1998-12-04

Fast, reliable head tracking under varying illumination

BU CS TR98-018 rev. 2. To appear in **Proc. IEEE Conf. on Computer Vision and Pattern Recognition**, June 1999*

Fast, Reliable Head Tracking under Varying Illumination

Marco La Cascia and Stan Sclaroff
Computer Science Department - Boston University
Boston, MA 02215

Abstract

An improved technique for 3D head tracking under varying illumination conditions is proposed. The head is modeled as a texture mapped cylinder. Tracking is formulated as an image registration problem in the cylinder's texture map image. To solve the registration problem in the presence of lighting variation and head motion, the residual error of registration is modeled as a linear combination of texture warping templates and orthogonal illumination templates. Fast and stable on-line tracking is then achieved via regularized, weighted least squares minimization of the registration error. The regularization term tends to limit potential ambiguities that arise in the warping and illumination templates. Tracking does not require a precise initial fit of the model; the system is initialized automatically using a simple 2D face detector. The only assumption is that the target is facing the camera in the first frame of the sequence. Experiments in tracking are reported.

1 Introduction

Three-dimensional head tracking is a crucial task for several applications of computer vision. Problems like face recognition, facial expression analysis, lip reading, *etc.*, are more likely to be solved if a stabilized image is generated through a 3D head tracker. Determining the 3D head position and orientation is also fundamental in the development of vision-driven user interfaces and, more generally, for head gesture recognition. Furthermore, head tracking can lead to the development of very low bitrate model-based video coders for video telephony, and so on. Most potential applications for head tracking require robustness to significant head motion, change in orientation, or scale. Moreover, they must work near video frame rates. Such requirements make the problem even more challenging.

1.1 Previous Work

In recent years several techniques have been proposed for 3D head motion and face tracking. Some of these techniques focus on 2D tracking (*e.g.*, [6, 10, 16, 22, 23]), while others focus on 3D tracking or stabilization.

Some methods for recovering 3D head parameters are based on tracking of salient points, features, or 2D image patches [1, 12]. Others use optic flow to constrain the motion of a rigid or non-rigid 3D surface model [2, 7]. In [14], a render-feedback loop was used to guide tracking for an image coding application. More complex physically-based models for the face that include both skin and muscle dynamics for facial motion were used in [8, 21]. Global head motion can also be tracked using a plane under perspective projection [4]. Recently [13] formulated the head tracking problem in terms of color image registration in the texture map of a 3D cylindrical model. Similarly Schödl, Haro and Essa [18] proposed a technique for 3D head tracking using a full head texture mapped polygonal model.

Most of the above mentioned techniques are not able to track the face in presence of large rotations or changes in lighting conditions, and some require accurate initial fit of the model to the data.

1.2 Approach

The method we propose builds on and extends the work of [13, 18]. The head is modeled as a texture mapped cylinder. Tracking is formulated as an image registration problem in the cylinder's texture map image. Our enhancements enable fast and stable on-line tracking of extended sequences, despite noise and large variations in illumination. In particular, the image registration process is made more robust and less sensitive to changes in lighting through the use of an illumination basis and regularization.

In this respect, this work is related to [10]. The main differences are: 1.) the use of a user-independent illumination basis, and 2.) the use of a regularization term that improves the tracking performance. A similar approach to estimating affine image motions and changes of view is proposed by [3]. This approach employed an interesting analogy with parametrized optical flow estimation; however, their iterative algorithm is unsuitable for real-time operation.

As will become evident in the experiments, our proposed technique can improve the performance of a tracker based on the minimization of sum of squared differences (SSD) in presence of illumination changes. To achieve this goal we solve the registration problem by modeling the residual error in a way similar to the one proposed in [10]. The method employs an orthogonal illumination basis that is precomputed off-line over a training set of face images collected under varying lighting conditions. In contrast to the previous approach of [10], the illumination basis is independent of the person to be tracked. Moreover, we propose the use of a regularizing term in the image registration. This improves the long-term robustness and precision of the SSD tracker considerably.

2 Formulation

The formulation we propose is based on a three-dimensional textured polygonal model whose parameters are estimated through image registration in the texture map [13]. The model we use is a rigid cylinder that is

parametrized by its 3D position and orientation. During initialization, the model is positioned, rotated and scaled to fit the head in the image plane. The reference texture $\mathbf{T}_0$ is then obtained by projecting the initial frame of the sequence $\mathbf{I}_0$ onto the visible part of the cylinder surface. An example mapping of the input frame onto the cylinder in the texture map is shown in Fig. 1.

As a new frame is acquired it is possible to find a set of cylinder parameters such that the texture extracted from the incoming frame best matches the reference texture. In other words, the 3D head parameters are recovered by performing image registration in the model's texture map. Due to the rotations of the heads the visible part of the texture can be shifted with respect to the reference texture. In the registration procedure we consider only the intersection of the two textures.

