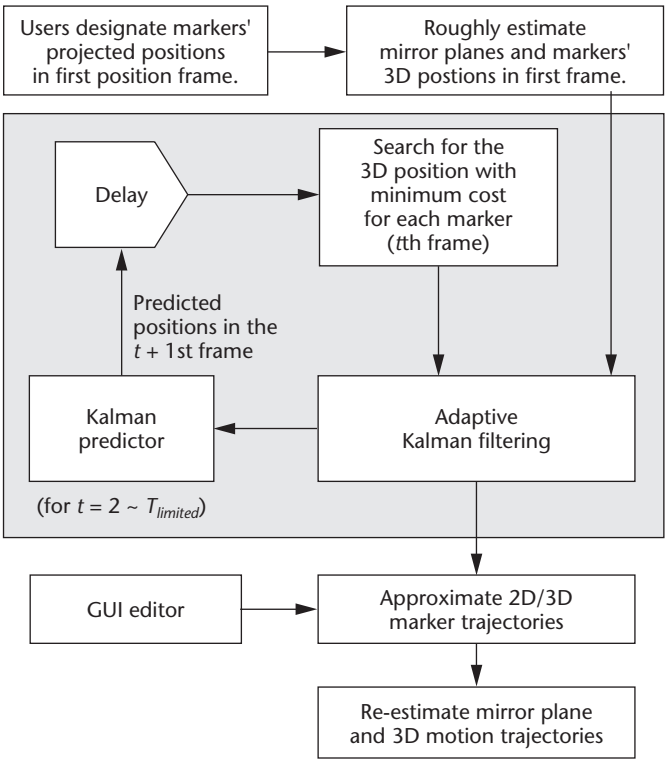
To use these 50 positions as feature points to drive 3D facial animation, we adhere colorful dot markers to them. The markers make tracking feature point movement much easier and more accurate. We use thin markers without protrusion to avoid interfering with natural lip motion. Figure 2 (next page) shows a conceptual diagram of our tracking equipment. We place two planar mirrors next to a subject's face and use only



(a)

(b)

2 (a) A concept diagram of the left and right mirrored heads. (b) The image captured by a digital video camera in 720 × 480-pixel resolution.



3 Semiautomatic 3D-motion marker tracking.

one digital video (DV) camera to capture perfectly synchronized images—one frontal view and two mirrored. We measured the camera's focal length and lens center in advance, during camera calibration.

The mirrors can be placed arbitrarily as long as they include the left and right mirrored face images. We can later estimate their positions and orientations by the proposed algorithm.

We use a semiautomatic approach to track the markers' 3D motion trajectories and apply an adaptive Kalman filter to reduce measurement errors. The filter measures 3D position data and bases predictions on 3D positions and velocities.

Figure 3 shows a flow diagram of the markers' 3D motion tracking.

The proposed solution

Once the method assigns or estimates real and mirrored markers' projected positions, we can calculate markers' 3D positions by first evaluating the mirror's orientation and location relative to the camera and then estimating markers' 3D positions as a minimization problem.

In the first step, we assume flat mirrors and use only the image data within the mirrors' range. We can represent a mirror's location and orientation using a plane equation:

$$ax + by + cz = d \qquad (1)$$

$u = (a, b, c)^t$, $||\mathbf{u}|| = 1$, where u is the plane's unit normal and vector $\mathbf{u}$ has two possible directions. Without loss of generality, we take the direction of $c < 0$. In the following discussion, we assume that I is the camera film's image plane and f is the focal length. We assume camera lens center O to be the origin in the coordinate system, and the camera's line of vision is the positive z axis.

