Pattern Analysis & Applications (1999)2:82–91

Tracking in 3D: Image Variability Decomposition for Recovering Object Pose and Illumination*

Peter N. Belhumeur[1] and Gregory D. Hager[2]

[1]*Department of Electrical Engineering;* [2]*Department of Computer Science, Center for Computational Vision and Control, Yale University, New Haven, CT, USA*

Abstract: As an object moves through space, it changes its orientation relative to the viewing camera and relative to light sources which illuminate it. As a consequence, the images of the object produced by the viewing camera may change dramatically. Thus, to successfully track a moving object, image changes due to varying pose and illumination must be accounted for. In this paper, we develop a method for object tracking that can not only accommodate large changes in object pose and illumination, but can recover these parameters as well. To do this, we separately model the image variation of the object produced by changes in pose and illumination. To track the object through each image in the sequences, we then locally search the models to find the best match, recovering the object's orientation and illumination in the process. Throughout, we present experimental results, achieved in real-time, demonstrating the effectiveness of our methods.

Keywords: Pose estimation; Illumination modelling; Image sequence analysis; Tracking

1. INTRODUCTION

The challenge in visual tracking is to quickly and accurately determine the image position or configuration of a target (an object or a region on the surface of an object) as the target moves through a camera's field of view. This problem becomes particularly challenging when the target is large enough and of sufficient geometric complexity to exhibit the full range of change in appearance caused by both geometry and illumination. A more difficult challenge is to also *estimate* the (3D) pose and illumination of the object through an image sequence. If this determination can be made reliably, it will not only increase the accuracy of the tracking process, but should also considerably increase the number of applications for which visual tracking is of use. For example, estimates of a tracked object's pose and illumination may prove invaluable in the development of systems for augmented reality, human/computer interfaces and surveillance.

In this paper, we develop a region tracking algorithm which efficiently and explicitly computes the three-dimensional pose and illumination parameters of its target. To do this, we explicitly model how changes in pose and illumination of an object, or target region on an object, produce changes in the observed images. We decompose the variability into its component parts – pose and illumination – each of which, when analysed separately, is well behaved. We call this approach to handling image changes *Image Variability Decomposition*.

In the case of pose variation, we exploit the results of Ullman and Basri [1], Jacobs [2] and Tomasi and Kanade [3] which show that, under weak perspective, the set of image coordinates of a rigidly moving object lies in low-dimensional linear subspace of the image coordinate space. This differs from earlier work on handling pose variation by ourselves and others [4–6], in that image coordinate deformations are not restricted to be affine. For planar or nearly planar targets, affine models work quite well. Yet, most targets are not planar, and consequently, as the target rotates in space affine models quickly break down. The

Received: 9 December 1998
Received in revised form: 23 December 1998
Accepted: 23 December 1998
* Presented at ICAPR '98

models described in this paper are specific to the target object's 3D geometry. As a consequence, not only is the tracking of the object's position more reliable, but we are also able to recover the 3D orientation of the object through the image sequence.

In the case of illumination variation, we exploit the results in Belhumeur and Kriegman [7], which show that the set of images of an object seen from a fixed viewpoint, under all possible illumination conditions, is a convex cone in the space of images, and this cone (termed the 'illumination cone') can often be constructed from as few as three images. This differs from earlier work [4], in that the changes in the image of the target region are not assumed to lie anywhere within a low-dimensional linear subspace, but are restricted to lie on or within a convex cone. This improvement models object shadowing, allowing the tracker to maintain lock under greater variation in illumination than before.

Given parametric models for image changes due to motion or changes in illumination, we show that it is possible to develop an efficient algorithm for perturbing state (pose and illumination) estimates of targets to fit observed data. The result is an efficient algorithm *which computes 3D object pose and illumination geometry in real-time on live video images*. At the end of the paper, we present two sets of experiments. The first experiment quantifies the accuracy of the pose determination. The second experiment qualitatively gauges the performance of the algorithm for tracking a human face.

The remainder of this article is structured as follows. Section 2 discusses related work. Section 3.1 describes how we construct the pose model, and Section 3.2 describes how we construct the illumination model for a target object. Section 4 describes how we combine these models to build a tracking system. Finally, Section 5 presents experimental results from tracking both a 'calibration' sphere and a human face.

2. RELATED WORK

Visual tracking can be usefully categorised as region- or feature-based. Most of the tracking systems that are able to use and/or compute 3D pose are feature-based [8–14]. These systems overcome the potentially deleterious effects of changes due to illumination by concentrating on a sparse set of edges or corners in images, i.e. discontinuities in the image intensity. However, models using only local features are impoverished – they throw out a great deal of information present in the original image. Furthermore, since features are typically local, feature-based tracking methods require good dynamical models and effective search methods to avoid feature matching ambiguity [13,9]. If the objects do not have piecewise constant albedo patterns, the detection and localisation of the edge or corner points are sensitive to changes in illumination.

Previous work on region-based tracking has almost entirely concentrated on the use of 2D linear models of image deformations, ranging from simple rigid translation [15], to affine or low-order polynomial deformations [6,4,16,17]. Affine models are correct for planar surfaces viewed under orthographic projection. More recently, 'appearance-based' approaches have been developed in an effort to use intensity information to model or learn a representation that captures a large set of the possible images of an object under pose and/or illumination variation [18–22]. These methods have been directly applied to tracking problems [18], and have also been applied in combination with image-level deformations [5]. Although these methods can be used to both track an object and to recover object pose, they require that the target has been previously observed under similar conditions.

The Image Variability Decomposition paradigm differs in that by decomposing image variability, we are able to uncover low-dimensional generative structures for the set of images. Thus, unlike appearance-based methods, it is not necessary to have seen the object under all of the seemingly infinite possible permutations of lighting conditions and pose. In addition, since the changes in object appearance under pose and illumination variation are in general not linear in image space, the dimensionality of the representation is likely to be far higher than that needed to describe the underlying generative structure of images. Feature-based methods require a method for choosing a set of features to track, and additional knowledge of the object coordinates of the features in order to compute pose. The Image Variability Decomposition approach can be seen as a way of using *all* of the grey-value structure of images without committing to a particular set of features. The arguments used here in support of region-based tracking are, of course, identical to arguments in support of region-based determination of binocular stereo correspondence.