As a precomputation, a collection of warping templates is computed by taking the difference between the reference texture $\mathbf{T}_0$ and the textures corresponding to warping of the input frame with slightly displaced cylinder parameters. Note that the motion templates used in [3, 10] are computed in the image plane. In our case the templates are computed in the texture map plane. A similar approach has been successfully used in [5, 9, 19].

Once the warping templates have been computed, the tracking can start. Each new input frame $\mathbf{I}$ is warped into the texture map using the current parameter estimate $\mathbf{a}^-$. This yields a texture map $\mathbf{T}$. The residual pattern (difference between the reference texture and the warped image) is modeled as a linear combination of the warping templates $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_K]$ and illumination templates $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_M]$ that model lighting effects.

The optimal set of coefficients is estimated via least squares. These coefficients are linearly related to the increment $\Delta\mathbf{a}$ of the cylinder parameters, so we can compute the new cylinder parameters estimate.

As we warp video into the texture plane, not all pixels have equal confidence. This is due to nonuniform density of pixels as they are mapped between the image and texture map planes. Note also that the texture map is 360^o wide but only a 180^o part of the cylinder is visible at any instant. Clearly, we should associate a zero confidence to the part of the texture corresponding to the back-facing portion of the surface.

Due to possible coupling between the warping templates and/or the illumination templates, the least squares solution may become ill-conditioned. As will be seen, this conditioning problem can be averted through the use of a regularization term. The complete formulation will now be discussed in detail.

2.1 Image warps

Each incoming image must be warped into the texture map. The warping function corresponds to the inverse texture mapping of a cylinder in arbitrary 3D position. In what follows we will denote the warping function:

$$\mathbf{T} = \Gamma(\mathbf{I}, \mathbf{a}) \tag{1}$$

where $\mathbf{T}$ is the texture corresponding to the frame $\mathbf{I}$ warped onto a cylinder with parameters $\mathbf{a}$. The parameter vector $\mathbf{a}$ contains simply the position and orientation of the cylinder. The use of linear combinations of these parameters leads to better conditioning of the problem[18] and is currently under investigation. An example of input frame $\mathbf{I}$ with cylinder model and the corresponding texture map $\mathbf{T}$ are shown in Fig. 1(a,b).

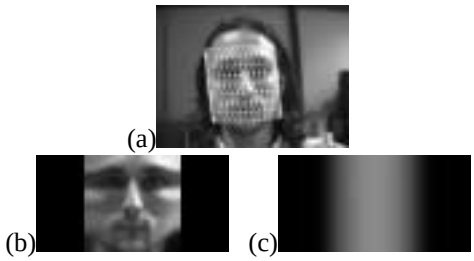


Figure 1: Example: (a) input frame $\mathbf{I}$ with the cylindrical model superimposed, (b) corresponding texture map $\mathbf{T}$, and (c) confidence map $\mathbf{W}$.

Note that as we warp video into the texture plane, not all pixels have equal confidence. In fact parts of the textures corresponding to the non-visible part of the cylinder contribute no pixels and therefore have zero confidence. Moreover due to nonuniform density of pixels as they are mapped between image and texture plane, different visible parts of the textures have different confidence. An approximate measure of the confidence can be derived in terms of the ratio of a triangle's area in the input image over the triangle's area in the texture map[13]. The associated confidence map will be denoted with $\mathbf{W}$. The confidence map for the example is shown in Fig. 1(c).

For notational convenience, all images are represented as long vectors obtained by lexicographic reordering of the corresponding matrices.

2.2 Model initialization

To start any registration based tracker, the model must be fit to the initial frame to compute the reference texture and the warping templates. This initialization can be accomplished automatically using a 2D face detector [17] and assuming that the subject is facing the camera. The approximate 3D position of the cylinder is then computed assuming a unit radius. Note that assuming unit radius is not a limitation as in any case we estimate the relative motion of the head. In other words people with a large head will be tracked as "farther from the camera" and people with a smaller head as closer.

It is important to note using a simple model for the head makes it possible to reliably initialize the system. Simple models require the estimation of fewer parameters in automatic placement schemes, and they are more robust to slight perturbations in parameters. A planar model [4] also offers these advantages; however, we have found that this model is not powerful enough to cope with the self occlusions generated by large rotations. On the other hand, we

also experimented with a complex rigid head model generated averaging the Cyberware scans of several people in known position. Using such a model we were not able to automatically initialize the model, since there are too many degrees of freedom. Furthermore, tracking performance was markedly less robust to perturbations in the model parameters. Even when fitting the detailed 3D model by hand, we were unable to gain improvement in the tracker precision or stability over a simple cylindrical model.