In Figure 4, m_i is the actual 3D position of marker i, $m_i = (x_{mi}, y_{mi}, z_{mi})^t$, and m_i' is the virtual 3D position of marker i in the mirrored space,

$$m_i' = \left(x_{mi}', y_{mi}', z_{mi}' \right)^t$$

P_i is the projection of m_i on I,

$$p_i = \left(f\frac{x_{mi}}{z_{mi}}, f\frac{y_{mi}}{z_{mi}}, f \right)^t = \left(x_{pi}, y_{pi}, z_{pi} \right)^t$$

p_i' is the projection of m_i' on I,

$$p_i' = \left(f\frac{x'_{mi}}{z'_{mi}}, f\frac{y'_{mi}}{z'_{mi}}, f \right)^t = \left(x'_{pi}, y'_{pi}, z'_{pi} \right)$$

(x_{pi}, y_{pi}) and (x'_{pi}, y'_{pi}) are the estimated 2D marker positions.

Mirror properties dictate that

$$m_i' = m_i + ku \tag{2}$$

where k is a scale value. Vectors $\mathbf{m_i}$, $\mathbf{m_i'}$, $\mathbf{u}$ are coplanar, and thus

$$\mathbf{m_i'} \cdot \left(\mathbf{u} \times \mathbf{m_i} \right) = 0 \tag{3}$$

"$\cdot$" is the dot product and $\times$ is the cross product.

From Equation 3, we reformulate in terms of p_i, p_i',

$$\frac{z'_{mi}}{f} p_i' \cdot \left[u \times \left(\frac{z_{mi}}{f} p_i \right) \right] = 0 \tag{4}$$

and simplify it as

$$p_i' U p_i = 0, \text{ where } U = \begin{bmatrix} 0 & -c & b \\ c & 0 & -a \\ -b & a & 0 \end{bmatrix} \tag{5}$$

We can then represent Equation 5 in terms of $\mathbf{u}$ as

$$\left[\left(y_{pi} - y'_{pi} \right) f \;\; \left(-x_{pi} + x'_{pi} \right) f \;\; \left(x_{pi} y'_{pi} - y_{pi} x'_{pi} \right) \right] \begin{bmatrix} a \\ b \\ c \end{bmatrix} = 0 \tag{6}$$

By collecting Equation 6 for each marker correspondence, we can form a matrix $\boldsymbol{M}$, $Mu = 0$, where

$$\boldsymbol{M} = \begin{bmatrix} \left(y_{p1} - y'_{p1} \right) f & \left(-x_{p1} + x'_{p1} \right) f & \left(x_{p1} y'_{p1} - y_{p1} x'_{p1} \right) \\ \left(y_{p2} - y'_{p2} \right) f & \left(-x_{p2} + x'_{p2} \right) f & \left(x_{p2} y'_{p2} - y_{p2} x'_{p2} \right) \\ \vdots & \vdots & \vdots \\ \left(y_{pn} - y'_{pn} \right) f & \left(-x_{pn} + x'_{pn} \right) f & \left(x_{pn} y'_{pn} - y_{pn} x'_{pn} \right) \end{bmatrix} \tag{7}$$

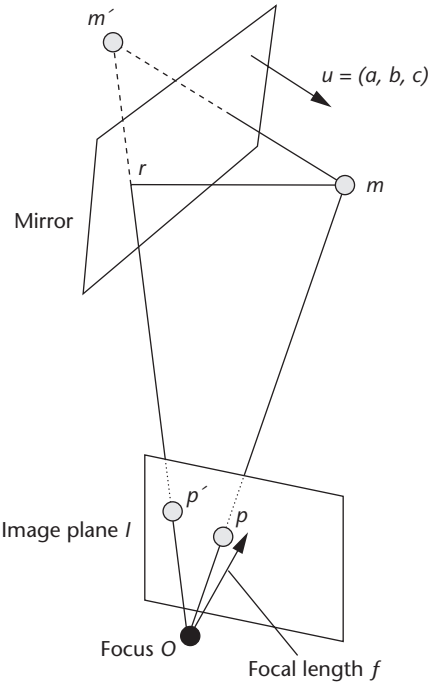
The mirror might not be perfectly flat, however, and we should also allow for noise in marker shape and position on image plane I. We therefore apply the least square method to estimate the vector $\mathbf{u}$ with the least error. It's well known that the solution of

$$\min_u \left\| Mu \right\| \text{ for } \left\| u \right\| = 1 \tag{8}$$

is the eigenvector corresponding to the smallest eigenvalue of the matrix $\boldsymbol{M^t M}$.