3. IMAGE VARIABILITY DECOMPOSITION

In the subsections below, we describe how we model the image variation due to changes in pose and illumination. In particular, we demonstrate how, from as few as three images of a target, it is possible to generate a complete set of images of the target over a large range of pose and illumination variation.

3.1. Pose

In this section, we apply the results of Ullman and Basri [1], Jacobs [2] and Tomasi and Kanade [3] which show that, under weak perspective, the set of image coordinates of a rigidly moving object lies in low-dimensional linear subspace of the image coordinate space. Unlike Tomasi and Kanade [3], however, our goal is not to explicitly determine the object's structure. Rather, our approach follows that of Ullman and Basri [1], in that we want to use the low-dimensionality of the image coordinate variation to predict what the object will look like over a range of viewing directions.

To start, let us represent the structure of an object by the positions in space of a collection of n points on the

object's surface. Let us choose the n points on the object's surface so that they map to a dense collection of pixels within an image. Note again that we choose a dense collection of points – not a select few fiducial points – in representing the object. We call this collection of points the 'target region'. The pose of the object relative to an arbitrary fixed coordinate system can be represented by a point in $\mathbb{R}^3 \times SO(3)$. For each position and orientation of the object, we can construct a $3n$-dimensional vector describing the positions in space of the n points on the objects surface, i.e. for each orientation and translation, we have a corresponding point in a $3n$-dimensional coordinate space.

The set of all possible 3D object coordinates under 3D rotation and translation is then a 6D manifold in this $3n$-dimensional coordinate space. (Due to possible symmetries in the object, the set of coordinates may not be a manifold, but a stratified set [23] composed of manifolds and singular sets of dimension six or less. For simplicity we will refer to these sets as manifolds.)

As evident from the work of Ullman and Basri [1], under scaled orthographic projection, the set of 2D image coordinates lie in an 8D linear subspace of the $2n$-dimensional image coordinate space. To see this, let $\mathbf{p} \in \mathbb{R}^3$ represent a point on the surface of an object, $\mathbf{R} \in SO(3)$ represent a rotation matrix, and $t \in \mathbb{R}^3$ represent a translation vector. Under scaled orthography, we can write the projection $(x,y)^T \in \mathbb{R}^2$ of $\mathbf{p}$ as

$$\begin{bmatrix} x \\ y \end{bmatrix} = s\mathbf{P}(\mathbf{R}\mathbf{p}+\mathbf{t}) \tag{1}$$

where s is a fixed scaling factor, and $\mathbf{P}$ is of the form

$$\mathbf{P} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

We can simplify this equation, and rewrite it in the form

$$\begin{bmatrix} x \\ y \end{bmatrix} = \mathbf{A}\mathbf{p} + \mathbf{d} \tag{2}$$

where $\mathbf{A}$ is a 2×3 matrix and $\mathbf{d} \in \mathbb{R}^2$.

It follows that the set of all image coordinates viewed under scaled orthography lie on a 6D manifold embedded within the 8D linear subspace represented by the (unconstrained) matrix $\mathbf{A}$ and vector $\mathbf{d}$. We call this 6D manifold the 'pose manifold'. As noted in the work of Ullman and Basri [1], because of the symmetry between the x and y image coordinates, both the x and y coordinates lie in the same 4D linear subspace of an n-dimensional coordinate space. This 4D subspace, which we call the 'pose subspace', defines the 8D linear subspace in which the pose manifold is embedded.

Returning to our problem, let vectors $\mathbf{x}$ and $\mathbf{y}$ describe the x-coordinates and the y-coordinates of a dense collection of n pixels in some 'model' image. Let us call this collection of pixels the 'target region'. As discussed above, if the object is viewed orthographically, the resulting vector of x-coordinates and the resulting vector of y-coordinates are each restricted to lie in the same 4D linear subspace, i.e. the pose subspace. Let $\hat{\mathbf{x}},\hat{\mathbf{y}},\hat{\mathbf{z}}$, and $\mathbf{1}$ (where $\mathbf{1}$ is an n-dimensional vector of ones) be the basis vectors for the 4D pose subspace. Let the vector function $\mathbf{f}(\mathbf{b})$ represent a new vector of x-coordinates and the vector function $\mathbf{g}(\mathbf{c})$ represent a new vector of y-coordinates, after a rotation and translation of the target region. Then we have

$$\begin{aligned} \mathbf{f}(\mathbf{b}) &= b_1\hat{\mathbf{x}} + b_2\hat{\mathbf{y}} + b_3\hat{\mathbf{z}} + b_4\mathbf{1} \\ \mathbf{g}(\mathbf{c}) &= c_1\hat{\mathbf{x}} + c_2\hat{\mathbf{y}} + c_3\hat{\mathbf{z}} + c_4\mathbf{1} \end{aligned} \tag{3}$$

The eight coefficients in $\mathbf{b}$ and $\mathbf{c}$ together determine a point in the 8D linear subspace in which the 6D pose-manifold is embedded. For rigid motions, this point must lie on the pose manifold.

To compute the basis vectors for the pose subspace, let us assume we have chosen n pixels defining the target region of an object. We then acquire a second image of the object after the object undergoes a small rotation (and translation) in space, with the proviso that the axis of rotation is not the camera's optical axis. From the correspondences of each pixel in the first image with those in the second image, we can determine the pose subspace [3]. Note that determining the pose subspace only fixes the *affine* structure of the object [25]. To determine the 3D Euclidean structure of the object or, equivalently, to determine the pose manifold, a third image is needed. (This, of course, was implicit in the work of Tomasi and Kanade [3], which demonstrated how these subspaces could be computed efficiently and, from them, the 3D coordinates of the points determined.)

Yet we stress that the goal of pose decomposition is not to necessarily determine the precise 3D structure of objects. Rather, the goal is to find a pose subspace or pose manifold (if more than two images are available) that can be used to accurately predict images of the object under a range of viewpoints. These objectives are not the same, as points on the objects surface in regions of constant intensity will yield inaccurate structure measurements, but will have little effect on the set of images modelled by pose deformations.