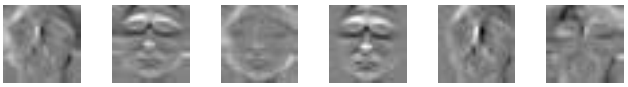
Once we know the initial position of the model we can generate a collection of *warping templates* from the reference texture[9, 13, 19]. Given a parameter displacement matrix $\mathbf{N}_a = [\mathbf{n}_1, \mathbf{n}_2, \ldots, \mathbf{n}_D]$ we compute the warping templates matrix $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_K]$ with columns:

$$\mathbf{b}_k = \mathbf{T}_0 - \Gamma(\mathbf{I}_0, \mathbf{a}_0 + \mathbf{n}_k) \tag{2}$$

where $\mathbf{I}_0$ is the initial frame of the sequence, $\mathbf{n}_k$ is the parameter displacement vector for the k^{th} difference vector (warping template), and $\mathbf{a}_0$ is the initial warping parameter vector (*i.e.*, the initial position and orientation of the cylinder).

In practice, four difference vectors per model parameter are sufficient. For the k^{th} parameter, these four difference images correspond with the difference patterns that result by changing that parameter by $\pm\delta_k$ and $\pm 2\delta_k$. The values of the δ_k can be easily determined such that all the difference images have the same energy as shown in [13]. Note that the need for using $\pm\delta_k$ and $\pm 2\delta_k$ is due to the fact that the warping function is only locally linear in $\mathbf{a}$. Experimental results confirmed this intuition. An analysis of the extension of the region of linearity in a similar problem has been conducted in [5].

Fig. 2 shows a few difference images (warping templates) obtained for a typical initial image.



(a) (b) (c) (d) (e) (f)

Figure 2: Example of warping templates corresponding to translations along the (x, y, z) axes (a,b,c) and Euler rotations (d,e,f). Note the similarity between the templates for horizontal translation (a) and vertical rotation (e). Note also the similarity between vertical translation (b) and horizontal rotation (d). Only that part of the template with nonzero confidence is shown.

2.3 Illumination

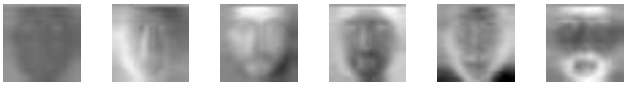
Tracking based on the minimization of the sum of squared differences between the incoming texture and a reference texture is inherently sensitive to changes in illumination. Better results can be achieved by minimizing the difference between the incoming texture and an illumination-adjusted version of the reference texture. If we assume a Lambertian surface in the absence of self-shadowing, then it has been shown that all the images of the same surface under different lighting conditions lie in a three-dimensional linear subspace of the space of all possible images of the object[20]. In this application, none of these conditions is met. Moreover, the nonlinear image warping from image plane to texture plane distorts the linearity of the three-dimensional subspace. Nevertheless, we can still use a linear model as an approximation [10, 11]:

$$\mathbf{T} - \mathbf{T}_0 \approx \mathbf{U}\mathbf{c}. \tag{3}$$

The columns of the matrix $\mathbf{U}$ constitute the *illumination templates*. In [10], these templates are obtained by taking the singular value decomposition (SVD) of a set of training images of the target subject taken under different lighting conditions. An additional training vector of ones is added to the training set to account for global brightness changes. The main problem of this approach is that the illumination templates are subject-dependent.

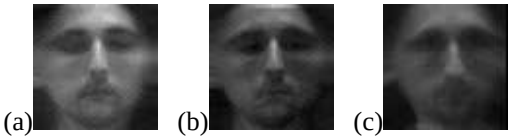
In our system, we generate a user-independent set of illumination templates. This is done by taking the SVD of a large set of textures corresponding to faces of different subjects, taken under varying lighting conditions. The SVD was computed after subtracting the average texture from each sample texture. The training set of faces we used was previously aligned and masked as explained in [15]. In practice, we found that ten illumination templates are sufficient to account for illumination changes.

Note that the illumination basis vectors are low-frequency images. Thus any mis-alignment between the illumination basis and the reference texture is negligible. The first few illumination basis vectors are shown in Fig. 3. Fig. 4 shows a reference image and the same image after the lighting correction (in practice $\mathbf{T}_0$ and $\mathbf{T}_0 + \mathbf{U}\mathbf{c}$).



(a) (b) (c) (d) (e) (f)

Figure 3: The first six illumination templates before masking. Only the part of the texture with nonzero confidence is shown.



(a) (b) (c)

Figure 4: Example of the lighting correction on the reference texture. The reference texture (a). Reference texture after the lighting correction (b) to match incoming texture (c).

2.4 Combined Parametrization

Following the line of [3, 10], we then model the residual images computed by taking the difference between the incoming texture and the reference texture as a linear combination of illumination templates $\{\mathbf{u}_1, \mathbf{u}_2, \ldots \mathbf{u}_M\}$ and warping templates $\{\mathbf{b}_1, \mathbf{b}_2, \ldots \mathbf{b}_L\}$:

$$\mathbf{T} - \mathbf{T}_0 \approx \mathbf{B}\mathbf{q} + \mathbf{U}\mathbf{c} \tag{4}$$

In our experience this is a reasonable approximation for low-energy residual textures. A multiscale approach is used so that the system can handle higher energy residual textures.