Another mirror property is symmetry:

$$\left(m_i' - \Theta \right) = H_u \left(m_i - \Theta \right) \tag{9}$$



4 The geometric representation of the physical point *m*, the reflected virtual point *m*′, and the projection points *p*, *p*′.

where Θ is an arbitrary point on the mirror plane Mirror; $\boldsymbol{H_u} = (\boldsymbol{I_{3\times3}} - 2uu^t)$ is the Householder matrix, and $\boldsymbol{I_{3\times3}}$ is the identity matrix. We choose $\Theta = (0,0,d/c)^t$ and deduce the equation

$$\begin{bmatrix} \left(\frac{2a^2-1}{2f} \right) x_{pi} + \left(\frac{ab}{f} \right) y_{pi} + ac & \frac{x'_{pi}}{2f} \\ \left(\frac{ab}{f} \right) x_{pi} + \left(\frac{2b^2-1}{2f} \right) y_{pi} + bc & \frac{y'_{pi}}{2f} \\ \left(\frac{ac}{f} \right) x_{pi} + \left(\frac{bc}{f} \right) y_{pi} + \frac{2c^2-1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} z_{mi} \\ z'_{mi} \end{bmatrix} = d \begin{bmatrix} a \\ b \\ c \end{bmatrix} \tag{10}$$

From Equation 10, we see that once we've determined vector $\mathbf{u}$, z_{mi} and z'_{mi} are proportional to variable d. We can determine the value d by comparing the scaled data with a reference ruler in the real world.

These steps, then, first estimate unit vector $\mathbf{u}$ by Equation 8, then reconstruct the position of $[x_{mi}, y_{mi}, z_{mi}]^t$ for each marker and stationary point from the depth information solved by the least square method in the form

$$\min_z \left\| G\mathbf{z} - du \right\| \tag{11}$$

based on singular value decomposition (SVD) or QR factorization.[4]

Moreover, to reduce the influence of marker position estimation errors in the front view image, we simply mirror the virtual marker m_i' back to the actual world, set as m_i'',

$$m_i'' = H_u^{-1} \left(m_i' - \Theta \right) + \Theta \tag{12}$$

Mirror Configuration

Fully configuring one mirror entails the following steps.

1. Initialize parameters. A user must manually designate $p_i(1)$, the projected position of actual marker i, and $p'_i(1)$, the projected position of mirrored marker i, in the first video clip ($t = 1$), for $i = 1 \ldots N$. N is the amount of markers the mirror covers.
2. Estimate rough mirror positions and orientations relative to the camera from physical-mirrored point correspondences assigned in the first frame. Estimate $m_i(0)$, the actual 3D position of marker i, for $i = 1 \ldots N$.
3. Predict the 3D position at $t + 1$ as $m_i(t + 1|t)$ and generate mirrored position $m'_i(t+1|t)$ for $i = 1 \ldots N$. Update the time stamp, set $t = t + 1$.
4. Project the actual and mirrored markers back to the camera's image plane I as $p_i(t|t - 1)$, $p'_i(t|t - 1)$. Within the searching area centered by $p_i(t|t - 1)$, find the best r (for example, $r = 6$) 2D projected candidates $pc_{ij}(t|t - 1)$ with minimum ColorCost, which is L2-norm of color differences in block matching compared to that of $p(t - 1)$ and $p(1)$. Repeat this process to find r candidates $pc'_{ik}(t|t - 1)$ of the mirrored part.
5. For each j and k combination, generate 3D candidates $mc_{ijk}(t|t - 1)$ from projected point correspondence of $pc_{ij}(t|t - 1)$, $pc'_{ik}(t|t - 1)$ and calculate the cost function