What this all means is that from as few as two images of the object seen from slightly different directions, we can determine the images of the object under a large range of viewing directions. Thus, in the case of scaled orthographic projection, there is indeed a simple, generative structure to the set of images of an object under varying pose.

Figure 1 gives a demonstration of an approximation to the pose subspace and resulting pose manifold for a human face. Here we have used 20 images, of which four are shown, to generate the pose subspace in which the pose manifold is embedded. The basis vectors spanning the pose subspace were crudely determined by first determining the correspondence using the optical flow techniques of Lucas and Kanade [24]. (The methods for establishing a dense correspondence can be improved by borrowing from work in binocular stereopsis [26].) We then randomly sampled the pose manifold, recreating the images of the object at the new pose. The pictures at the bottom of the figure are artificially generated images of the human face. Note that there is little variation in the original images shown in the top, left of the figure, yet the artificially generated images

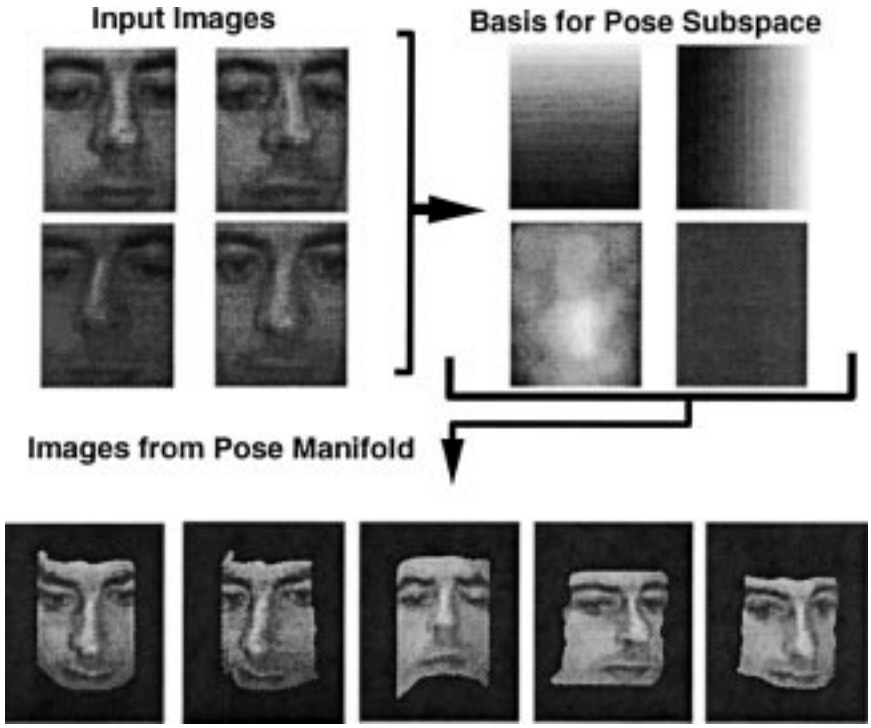

Fig. 1. Pose decomposition. In this figure, we have used 20 images, four of which are shown here, to generate the pose subspace in which the pose manifold is embedded. The basis vectors spanning the pose subspace were crudely determined by first determining the correspondence using the optical flow techniques of Lucas and Kanade [24]. We then randomly sampled the pose manifold, recreating the images of the object at the new pose. The pictures at the bottom of the figure are artificially generated images of the human face. Note that there is little variation in the original four images shown in the top, left of the figure, yet the artificially generated images demonstrate a large range of pose variation.

demonstrate a wider range of pose variation. Furthermore, while we have shown only five artificially generated images, we can actually generate any image over a continuum of poses.

3.2. Illumination

Image variability due to illumination has received relatively little attention in the computer vision literature. Yet the same object seen from the same pose can appear drastically different depending the directions and strengths of the light sources [27,28]. This fact has implications not only for object recognition, but also for object tracking.

The variability due to illumination may be much larger than that due to pose as the set of possible lighting conditions is infinite dimensional. Still, it turns out that while the set of images of an object under varying illumination may be large, it has a great deal of implicit structure. The set of images of an object with arbitrary reflectance functions seen under arbitrary illumination conditions is a convex cone in $\mathbb{R}^n$, where n is the number of pixels in each image [7]. And if the object has a convex shape and a Lambertian reflectance function, the set of images under an arbitrary number of point light sources at infinity is a convex polyhedral cone in $\mathbb{R}^n$, which can be determined *exactly* from as few as three images, see again [7].

To express the stated relations through equations, let us assume we have a target object seen from a fixed point, but under varying illumination. Again, let vectors x and y describe the x-coordinates and the y-coordinates of a dense collection of n pixels in some image referred to as the 'target region'. Let $R(x,y)$ denote the vector of brightness values in the target region. Due to its convexity, the illumination cone of the target region, i.e. the set of images of the object under all lighting conditions, is defined by a collection of extreme rays (images) $R_j(x,y)$ with $j = 1,\ldots,m$. (The number of extreme rays m may be quite large, but we approximate the illumination cone by choosing a small subset.) Any model image of the target region seen from the same viewpoint, but under arbitrary illumination, can then be constructed from a convex combination of the extreme rays:

$$M(x,y,a) = \sum_{j=1}^{m} a_j R_j(x,y) \tag{4}$$

where $a_j \geq 0 \ \forall j$.

The determination of the extreme rays (the R_j) defining the convex cone can be done in two ways. First, we can simply gather a collection of images of the target regions illuminated by point light sources of different directions, and use these to define the extreme rays. This method is similar to that of Murase and Nayar [18], except the illumination cone representation implicitly models multiple light

sources, since it allows for convex combinations of the extreme rays (images). A drawback of this method (and that of Murase and Nayar [18]) is that many images are needed to construct the model.

A second method for determining the extreme rays is to gather only a small number of images of the object under varying illumination, and use these to generate the complete set of extreme rays. If the object's surface has a Lambertian reflectance function and the shape of the object is roughly convex, then the extreme rays defining the illumination cone can be generated from a 3D linear subspace in the *n*-dimensional image space [7]. Furthermore, this 'illumination subspace' can be generated from as few as three images [29].