2.5 Tracking

To find the warping parameters $\mathbf{a}$, we first find $\mathbf{c}$ and $\mathbf{q}$ by solving the following weighted least squares problem:

$$\mathbf{W}(\mathbf{T} - \mathbf{T}_0) \approx \mathbf{W}(\mathbf{B}\mathbf{q} + \mathbf{U}\mathbf{c}) \tag{5}$$

where $\mathbf{W} = diag[\mathbf{T}_w]$ is the weighting matrix, containing the confidence weights mentioned earlier.

If we define:

$$\mathbf{R} = \mathbf{T} - \mathbf{T}_0; \mathbf{x} = \begin{bmatrix} \mathbf{c} \\ \mathbf{q} \end{bmatrix}; \mathbf{M} = [\mathbf{U}|\mathbf{B}]; \tag{6}$$

we can write the solution as:

$$\begin{aligned} \mathbf{x} &= \arg\min_{\mathbf{x}} \|\mathbf{R} - \mathbf{M}\mathbf{x}\|_W & (7) \\ &= \mathbf{K}\mathbf{R} & (8) \end{aligned}$$

where $\mathbf{K} = [\mathbf{M}^T\mathbf{W}^T\mathbf{W}\mathbf{M}]^{-1}\mathbf{M}^T\mathbf{W}^T\mathbf{W}$ and $\|\mathbf{x}\|_W = \mathbf{x}^T\mathbf{W}^T\mathbf{W}\mathbf{x}$ is a weighted L-2 norm. If we are interested only in the increment of the warping parameter $\Delta\mathbf{a}$ we can compute only the $\mathbf{q}$ part of $\mathbf{x}$. Finally:

$$\mathbf{a} = \mathbf{a}^- + \mathbf{N}_\mathbf{a}\mathbf{q}. \tag{9}$$

Note that this computation requires only a few matrix multiplications and the inversion of a relatively small matrix. No iterative optimization [3] is involved in the process. This is why our method is fast.

2.6 Regularization

Independently of the weighting matrix $\mathbf{W}$ we observed that the matrix $\mathbf{K}$ is sometimes close to singular. This is a sort of *generalized aperture problem* and is due mainly to the intrinsic ambiguity between small horizontal translation and vertical rotation, and between small vertical translation and horizontal rotation. Moreover, we found that a coupling exists between some of the illumination templates and the warping templates.

Fig. 5 shows the matrix $\mathbf{M}^T\mathbf{M}$ for a typical sequence. Each square in the figure corresponds to an entry in the matrix. Bright values correspond to large values in the matrix, dark squares have small values in the matrix. If the system were perfectly decoupled, then all off-diagonal elements would be dark. In general, brighter off-diagonal elements indicate a coupling between parameters.

By looking at the figure, it is possible to see the coupling that can cause ill-conditioning. The top-left part of the matrix is diagonal because it corresponds with the orthogonal illumination basis vectors. This is not true for bottom-right block of the matrix. This block of the matrix corresponds with the warping basis images. Note that the coupling between warping parameters and appareance parameters is weaker than the coupling within the warping parameter space. This last kind of coupling could be reduced with a different warping function parametrization.

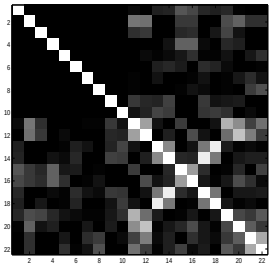


Figure 5: Example of matrix $\mathbf{M}^T\mathbf{M}$.

Such couplings can lead to instability or ambiguity in the solutions for tracking. To alleviate this problem, we regularize our system. We define the regularizer by adding a penalty term to the image energy shown in the previous section, and then minimize with respect to $\mathbf{c}$ and $\mathbf{q}$:

$$\begin{aligned} E &= \|(\mathbf{T} - \mathbf{T}_0) - (\mathbf{B}\mathbf{q} + \mathbf{U}\mathbf{c})\|_W + \gamma_1[\mathbf{c}^T\Omega_a\mathbf{c}] \\ &\quad +\gamma_2[\mathbf{a}^- + \mathbf{N}_\mathbf{a}\mathbf{q}]^T\Omega_w[\mathbf{a}^- + \mathbf{N}_\mathbf{a}\mathbf{q}] \end{aligned} \tag{10}$$

The diagonal matrix Ω_a is the penalty term associated with the appearance parameter $\mathbf{c}$, and Ω_w is the penalty associated with the warping parameters $\mathbf{a}$.