$$\text{Cost}_{jk} = \alpha\text{DistCost}_{jk} + \beta(\text{ColorCost}_j + \text{ColorCost}_k),$$
$$\text{DistCost}_{jk} = f(\| m_i(t|t - 1) - mc_{ijk}(t|t - 1) \|)$$

   where α and β are user-defined constant values and f is a user-defined monotonically increasing function.
6. Find the best candidate with the minimum Cost_{jk}, and adjust the measurement error variances according to $\text{ColorCost}_j + \text{ColorCost}_k$. Set the best candidate to the measured 3D position and filter it as $m_i(t)$.
7. If $t < T_{\text{limit}}$, return to step 3 or refer to the user manual for fine-tuning.
8. Calculate U_{fine}, the mirror's fine positions and orientations, from user-tuned projected point correspondences. Reestimate accurate 3D markers' motion trajectories by U_{fine} and tuned projected point correspondences.

The process for another mirror is similar. Adjusting measurement-error variances in step 6 accords with image similarity. When a marker image is occluded or interfered with by interlace effect or intense specular-lighting noise, the cost function value will be dramatically high, and the measurement error variances will be large. This decreases the Kalman gain. In other words, the impact weights of contaminated measurement data are diminished and the effects of noise or occlusion can be alleviated. Details on Kalman filter use and adaptation are available elsewhere.[1,2]

References

1. Y. Altunbasak, A.M. Tekalp, and G. Bozdagi, "Simultaneous Stereo-motion Fusion and 3D Motion Tracking," *Proc. Int'l Conf. Acoustics, Speech, and Signal Processing* (ICASSP 95), vol. 4, IEEE Press, Piscataway, N.J., 1995, pp. 2277-2280.
2. S.M. Bozic, *Digital and Kalman Filter*, Edward Arnold Ltd., London, 1979.

and take $m''' = (m_i + m''_i)/2$ as the 3D position of marker i.

To more accurately estimate m''', we can also apply nonlinear maximum likelihood optimization that minimizes the location variation on an image plane to improve the estimated mirror normal u. However, a mirror plane's useful properties mean the vector $\mathbf{u}$ estimated by a linear algorithm is sufficiently accurate. In our simulation, maximum likelihood optimization improved less than 2 percent of the root mean square (RMS) 3D position error under quite noisy circumstances.

The previous steps estimated markers' 3D positions. However, because test subjects may swing or nod their heads when speaking and making facial expressions, both facial and head motions cause 3D marker movement. To capture precise facial motion, we must estimate the head motion and remove it from 3D facial expression data.

We fix four additional markers on the performer's ears and regard them as points on a rigid head. We then apply a rigid-body motion estimation algorithm[1] to determine the head motion.

The sidebar "Mirror Configuration" describes the full procedure for one mirror.

3D position estimation concepts

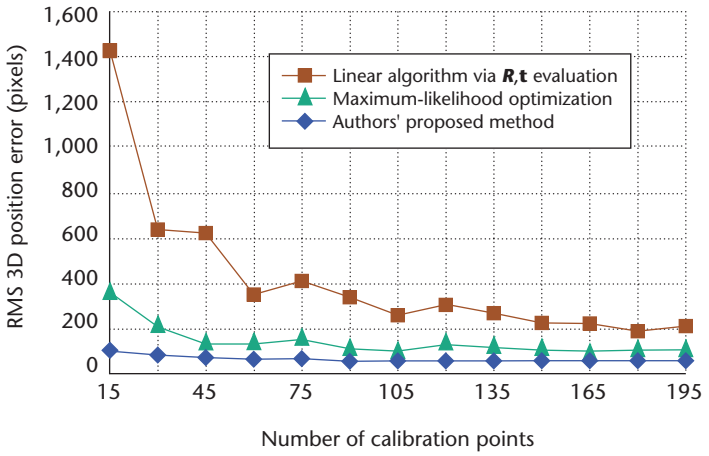
Intuitively, estimating 3D position from mirror-reflected multiview images should prove more robust than methods that estimate 3D position by calculating rotation matrix $\boldsymbol{R}$ and translation vector $\mathbf{t}$ between two cameras. Each of these values has three degrees of freedom. In our case, we evaluate the mirror plane normal u and scale d, which has only four DOF. Fewer DOF mean we can use much less information to reach accuracy of the same magnitude.