Even if the surface reflectance function is more general than Lambertian and the object is non-convex in shape, this method still seems to work quite well. Figure 2 gives a demonstration of such an approximation for varying illumination of a face. Here we have used six images, of which four are shown, to generate what we call the illumination subspace which, in turn, generates the extreme rays which define the illumination cone. We then randomly sampled the illumination cone. The pictures at the bottom of the figure are artificially generated images of the face.

4. THE TRACKING SYSTEM

In this section, we describe our procedure for tracking the target region through an image sequence. We pose the tracking problem as one of finding an optimal (according to the criterion stated below) pair of trajectories through the illumination cone and along the pose manifold.

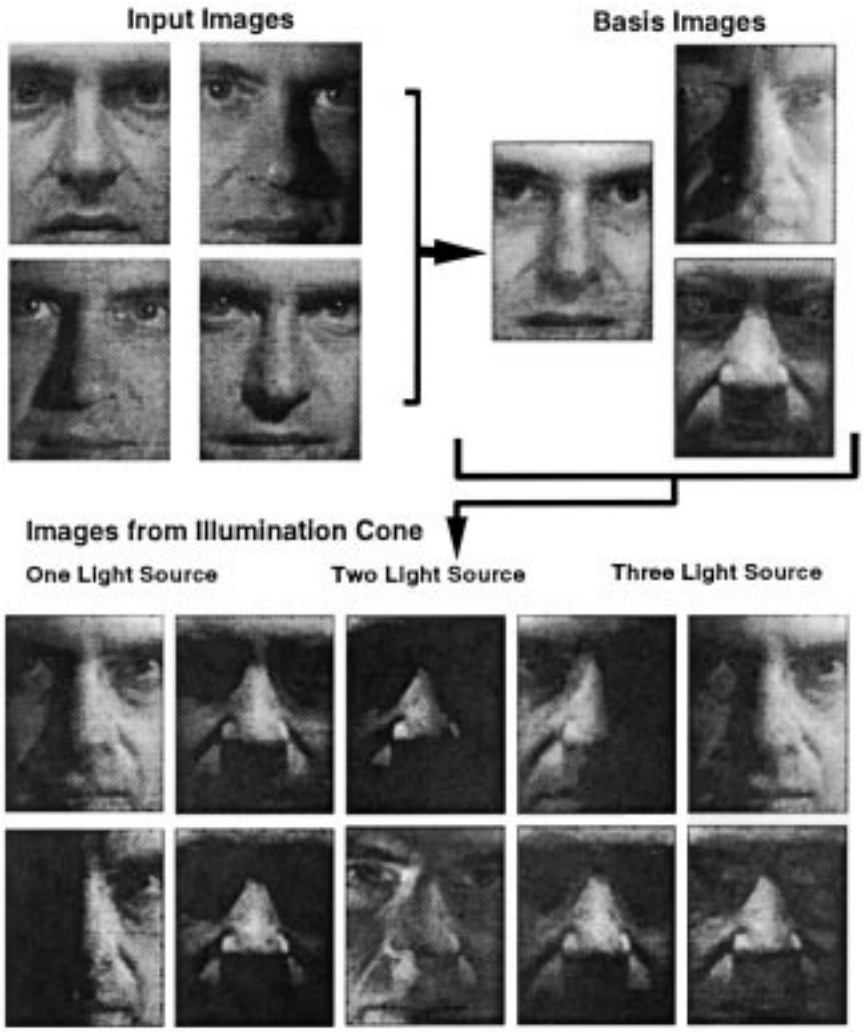

Fig. 2. Illumination decomposition. The set of all possible *n*-pixel images of an object in fixed pose under variable illumination including shadows is a convex cone in $\mathbb{R}^n$ (the image space). When the object surface reflectance is Lambertian, the cone can be *exactly* determined from as few as three images. In this case, six images, of which four are shown, are used to construct the cone of a target region centered on a human face. Basis images for this 3D illumination subspace can be estimated using SVD from three or more images; the direction of light sources is not needed. The extreme rays of the illumination cone can then be constructed from the illumination subspace. At the bottom of the figure are artificially generated images of the face that lie in the cone.

Let $I(x,y,t)$ denote the brightness value at the location $(x,y)^T$ in an image acquired at time t. If, as before, x and y are the x and y coordinates of n pixels defining the target region, let $I(\mathrm{x},\mathrm{y},t)$ denote an image of the target region. In the absence of noise, and if our pose and illumination models are exact, then it follows that the image of the target region must satisfy the following relation:

$$I(\mathrm{x},\mathrm{y},t) = M(\mathrm{f}(\mathrm{b}),\mathrm{g}(\mathrm{c}),\mathrm{a}) \tag{5}$$

for some $\mathrm{a} \in \mathbb{R}^m$, $\mathrm{b} \in \mathbb{R}^4$ and $\mathrm{c} \in \mathbb{R}^4$. The expression on the right-hand side can be interpreted as follows: an image is first synthesized from the extreme rays of the illumination cone, and then this image is 'warped' by a deformation selected from the pose manifold so that the result is identical to an observed image.

It follows that the temporal correspondence of the target region across an image sequence can be determined by finding the pose parameters and the illumination parameters that minimise

$$O(\mathrm{a},\mathrm{b},\mathrm{c}) = \|I(\mathrm{x},\mathrm{y},t) - M(\mathrm{f}(\mathrm{b}),\mathrm{g}(\mathrm{c}),\mathrm{a})\|^2 \tag{6}$$

As recently shown [4], it is possible to efficiently compute the parameters in a problem of this form. Briefly, the procedure is to linearise the above expression and solve for the pose and illumination parameters incrementally in each frame. The resulting system is then factored into time-invariant and time-varying terms. As a result, the online portion of the estimation procedures can be simplified to a form which is easily computed in real-time.