We can define:

$$\mathbf{p} = \begin{bmatrix} \mathbf{0} \\ \mathbf{a}^- \end{bmatrix}; \mathbf{N} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{N}_\mathbf{a} \end{bmatrix}; \tag{11}$$

$$\Omega = \begin{bmatrix} \frac{\gamma_1}{\gamma}\Omega_a & \mathbf{0} \\ \mathbf{0} & \frac{\gamma_2}{\gamma}\Omega_w \end{bmatrix} \tag{12}$$

and then rewrite the energy as:

$$E = \|\mathbf{R} - \mathbf{M}\mathbf{x}\|_W + \gamma[\mathbf{p} + \mathbf{N}\mathbf{x}]^T\Omega[\mathbf{p} + \mathbf{N}\mathbf{x}]. \tag{13}$$

By taking the gradient of the energy with respect to $\mathbf{x}$, and equating it to zero we get:

$$\mathbf{x} = \tilde{\mathbf{K}}\mathbf{R} + \mathbf{Q}\mathbf{p}, \tag{14}$$

where $\mathbf{K} = [\mathbf{M}^T\mathbf{W}^T\mathbf{W}\mathbf{M} + \gamma\mathbf{N}^T\Omega\mathbf{N}]^{-1}\mathbf{M}^T\mathbf{W}^T\mathbf{W}$ and $\mathbf{Q} = \gamma[\mathbf{M}^T\mathbf{W}^T\mathbf{W}\mathbf{M} + \gamma\mathbf{N}^T\Omega\mathbf{N}]^{-1}\mathbf{N}^T\Omega$.

As before, if we are interested only in the warping parameter estimate, then we can save computation time by solving only for the $\mathbf{q}$ part of $\mathbf{x}$. We can then find $\Delta\mathbf{a}$.

The choice of a diagonal regularizer implicitly assumes that the subvectors $\mathbf{c}$ and $\mathbf{q}$ are independent. In practice this is not the case. However, our experiments consistently proved that the performance of the regularized tracker is considerably superior with respect to the unregularized one.

The matrices Ω_a and Ω_w were chosen for the following reasons. Recall that the appearance basis $\mathbf{U}$ is an eigenbasis for the texture space. If Ω_a is diagonal and with elements equal to the inverse of the corresponding eigenvalues, then the penalty term $\mathbf{c}^T\Omega_a\mathbf{c}$ is proportional to the *distance in feature space*[15]. This term thus prevents an

artificially large illumination term from dominating, and misleading the tracker.

The diagonal matrix Ω_w is the penalty associated with the warping parameters. We assume that the parameters are independently Gaussian distributed around the initial position . We can then choose Ω_w to be diagonal, with diagonal terms equal to the inverse of the expected variance for each parameter. In this way we prevent the parameters from *exploding* when the track is lost. Our experience has shown that this term generally makes it possible to swiftly recover if the track is lost. We defined the standard deviation for each parameter as a quarter of the range that keeps the model entirely visible (within the window).

Note that this statistical model of the head motion is particularly suited for video taken from a fixed camera (for example a camera on the top of the computer monitor). In a more general case (for example to track heads in movies) a random walk model[1, 12] would probably be more effective. Furthermore, the assumption of independence of the parameters could be removed and the full non-diagonal 6×6 covariance matrix estimated from example sequences.

3 Implementation details

To allow for larger displacements in the image plane we implemented our system using a multiscale framework. The warping parameters are initially estimated at the higher level of a Gaussian pyramid and the parameters are propagated to the lower level. In our implementation we found that a three level pyramid was sufficient.

We represented the cylindrical model as a set of texture mapped triangles in 3D space. When the cylinder is superimposed onto the input video frame, each triangle in image plane maps the underlying pixels of the input frame to the corresponding triangle in texture map. The confidence map is generated using a standard triangular area fill algorithm. The map is first initialized to zero. Then each visible triangle is rendered into the map with a fill value corresponding to the confidence level. This approach allows the use of standard graphics hardware to accomplish the task.

The illumination basis has been computed from a MIT database[15] of 1,000 aligned frontal view of faces under varying lighting conditions. Since all the faces are aligned, we had to determine by hand the position of the cylinder only once and then used the same warping parameters to compute the texture corresponding to each face. Finally, the average texture was computed and subtracted from all the textures before computing the SVD.

4 Experimental results

The system has been extensively tested using twenty sequences. These specific sequences were selected because they caused the previous SSD [13] to fail (tracking was lost or diverged). Ten of the sequences were captured via a low quality SGI O2 camera positioned atop the computer monitor in a poorly illuminated environment with dominant directional lights. The other ten sequences were captured with a Sony consumer camera on a tripod in a well-illuminated set. The images in the video sequences had a pixel resolution of 320×240. In each sequence, the head occupied between 20 and 50 percent of the total frame area.