Also, when estimating $\boldsymbol{R}$ and $\mathbf{t}$, we must first evaluate the essential matrix, which has eight DOF, then estimate an analogous rotation matrix $\boldsymbol{W}$. However, because $\boldsymbol{W}$ usually doesn't have a rotation matrix's properties, such as orthogonality, we must then further adjust $\boldsymbol{W}$ to fit the properties. We can then evaluate the vector $\mathbf{t}$. Each step involves many numerical matrix computations, and errors accumulate with each step. Therefore, the two-view linear algorithm yields distorted $\boldsymbol{R}$ and $\mathbf{t}$ estimations, necessitating successive nonlinear optimizations such as maximum-likelihood evaluations. Weng et al. discuss error analysis and 3D position estimation and structure reconstruction from stereovision approaches.[2]

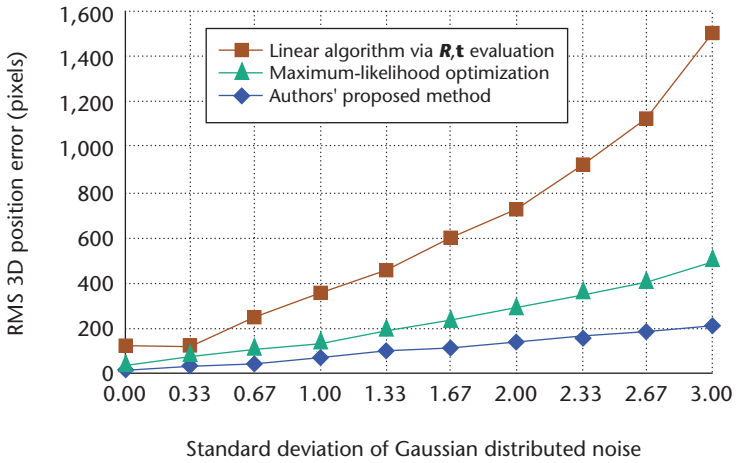
Error estimation

To compare our approach's accuracy and robustness with general-purpose stereovision approaches, we conducted computer simulation experiments using three subject algorithms:

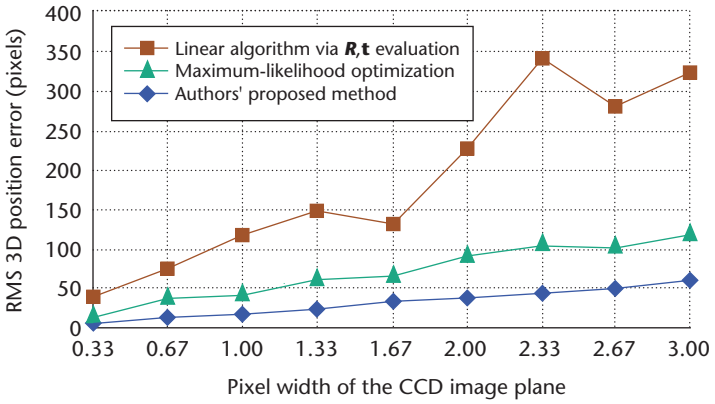
1. Our proposed linear algorithm reconstructs 3D positions via mirror-plane normal U evaluation.
2. A linear virtual camera approach estimates 3D positions by evaluating rotation R and translation $\mathbf{t}$ between cameras.
3. A maximum-likelihood optimization improves the linear virtual camera approach's results.