The result of this process is both a vector of coefficients characterising the object's pose and a vector of coefficients describing a point in the illumination cone. To determine the actual pose, we choose a specific representation for rotations [30] as

$$\mathrm{R} = \mathrm{R}_z(\gamma)\mathrm{R}_y(\beta)\mathrm{R}_x(\alpha) \tag{7}$$

where $\mathrm{R}_i(\theta)$ is a rotation by an angle θ about the axis i. We are particularly interested in the out-of-plane rotations α and β, which we subsequently refer to as 'tilt' and 'pan', respectively. After substituting Eq. (7) into Eq. (1) and equating the terms of the affine representation, simple algebra yields the vector equation

$$\begin{bmatrix} b_3 \\ c_3 \end{bmatrix} = \begin{bmatrix} -S_\beta C_\alpha \\ -S_\alpha \end{bmatrix} \tag{8}$$

which is readily solved for the pan and tilt angles, where b_3 and c_3 are given in Eq. (3).

In addition, if we know the absolute scale of the depth values, then can use the estimated pan and tilt angles to enforce consistency constraints on remaining parameters (effectively mapping the $\mathbb{R}^8$ parameter vector to SE (3)), thereby increasing the robustness of the tracking process. Alternatively, if the scale on depth is not known, the same consistency constraints be used to estimate the unknown scale parameter. This can be done continually as the object is being tracked, i.e. we can recover the object's 3D position and orientation at each instant in time. Likewise, the illumination coefficients each correspond to different light source directions, with the magnitudes of the coefficients proportional to the light source strengths.

We should add that, while the method in Murase and Nayar [18] could also determine pose, an advantage of this method is that for objects with similar geometry, but different albedo patterns, all that is required is that the illumination cone be transformed. There is no need to re-learn the pose manifold or illumination subspace. For the problem of face tracking, all that is required is that a canonical (or average) face geometry be known in advance. Thus, the tracking system described in this paper could work for any face without having to undergo a costly retraining procedure. In fact, the results in the next section use the geometry of the first author's face to track the face of the second author.

5. RESULTS

A tracking algorithm based on these ideas has been implemented in the XVision tracking environment [31]. In this section we first demonstrate the accuracy of the algorithm by presenting data from tracking a sphere for which pose is known. We then present three runs of the system applied to the problem of human face tracking to show its applicability to real world situations. We note that the pose and illumination models for the face have been described in Sections 3.1 and 3.2, respectively. All experiments were performed in real time on live video data on an SGI Indy equipped with a VINO digitising system.

5.1. Pose Determination for a Sphere

In this experiment, we chose to use a known object – a 'calibration sphere' – for which an exact warping model is known. The target was tracked while mounted on a pan-tilt head, allowing for exact control of the two angles of orientation of the sphere. Figure 3 shows an image of the target. Figure 4 contains two graphs showing the accuracy with which various trajectories were estimated. In all tests, the (imaged) diameter of the ball was 160 pixels. The size of the tracking window was 100×100 averaged to half resolution.

Several observations can be made from the graphs:

- The graph at the lower left shows the absolute pan angle and error in estimated pan angle over an interval from $-10°$ to $+10°$ from the starting position. In general, the error is less than a degree of angle except for a small spike near 0°. To place this in perspective, the range of depths subtended by the tracking window is about 45 pixels. It follows that an error of one pixel corresponds to $\sin^{-1}(1/45) = 1.27°$. We perform no smoothing on the incoming images and we are operating at half resolution, so the results shown here correspond to matching images to sub-pixel accuracy. The graph also clearly reflects aliasing effects that could possibly be alleviated through the use of smoothing.
- The graph at the right shows the results of a more

complex trajectory where estimated pan and tilt have been plotted against each other. Note, in particular, that the diagonal shows strong aliasing on the order of 1° to 1.5° – about what is expected in this case. The motion left and right shows a small amount of hysteresis, possibly indicating two nearby local minima, and both axes exhibit a small amount of systematic bias. Both errors are attributable to quantisation error.

Overall, these results seem to indicate that it is possible to obtain accurate estimates of object pose. It is interesting to note that tracking a sphere is in fact an extremely challenging problem because the motion templates for translation and the templates for rotation are quite similar. Objects with more variable surfaces usually lead to better conditioning of the estimation problem.

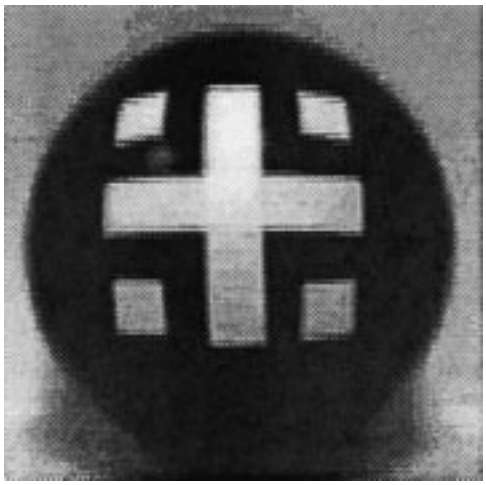

Fig. 3. Image of a calibration sphere.

5.2. Face Tracking

The second test object is an unadorned human face. In the first experimental run with a face, we emphasise the use of the pose manifold. Since ground-truth values for head pose are not available, we chose to demonstrate the accuracy of the pose estimates by using them to animate a range model of a human head. The results are shown in Fig. 5. The first row of images are frames taken from the live sequence; the second row are the corresponding frames from the animation. Note, in particular, that the algorithm computes accurate pose information even when significant areas of the face become occluded.

In the second run, we vary the illumination dramatically, but keep the pose relatively constant. In Fig. 6, we show three frames excerpted from the sequence and below them the corresponding artificial images synthesised using the lighting coefficients computed by the tracking algorithm.

In the third run, we vary both pose and illumination. The results appear in Fig. 7. The graph at the upper left shows the pose angles for the face, and the graph at the upper right shows the illumination coefficients for three of the rays defining the cone. The coefficients represent the contribution of illumination from the left, illumination from the right, and illumination from above. The lower row of images are frames from the sequence.