The sequences were selected in such a way that a wide variety of common head movements were included in the test set. Head motions present in the test sets include significant out of plane rotations (up to about 60^o). A few of them were of a people telling stories using American Sign Language (ASL). These sequences include very rapid head motion and frequent occlusions of the face with the hand(s). Each of the twenty video sequences was between 8 and 15 seconds in duration.

A number of different experiments were conducted. The first set of experiments was intended to evaluate the improvements gained in adding the illumination basis to the SSD tracker described in [13]. Note that to test the improvement gained by modeling illumination, the regularization term was omitted in this first experiment. All test sequences were tracked using both the old and new formulations, and then tracking performance compared.

For both classes of sequences the new tracker behaved consistently better. However, as expected, the improvement over the standard SSD tracker was much bigger for the first class of sequences (the ones with prominent directional light). In particular for the first class of sequences, the introduction of the lighting correction term greatly improved the precision of head motion estimate throughout tracking. The new formulation also tended to be more stable, allowing accurate tracking over longer sequences. It achieved accurate tracking over the full sequences in many cases that previously diverged in the previous SSD tracker formulation [13].

Only in five out of twenty sequences was the track lost at the same point in the sequence in both the old and new tracking formulations. These cases were characterized by very fast head motion or extreme rotation. One example is shown in Fig. 6. In the other cases, the new formulation was often able to reliably track the target during the whole sequence.

The second experiment was intended to test the improvement gained by incorporating both the illumination and regularization terms in the tracker. The stability further increased using the regularization energy term. The system successfully tracked through the points in all twenty video sequences where the previous SSD tracker diverged. With the added regularization term, tracking was stable enough to allow stable and accurate tracking over the full 8-15 sec. sequence in 17 out of 20 test cases.

Table 1 shows the average results for the experiments conducted. In particular the table shows averages of length of the sequences, number of frames that were tracked before losing the track, and residual error for the three types of tracker analyzed. The averages are taken over the two classes of sequences. Note that the gain in performance due to the use of the illumination templates is much smaller for sequences in class 2 that were taken in a uniformly illuminated environment. The positive effect of the regularizer was strong in both classes of sequences.

		# of frames tracked			Residual Error		
Class	Ave. Length	T1	T2	T3	T1	T2	T3
1	343	102	207	307	82	49	48
2	320	182	190	297	375	324	299

Table 1: Cumulative results of experiments. T1 is the standard SSD tracker, T2 is the tracker using illumination templates and T3 is the regularized version of T2.

The top row of Fig. 6(a) shows a few example frames of a typical sequence taken from the SGI O2 camera. The subsequent rows of Fig. 6 show tracking results for: (b) the old SSD tracker, (c) SSD with additional illumination term, and (d) SSD with both illumination and regularization term. This is an example of one of the few cases where the tracker with the lighting term lost the track at almost the same point in the sequence as the older SSD formulation. In any case, the residual error was much lower, and then the precision of the tracking much better. The regularized tracker was able to track all 450 frames of the sequence.

Fig. 7 shows a few frames from a long sequence that was collected with the Sony camera. Ground truth for this sequence was simulatenously collected via a "Flock of Birds" 3D tracker. The transmitter was attached to the subject's head. Tracking parameters were recovered using the new tracking formulation that includes lighting correction and regularization terms. The graphs show the estimated rotation and translation parameters during tracking compared to ground truth.

The tracker is extremely fast. We expect to track video at near frame rate using the technique proposed in this paper. The demo version we implemented using the openGL library and hardware texture mapping runs on an SGI O2 R5000 at about 4Hz reading the input stream from a movie file. This figure is for unoptimized C++ code, and most of the time is spent in reading/uncompressing movie files, and graphical display information that is not essential to the tracking system. By our calculations, if we subtract this non-essential computation and I/O overhead, our tracker could easily run at frame rates. We are therefore confident that when sending the video stream directly into texture map memory, a real-time implementation will be possible.

5 Conclusions

We proposed a fast, stable and accurate technique for 3D head tracking in presence of varying lighting conditions. We presented experimental results that show how our technique greatly improves the standard SSD tracking without the need of a subject-dependent illumination basis or the use of iterative techniques. Our method is accurate and stable enough that the estimated pose and orientation of the head is suitable for application of head gesture recognition and visual user interfaces.

The texture map provides a stabilized view of the face that can be used for facial expression recognition, and other applications requiring that the position of the head is frontal view and almost static. Moreover, when we compute the full $\mathbf{x}$ vector we have also a very compact representation of the head that could be used for model-based very low bitrate video coding.

In the future we plan to develop a version of our method that employs robust cost functions. We suspect that this could further improve the precision and stability of the tracker in presence of occlusions. A real-time implementation of the current formulation is also under development.

Acknowledgments

This work was supported in part through Office of Naval Research Young Investigator Award N00014-96-1-0661, and National Science Foundation grants IIS-9624168 and EIA-9623865.