5 Error estimation of three approaches with different numbers of point correspondences. The *y* axis is the scale of RMS 3D position error (in pixels). We applied Gaussian distributed noise (mean = 0, standard deviation = 1 pixel in both *x* and *y* axes) to disturb the projection on the image plane before digitizing the charge-coupled device (CCD) array.



6 Error estimation of three approaches under Gaussian distributed noise. We applied Gaussian distributed noise (mean = 0) of different standard deviations (in both *x* and *y* axes) to disturb the projection on the image plane before digitizing the CCD array.



7 Error estimation of three approaches under conditions of different pixel widths in both *x* and *y* axes. The simulated camera has a focal length of 1,500 pixels and has a fixed image plane size but variable pixel widths. At pixel width 1.0, the CCD array is 720 × 720 pixels; at pixel width 0.333, the CCD array is about 2,160 × 2,160 pixels.

We adapted methods 2 and 3[2] by flipping projected mirrored-reflected images to form a virtual camera view. Figures 5, 6, and 7 show the results of our simulations, all performed using Mathwork's MatLab numerical computation software.

For testing, we used sets of randomly generated 3D points within a 9,000 × 18,000 × 9,000 pixel cube and 40,000 pixels away from the lens center. For the second and third tests, we applied 60 randomly generated 3D points as a 3D object. The simulated camera has a 720 × 720 pixel charge-coupled device (CCD) array and a 1,500-pixel focal length. Assuming the object is 2 meters away, one pixel length equals 0.05 mm.

For the first test, we applied normally distributed noise with constant standard deviation to simulate the sum of various noise types and then truncated the contaminated projection point data to fit pixel grids on a simulated CCD array. The noise was random and had a mean of zero. We can therefore better diminish the effects of disturbance in an overdetermined system.

Because the unknown parameters in the U evaluation have fewer DOF than the $\boldsymbol{R}$, $\mathbf{t}$ ones, our method can reach the same accuracy with fewer point correspondences than the general-purpose stereovision one.

In the second test, we fixed the number of point correspondences at 60, and the standard deviation of noise varied from 0 to 3 in both x and y axes. As in the previous test, the fewer DOF in unknown parameters made our method more robust that the other two under noisy conditions. The third test demonstrated that our method reaches the same accuracy with lower resolution than linear or nonlinear virtual camera approaches.

We also developed an experiment to evaluate accuracy. We attached 20 markers, each 3 mm in diameter, to the right side of a plastic dummy's face, which was 2 meters away from the camera, and placed a planar mirror next to the right cheek. To mimic reality, the front and side views of the face's right side only occupied the full image's left half. Because a 3D laser scanner has a measurement error range of less than 0.2 mm, we assumed that it provided exact data. Comparing positions estimated using our method with the 3D scanned data, we found our method's RMS 3D position error to be 1.95 mm. The maximal error of 2.94 mm occurs at a marker position beneath the lower lip.

Advantages

Compared to the commonly used stereovision approach that adopts two-view images, estimating 3D positions and motions via mirror plane evaluation from mirror-reflected multiview images has many advantages.

Simplicity and computational efficiency. In our algorithm, evaluating the mirror plane normal U requires solving only one equation by the linear least square evaluation, as shown in Equation 8, where the corresponding matrix $\boldsymbol{M}$ is $n \times 3$. With the general-purpose two-view algorithm, however, estimating rotation matrix $\boldsymbol{R}$ and unit translation vector $\mathbf{T}^0$ requires processing three linear least square evaluations, and their associated matrices are $\boldsymbol{n} \times 9$, 3×3, and 3×3. Furthermore, to obtain reasonable results, maximum-likelihood evaluation must be used. Because this optimization process is a kind of nonlinear iterative improvement, more computation results than with the linear approach. For depth evaluation, both the proposed method and the two-view approach require another least square evaluation for each point correspondence.

Accuracy and robustness. Our method has four unknown parameters rather than the six of general-purpose two-view approaches. We demand less information, such as fewer point correspondences to reach the same accuracy as with stereovision. Our method also has a larger error tolerance.