By comparing the graphs and the images, we see that the parameters vary in exact correspondence to the physical

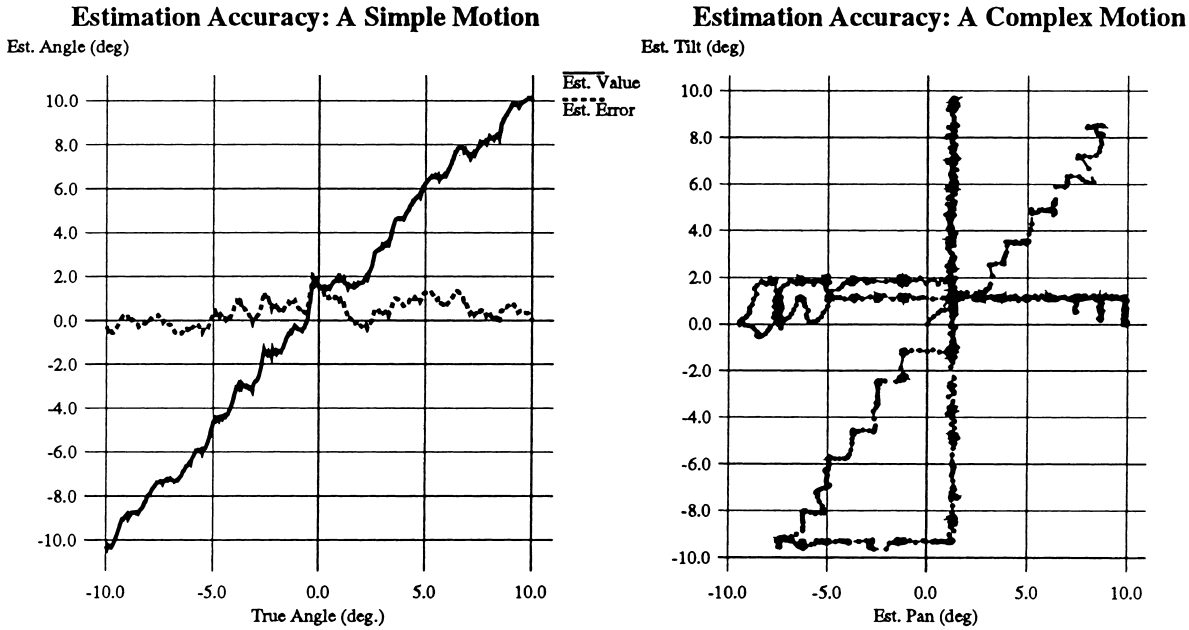

Fig. 4. Left, a graph of the estimated position with respect to ground truth, and the error between ground truth and the estimate. Right, the estimated pan and tilt angles for a more complex trajectory consisting of a motion 10° to the left, 20° to the right, 10° back to the left, 10° upward, 20° downward, 10° to the left, and diagonally upward 20° in both axes.

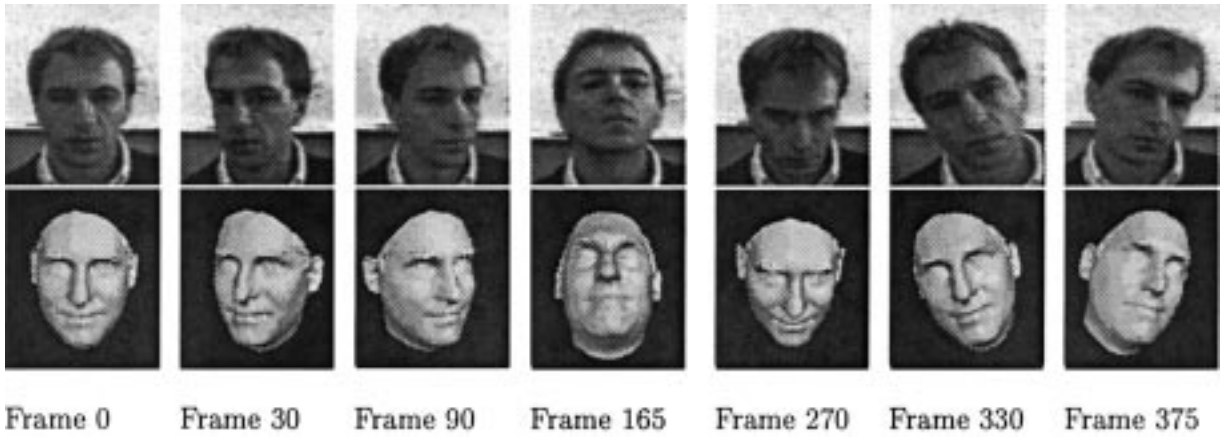

Fig. 5. The results of animating a head using the pose coefficients of the tracking algorithm. Above, the live images, and below an image of the range model of a head rotated by the angles computed by the tracking algorithm.

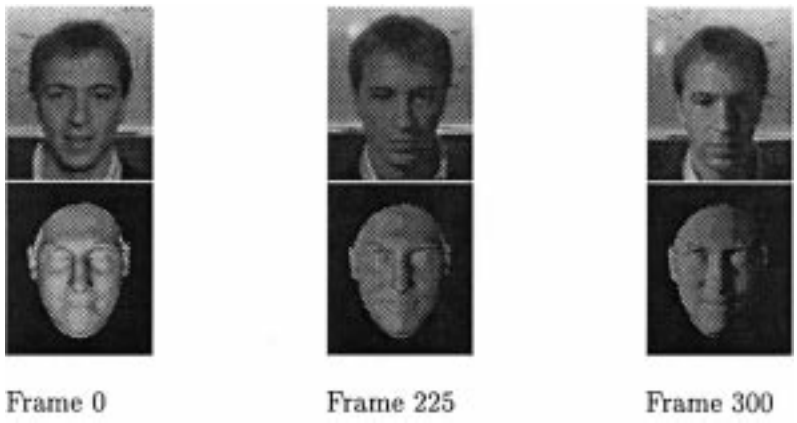

Fig. 6. The results of animating a head using the illumination coefficients computed by the tracking algorithm. Above, three frames excerpted from the sequence and below the corresponding artificial images synthesized using the lighting coefficients computed by the tracking algorithm.

situation. At the start (Frame 0), the face is illuminated from the front and is facing forward. As the face tips downward (Frame 75), the ray describing illumination from above and the tilt angle increase. Next, as the source moves to the right, the contribution of the corresponding illumination ray increases (Frame 165) until the face is turned toward the light (Frame 225). The face turns back (Frame 300), then the source is moved to the left (Frame 375) causing the contribution of the corresponding illumination coefficient to increase. Finally, the face turns to the left (Frame 420) causing it to be more centrally illuminated, at which point the contribution of illumination from the left decreases.