References

[1] A. Azarbayejani, T. Starner, B. Horowitz, and A. Pentland. Visually controlled graphics. *PAMI* 15(6), 1993.

[2] S. Basu, I. Essa, and A. Pentland. Motion regularization for model-based head tracking. *ICPR*, 1996.

[3] M.J. Black and A. Jepson. Eigentracking: Robust matching and tracking of articulated objects using a view-based representation. *IJCV*, 26(1):63–84, 1998.

[4] M.J. Black and Y. Yacoob. Tracking and recognizing rigid and non-rigid facial motions using local parametric models of image motions. *ICCV*, 1995.

[5] T.F. Cootes, G.J. Edwards, and C.J. Taylor. Active appearance model. ECCV, 1998.

[6] J.L. Crowley and F. Berard. Multi-modal tracking of faces for video communications. *CVPR*, 1997.

[7] D. DeCarlo and D. Metaxas. The integration of optical flow and deformable models with applications to human face shape and motion estimation. *CVPR*, 1996.

[8] I.A. Essa and A.P. Pentland. Coding analysis, interpretation, and recognition of facial expressions. *PAMI*, 19(7):757–763, 1997.

[9] M. Gleicher. Projective registration with difference decomposition. *CVPR*, 1997.

[10] G.D. Hager and P.N. Belhumeur. Efficient region tracking with parametric models of geometry and illumination. *PAMI*, 20(10):1025–1039, 1998.

[11] P. Hallinan. A low-dimensional representation of human faces for arbitrary lighting conditions. CVPR, 1994.

[12] T.S. Jebara and A. Pentland. Prametrized structure from motion for 3d adaptative feedback tracking of faces. *CVPR*, 1997.

[13] M. La Cascia, J. Isidoro, and S. Sclaroff. Head tracking via robust registration in texture map images. *CVPR*,1998.

[14] H. Li, P. Rovainen, and R. Forcheimer. 3-d motion estimation in model-based facial image coding. *PAMI*, 15(6):545–555, 1993.

[15] B. Moghaddam and A. Pentland. Probabilistic visual learning for object representation. *PAMI*, 19(7), July 1997.

[16] N. Olivier, A. Pentland, and F. Berard. Lafter: Lips and face real time tracker. *CVPR*, 1997.

[17] H.A. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *PAMI*, 20(1):23–28, 1998.

[18] A. Schödl, A. Haro, and I. Essa. Head tracking using a textured polygonal model. *Proc. Workshop on Perc. User Interfaces*, 1998.

[19] S. Sclaroff and J. Isidoro. Active blobs. *ICCV*, 1998.

[20] A. Shashua. *Geometry and photometry in 3D visual recognition*. PhD thesis, MIT, 1992.

[21] D. Terzopoulos and K. Waters. Analysis and synthesis of facial image sequences using physical and anatomical models. *PAMI*, 15(6):569–579, 1993.

[22] Y. Yacoob and L.S. Davis. Computing spatio-temporal representations of human faces. *PAMI*, 18(6):636–642, 1996.

[23] A.L. Yuille, D.S. Cohen, and P.W. Hallinan. Feature extraction from faces using deformable templates. *Proc. ICPR*, 1994.