Perfect synchronization and low cost. Multiple-camera approaches face the critical problem of camera synchronization. In facial motion capture, the tip of a subject's lower lip moves down more than 1 cm within 30 ms when pronouncing "pa," for example. When using only video-based synchronization, imperfect synchronization can make the expected value of measurement error of the lip's tip more than 0.5 cm. Therefore, accurate data capture by multiple cameras demands special synchronization devices. In our approach, one camera and two mirrors can simultaneously capture three images of different viewpoints. Perfect synchronization among multiple views is inherent in our system.

Disadvantages

Our method has two main disadvantages.

Restricted measurement range. Because our method uses a single camera to capture three different views simultaneously, measured targets' motion range must be within the volume of space between two mirrors. The mirrors' orientation and size therefore limit the method's applications.

Limited image area for each view. Because our method includes three view images in a snapshot, each view can take up just one-third of the total image area.

However, our third computer simulation (Figure 7) demonstrates that our method offers similar or even better accuracy than the maximum-likelihood optimized two-view approach: it provides identical point correspondence but four times the image resolution (two times both pixel width and height).

Applications

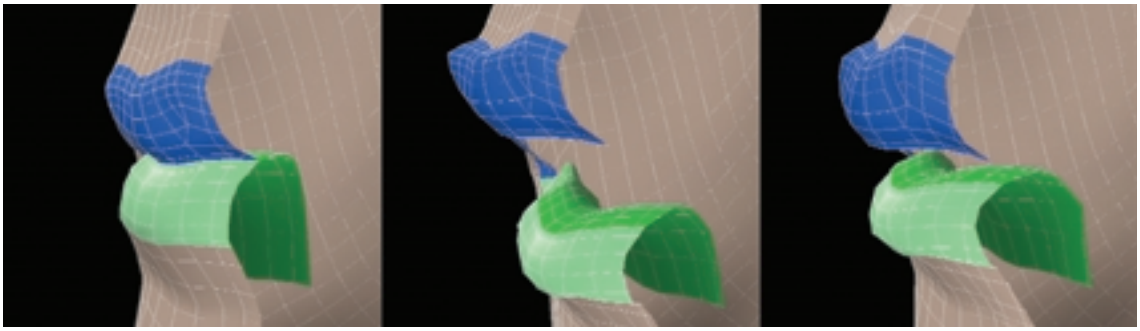
For motion tracking without limited action space, stereovision approaches employing multiple cameras remain irreplaceable. Synchronization hardware and camera calibration with many point correspondences can probably overcome the disadvantages of difficult synchronization and higher noise sensitivity when using multiple cameras.

Nevertheless, our method provides a good and inexpensive solution in applications where motion ranges are restricted, such as 3D facial animation parameters, or finger or 3D hand gesture tracking. Because our method uses only a single camera and mirrors can reach high accuracy with few point correspondences, it doesn't require heavy calibration. This makes the proposed algorithm adequate also for applications requiring fast or even real-time dynamic calibration.

Realistic facial animation

We applied the estimated facial motion parameters to our facial animation system, which can synthesize realistic facial expressions at more than 30 frames per second on a Pentium III 1-GHz PC with an Nvidia GeForce 2MX OpenGL acceleration card. For head modeling, we applied a generic model to fit depth range images acquired by a 3D laser scanner. We adopted the radial basis[3] scatter data interpolation deformation method, a smooth interpolation function that can distribute the effects of feature points to nonfeature points.

We separated our generic face into 11 regions. Control points in a region can only affect vertices within that region. We also applied interpolation to smooth the jitter effects at region boundaries. The control points consist of motion-captured feature points and *supplementary hypothetical points*, that is, points diffi-



8 A cross-sectional view of the lips: the lower outer lip (light green) and supplementary inner lip model (dark green).

9 Synthetic subtle facial expressions of pouting and mouth twisting. Using two mirrors permits capturing asymmetric facial motions.

cult to capture (such as the jaw near the ear) due to video viewpoint limitations. We use hypotheses to drive these points according to related feature points.