6. CONCLUSION

We have developed a method for tracking objects *and* computing their pose and illumination in a computationally efficient manner. To do this, we build an image-based model of the target object which can predict the target appearance over a large range of variation in pose and illumination. The models are built from a small number of images over a small range of object variation. Yet the method is able to extrapolate to more extreme conditions.

At the moment, the pose manifolds and illumination cone are built before the tracking process is initiated. However, we believe it may be possible to estimate one or both during the tracking procedure. All that is needed to build the pose manifolds are images of the target seen under several rotations – rotations small enough for a tracker using only affine deformations to keep a lock on the target. Likewise, to determine the illumination cone, the tracking system must be sufficiently robust to tolerate novel deviations in illumination long enough to incorporate these deviations into the cone representation.

In future work, we plan to investigate the online development of pose and illumination models, and also to extend the algorithm to handle multiple views. The principle problem in multiple-view tracking is to develop effective methods for indexing the views and moving smoothly (in parameter space) between them as the target itself moves. Initial investigations in this direction suggest that the use of differential methods (e.g. methods similar to those of Stein and

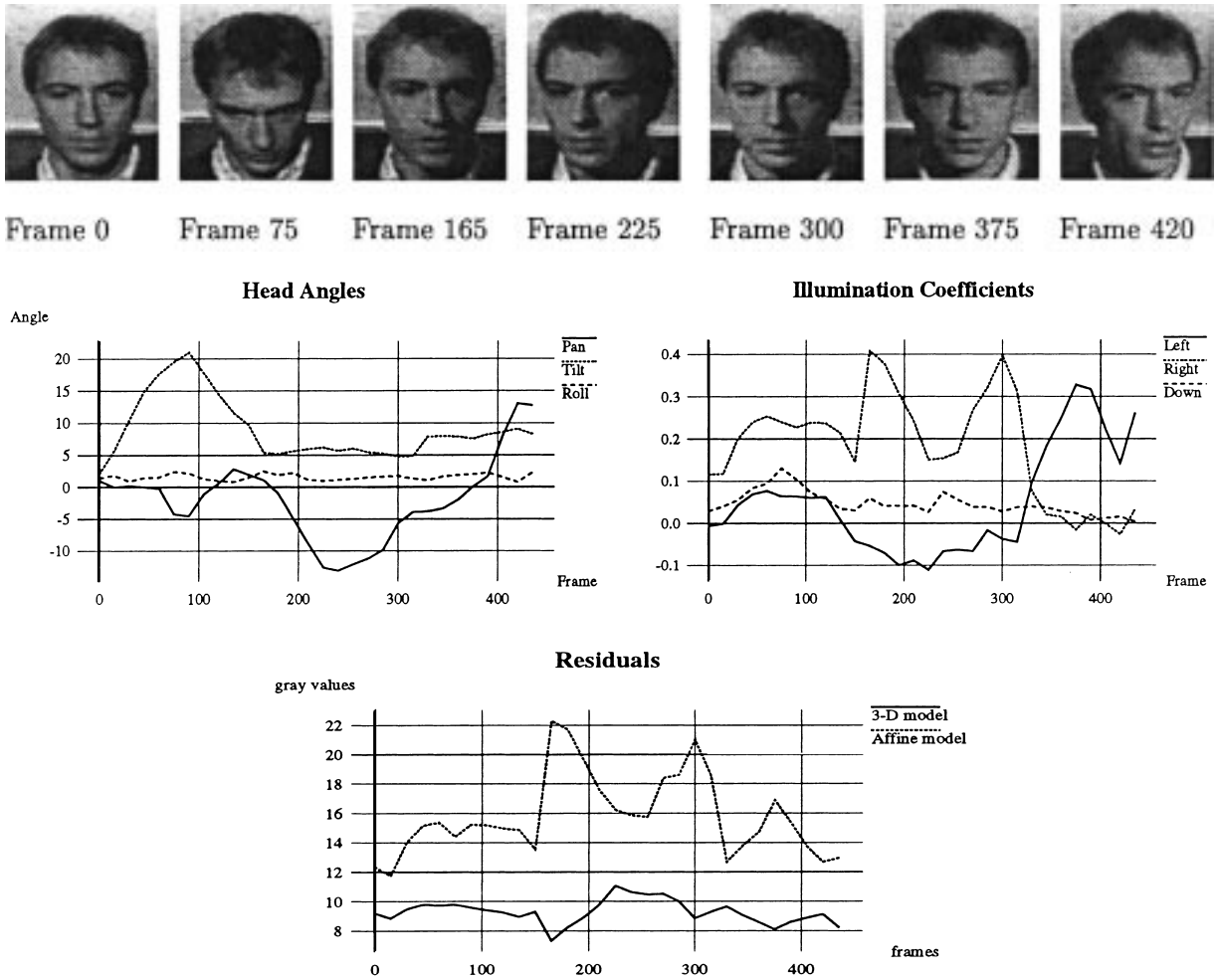

Fig. 7. The top row of images are selected frames from a tracking sequence. The next two graphs show the computed head angles and illumination coefficients for this sequence. The graph at the bottom compares the least square residuals for an affine model against the 3D pose-based model.

Shashua [32]) for computing pose variation work well for moderately dense (in pose space) images of the target. Furthermore, this approach appears to be compatible with the class of tracking algorithms we currently use, and is of similar computational complexity.

Successful incorporation of view switching will make it possible to track objects through the complete pose space. By successfully learning views online (each with its own pose and illumination model), we hope to demonstrate completely automated tracking of novel objects in real time.

†G. Hager was supported by Army DURIP grant DAAH04-95-1-0058, by National Science Foundation grant IRI-9420982, and by funds provided by Yale University.

Acknowledgements

P.N. Belhumeur was supported by ARO grant DAAH04-95-1-0494 and a Presidential Early Career Award for Scientists and Engineers IRI-9703134.