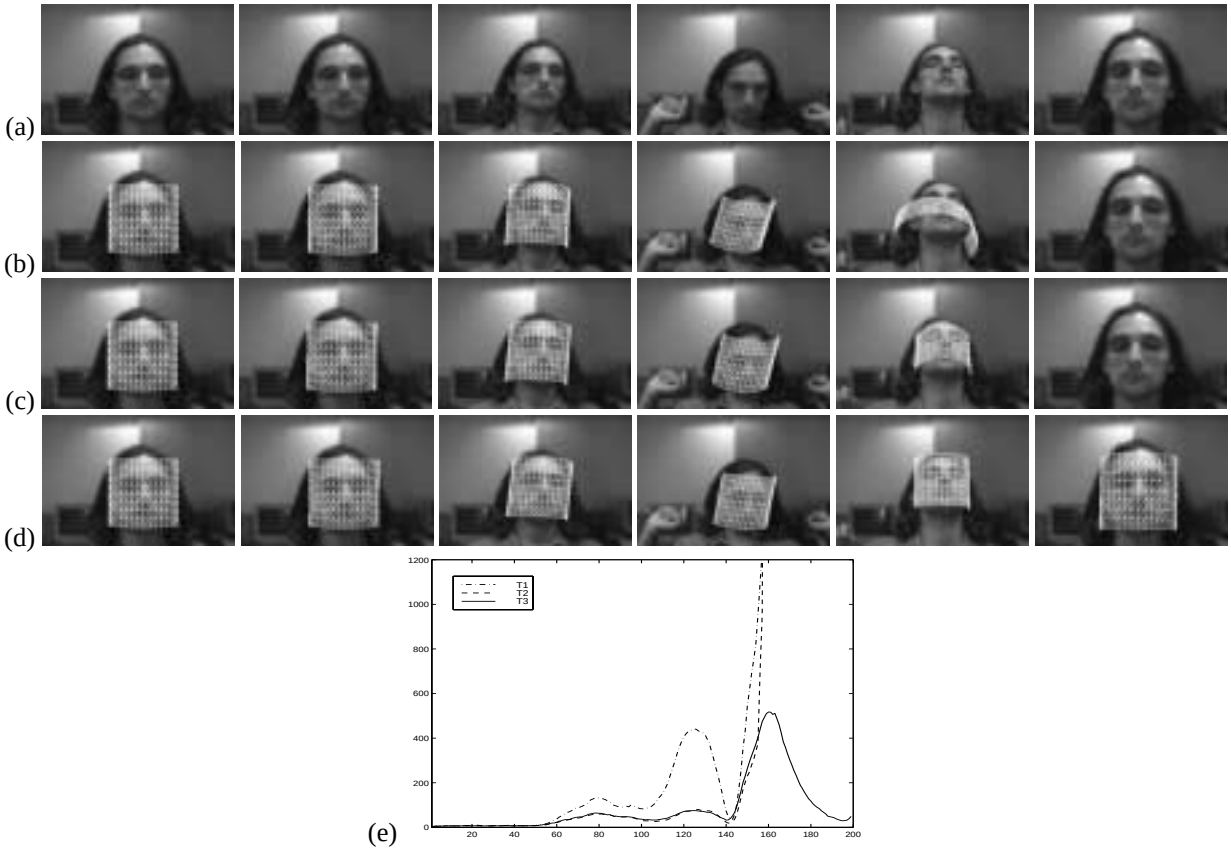


Figure 6: Example input video frames (a) taken from a test sequence that caused some of the trackers to diverge. The resulting tracking for each of the three trackers tested is shown below each input image taken from the sequence: (b) head tracking using standard SSD tracking, (c) SSD with lighting correction, and (d) SSD with lighting correction and regularization. The frames reported are 0, 40, 80, 120, 160, and 200 (left to right). The Residual error for the different trackers is shown in (e) and is indicated respectively with T1, T2 and T3.

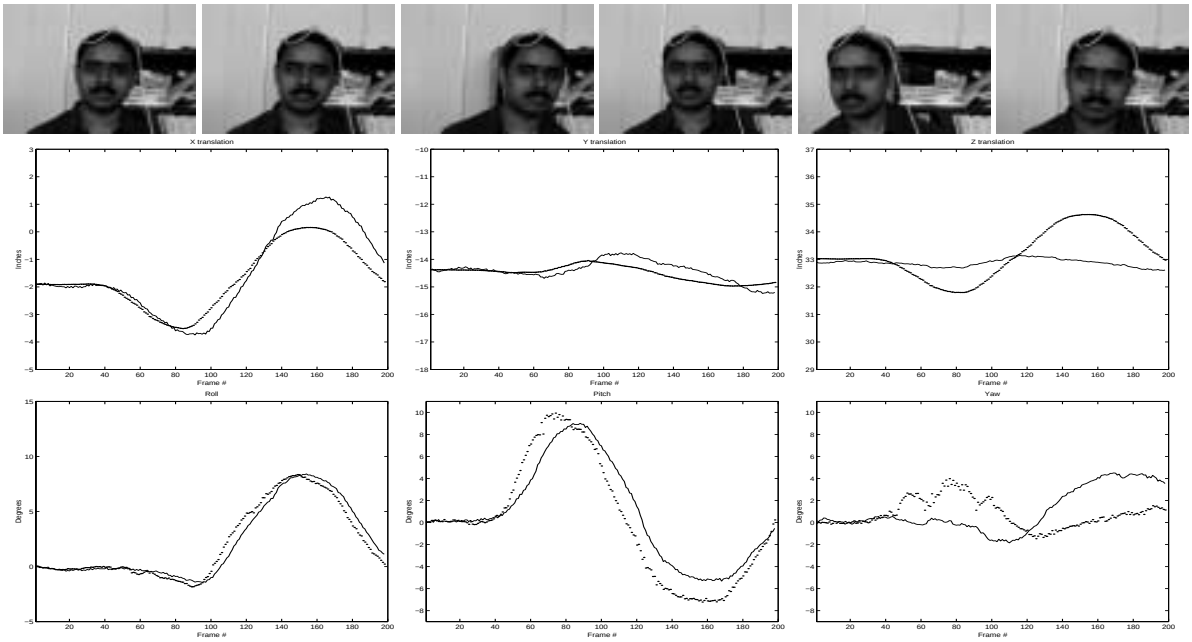


Figure 7: Frames taken from test sequence in which ground truth was collected via a 3D "Flock of Birds" sensor. The graph shows estimated translations and rotations compared with ground truth. The measures are in inches and degree, the solid line is the ground truth.