The inner lips represent another important and difficult-to-track facial region. A lip's inner surface, hidden behind the outer part in a neutral face, partially appears when the mouth is open. The lower inner lip is especially important when a mouth is puckering or rounding, as it does when pronouncing "u" or "o." At that time, almost half of the lower inner lip protrudes, forming the lower lip's inner contour. We therefore used a supplementary inner lip model, shown in Figure 8. Light green represents the lower outer lip, driven by six feature points in motion captured data, and dark green depicts the supplementary inner lip model, a modified Hermite surface controlled by outer lip and jaw surface tangent vectors.

When we deform the synthetic head according to motion-tracked data frame by frame, we can generate realistic facial animation. Figure 9 shows that subtle asymmetric facial expressions such as twisting the mouth can be synthesized because two mirrors capture the whole face's motion.

We could also use our method to collect 3D facial motion data sets for coarticulation analysis and synthesis of a speech-driven talking head. You'll find a demonstration video at http://www.cmlab.csie.ntu.edu.tw/~ichen/RFAP/RFAP_Intro.htm.

Correct and complete lip modeling is a significant factor in realistic face synthesis because lip silhouettes determine the mouth's inner contours. In our future work, we'll further estimate motion of the inner lip surfaces. We'll also use lighting and high-reflectivity markers in a specific light spectrum, such as infrared rays or ultraviolet, to improve tracking accuracy. ■

Acknowledgments

This work is a part of a collaborative project of Inria, France, and National Taiwan University, Taiwan. We would like to acknowledge Nathalie Parlangeau-Valles, Yves Laprie, Dominique Fohr, and other researchers of the speech group of Loria, France, for helping with our experiments in French. We'd especially like to thank Michel Pitermann, whose face data we captured for one of the models shown here. We'd also like to thank Con-

nie Tao, a visiting student from the Massachusetts Institute of Technology, who helped us with grammatical and style revision.

References

1. K.S. Arun, T.S. Huang, and S.D. Blostein, "Least Square Fitting of Two 3D Point Sets," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 9, no. 5, Sept. 1987, pp. 698-700.
2. J. Weng, T.S. Huang, and N. Ahuja, *Motion and Structure from Image Sequences*, Springer-Verlag, Heidelberg, 1993.
3. G.M. Nielson, "Scattered Data Modeling," *IEEE Computer Graphics and Applications*, vol. 13, no. 1, Jan. 1993, pp. 60-70.
4. G. Golub and C. F. Van Loan, *Matrix Computation* (3rd edition), John Hopkins Univ. Press, Baltimore, 1996.

I-Chen Lin *is a PhD candidate in computer science and information engineering at National Taiwan University. His research interests include facial animation and modeling, motion tracking, and stereo computer vision. He received a BS in computer science from National Taiwan University. He is a student member of ACM, IEEE, and IEEE Computer Society.*

Jeng-Sheng Yeh *is a PhD candidate at National Taiwan University. His research interests include facial animation, modeling, and tracking. He is also interested in nonphotorealistic rendering, Chinese painting, and force-feedback devices. He received a BS in computer science and information engineering from National Taiwan University. He is a student member of Siggraph.*

Ming Ouhyoung *is a professor of computer science and information engineering at National Taiwan University. His research interests include computer graphics, virtual reality, and multimedia systems. He received a BS and MS in electrical engineering from National Taiwan University and a PhD from the University of North Carolina at Chapel Hill. He's a member of IEEE and ACM.*

Readers may contact authors at the Dept. of Computer Science and Information Engineering, National Taiwan University, No. 1, Roosevelt Rd. Sect. 4, Taipei, 106 Taiwan; {ichen, jsyeh}@cmlab.csie.ntu.edu.tw, and ming@csie.ntu.edu.tw, http://www.cmlab.csie.ntu.edu.tw/rp.html.