References

1. Ullman S, Basri R. Recognition by linear combinations of models. IEEE Trans Pattern Analysis and Machine Intelligence 1991;13(10):992–1006
2. Jacobs DW. Space efficient 3D model indexing. Proceedings IEEE Conference Computer Vision Pattern Recognition 1992; 439–444
3. Tomasi C. Kanade T. Shape and motion from image streams under orthography: A factorization method. International Journal of Computer Vision 1992;9(2):137–154
4. Hager GD, Belhumeur PN. Efficient region tracking of with parametric models of illumination and geometry. IEEE Transactions Pattern Analysis Machine Intelligence 1998;20(10):1025–1039
5. Black MJ, Jepson AD. Eigentracking: Robust matching and tracking of articulated objects using a view-based representation. International Journal of Computer Vision 1998;26(1):63–84
6. Shi J, Tomasi C. Good features to track. Proc IEEE Computer Society Conference Computer Vision and Pattern Recognition 1994;593–600
7. Belhumeur PN, Kriegman DJ. What is the set of images of an

object under all possible illumination conditions. International Journal of Computer Vision 1998;28(3)

8. Gennery DB. Visual tracking of known three-dimensional objects. International Journal of Computer Vision 1992;7(3):243–270
9. Lowe DG. Robust model-based motion tracking through the integration of search and estimation. International Journal of Computer Vision 1992;8(2):113–122
10. Gavrila DM, Davis LS. Tracking humans in action: A 3D model-based approach. Proc DARPA Image Understanding Workshop 1996, pp 737–746
11. Isard M, Blake A. Contour tracking by stochastic propagation of conditional density. Proceedings European Conference on Computer Vision 1996, pp 343–356
12. Blake A, Isard M. Active Contours. Springer-Verlag, 1998
13. Reynard D, Wildenberg A, Blake A, Marchant J. Learning dynamics of complex motions from image sequences. Proceedings European Conference on Computer Vision 1996, pp 357–368
14. Deriche R, Faugeras O. Tracking line segments. In: O Faugeras (ed), Proceedings European Conference on Computer Vision, Springer-Verlag, 1990, pp 259–268
15. Papanikolopoulos NP, Khosla PK. Adaptive robot visual tracking: theory and experiments. IEEE Trans Automatic Control 1993;38(3):429–445
16. Black MJ, Yacoob Y. Tracking and recognizing rigid and non-rigid facial motions using local parametric models of image motion. International Journal of Computer Vision 1997;25(1):23–48
17. Rehg JM, Witkin AP. Visual tracking with deformation models. Proc IEEE International Conference Robot Automat 1991, pp 844–850
18. Murase H, Nayar S. Visual learning and recognition of 3-D objects from appearance. International Journal of Computer Vision 1995;14(5–24)
19. Pentland A, Moghaddam B, Starner. View-based and modular eigenspaces for face recognition. Proceedings IEEE Conference Computer Vision Pattern Recognition 1994, pp 84–91
20. Poggio T, Sung KK. Example-based learning for view-based human face detection. Proc Image Understanding Workshop 1994;II:843–850
21. Sirovitch L, Kirby M. Low-dimensional procedure for the characterization of human faces. Journal of Optical Society America A 1987;2:519–524
22. Turk M, Pentland A. Eigenfaces for recognition. Journal of Cognitive Neuroscience 1991;3(1)
23. Goresky M, Macpherson R. Stratified Morse Theory. Springer-Verlag, 1980
24. Lucas BD, Kanade T. An iterative image registration technique with an application to stereo vision. Proceedings International Joint Conference on Artificial Intelligence 1981, pp 674–679
25. Koenderink JJ, Van Doorn AJ. Affine structure from motion. Journal of Optical Society Am 1991;8(2):337–385
26. Belhumeur PN. A Bayesian approach to binocular stereopsis. International Journal of Computer Vision 1996;19
27. Hallinan P. A Deformable Model for Face Recognition Under Arbitrary Lighting Conditions. PhD thesis, Harvard University, 1995
28. Moses Y, Adini Y, Ullman S. Face recognition: The problem of compensating for changes in illumination direction. European Conference Computer Vision 1994, pp 286–296
29. Shashua A. Geometry and Photometry in 3D Visual Recognition. PhD thesis, MIT, 1992
30. Paul R. Robot Manipulators. MIT Press, 1981
31. Hager GD, Toyama K. The "XVision" system: A general purpose substrate for real-time vision applications. Computer Vision, Image Understanding 1998;69(1):23–27
32. Stein GP, Shashua A. Direct estimation of motion and extended scene structure from a moving stereo rig. Proceedings IEEE Conference Computer Vision Pattern Recognition 1998, pp 211–218

Gregory D. Hager received a BA degree in computer science and mathematics from Luther College in 1983, and his MS and PhD in computer science from the university of Pennsylvania in 1985 and 1988, respectively. From 1988 to 1990 he was a Fulbright junior research fellow at the University of Karlsruhe and the Fraunhofer Institute IITB in Karlsruhe Germany. Upon returning to the states he joined the computer science department at Yale University where he is currently an associate professor. He is a member of IEEE and AAAI, and is currently co-chairman of the Robotics and Automation Society Technical Committee on Computer and Robot Vision. His research interests include visual tracking, hand-eye coordination, sensor data fusion and sensor planning. A book on his dissertation work entitled '*Task-Directed Sensor Fusion and Planning*' is published by Kluwer Academic Publishers, Inc.

Peter N. Belhumeur graduated in 1985 from Brown University with Highest Honors, receiving a ScB. degree in Computer and Information Engineering. He received an SM in 1991 and a PhD in 1993 from Harvard University where he studied under a Harvard Fellowship. In 1993 he was a Postdoctoral Fellow at the University of Cambridge's Sir Isaac Newton Institute for Mathematical Sciences. He was appointed Assistant Professor of Electrical Engineering at Yale University in 1994, and was given a joint appointment with the Department of Computer Science in 1998. He is a recipient of the Presidential Early Career Award for Scientists and Engineers, the National Science Foundation Career Award, and the Yale University Junior Faculty Fellowship for the Natural Sciences. He won the 'Best Paper Award' at the *1996 IEEE Conference on Computer Vision and Pattern Recognition*.

Correspondence and offprint requests to: P. N. Belhumeur, Department of Electrical Engineering, Center for Computational Vision and Control, Yale University, New Haven, CT 06520-8267, USA, Fax: (203) 432 7481.