EigenTracking: Robust Matching and Tracking of Articulated Objects Using a View-Based Representation

Michael J. Black[1] and Allan D. Jepson[2]

[1] Xerox Palo Alto Research Center, 3333 Coyote Hill Road, Palo Alto, CA 94304
black@parc.xerox.com

[2] Department of Computer Science, University of Toronto, Ontario M5S 3H5
and Canadian Institute for Advanced Research
jepson@vis.toronto.edu

Abstract. This paper describes a new approach for tracking rigid and articulated objects using a view-based representation. The approach builds on and extends work on eigenspace representations, robust estimation techniques, and parameterized optical flow estimation. First, we note that the least-squares image reconstruction of standard eigenspace techniques has a number of problems and we reformulate the reconstruction problem as one of robust estimation. Second we define a "subspace constancy assumption" that allows us to exploit techniques for parameterized optical flow estimation to simultaneously solve for the view of an object and the affine transformation between the eigenspace and the image. To account for large affine transformations between the eigenspace and the image we define an EigenPyramid representation and a coarse-to-fine matching strategy. Finally, we use these techniques to track objects over long image sequences in which the objects simultaneously undergo both affine image motions and changes of view. In particular we use this "EigenTracking" technique to track and recognize the gestures of a moving hand.

1 Introduction

View-based object representations have found a number of expressions in the computer vision literature, in particular in the work on eigenspace representations [10, 13]. Eigenspace representations provide a compact approximate encoding of a large set of training images in terms of a small number of orthogonal basis images. These basis images span a subspace of the training set called the eigenspace and a linear combination of these images can be used to approximately reconstruct any of the training images. Previous work on eigenspace representations has focused on the problem of object recognition and has only peripherally addressed the problem of tracking objects over time. Additionally, these eigenspace reconstruction methods are not invariant to image transformations such as translation, scaling, and rotation. Previous approaches have typically assumed that the object of interest can be located in the scene, segmented, and transformed into a canonical form for matching with the eigenspace. In this paper we will present a robust statistical framework for reconstruction using the eigenspace that will generalize and extend the previous work in the area to ameliorate some of these problems. The work combines lines of research from object recognition using eigenspaces, parameterized optical flow models, and robust estimation techniques into a novel method for tracking objects using a view-based representation.

There are two primary observations underlying this work. First, standard eigenspace techniques rely on a least-squares fit between an image and the eigenspace [10] and this can lead to poor results when there is structured noise in the input image. We reformulate the eigenspace matching problem as one of robust estimation and show how it overcomes the problems of the least-squares approach. Second, we observe that rather than try to represent all possible views of an object from all possible viewing positions, it is more practical to represent a smaller set of canonical views and allow a parameterized transformation (eg. affine) between an input image and the eigenspace. This allows a *multiple-views plus transformation* [12] model of object recognition. What this implies is that matching using an eigenspace representation involves both estimating the view as well as the transformation that takes this view into the image. We formulate this problem in a robust estimation framework and simultaneously solve for the view and the transformation. For a particular view of an object we define a *subspace constancy assumption* between the eigenspace and the image. This is analogous to the "brightness constancy assumption" used in optical flow estimation and it allows us to exploit parameterized optical flow techniques to recover the transformation between the eigenspace and the image. Recovering the view and transformation requires solving a non-linear optimization problem which we minimize using gradient descent with a continuation method. To account for large transformations between model and image we define an EigenPyramid representation and a coarse-to-fine matching scheme. This method enables the tracking of previously viewed objects undergoing general motion with respect to the camera. This approach, which we call *EigenTracking*, can be applied to both rigid and articulated objects and can be used for object and gesture recognition in video sequences.

2 Related Work

While eigenspaces are one promising candidate for a view-based object representation, there are still a number of technical problems that need to be solved before these techniques can be widely applied. First, the object must be located in the image. It is either assumed that the object can be detected by a simple process [9, 10] or through global search [9, 13]. Second, the object must be segmented from the background so that the reconstruction and recognition is based on the object and not the appearance of the background. Third, the input image must be be transformed (for example by translation, rotation, and scaling) into some canonical form for matching. The robust formulation and continuous optimization framework presented here provide a local search method that is robust to background variation and simultaneously matches the eigenspace and image while solving for translation, rotation, and scale.

To recognize objects in novel views, traditional eigenspace methods build an eigenspace from a dense sampling of views [6, 7, 10]. The eigenspace coefficients of these views are used to define a surface in the space of coefficients which interpolates between views. The coefficients of novel views will hopefully lie on this surface. In our approach we represent views from only a few orientations and recognize objects in other orientations by recovering a parameterized transformation (or warp) between the image and the eigenspace. This is consistent with a model of human object recognition that suggests that objects are represented by a set of views corresponding to familiar orientations and that new views are transformed to one of these stored views for recognition [12].

To track objects over time, current approaches assume that simple motion detection and tracking approaches can be used to locate objects and then the eigenspace matching verifies the object identity [10, 13]. What these previous approaches have failed to exploit is that the eigenspace itself provides a representation (i.e. an image) of the object that can be used for tracking. We exploit our robust parameterized matching scheme to perform tracking of objects undergoing affine image distortions and changes of view.

This differs from traditional image-based motion and tracking techniques which typically fail in situations in which the viewpoint of the object changes over time. It also differs from tracking schemes using 3D models which work well for tracking simple rigid objects. The EigenTracking approach encodes the appearance of the object from multiple views rather than its structure.

Image-based tracking schemes that emphasize learning of views or motion have focused on region contours [1, 5]. In particular, Baumberg and Hogg [1] track articulated objects by fitting a spline to the silhouette of an object. They learn a view-based representation of people walking by computing an eigenspace representation of the knot points of the spline over many training images. Our work differs in that we use the brightness values within an image region rather than the region outline and we allow parameterized transformations of the input data in place of the standard preprocessing normalization.

3 Eigenspace Approaches

Given a set of images, eigenspace approaches construct a small set of basis images that characterize the majority of the variation in the training set and can be used to approximate any of the training images. For each $n \times m$ image in a training set of p images we construct a 1D column vector by scanning the image in the standard lexicographic order. Each of these 1D vectors becomes a column in a $nm \times p$ matrix A. We assume that the number of training images, p, is less than the number of pixels, nm and we use Singular Value Decomposition (SVD)[3] to decompose the matrix A as

$$A = U \Sigma V^T . \tag{1}$$

U is an orthogonal matrix of the same size as A representing the principle component directions in the training set. Σ is a diagonal matrix with singular values $\sigma_1, \sigma_2, \ldots, \sigma_p$ sorted in decreasing order along the diagonal. The $p \times p$ orthogonal matrix V^T encodes the coefficients to be used in expanding each column of A in terms of the principle component directions.

If the singular values σ_k, for $k \geq t$ for some t, are small then, since the columns of U are orthonormal, we can approximate some new column $\mathbf{e}$ as

$$\mathbf{e}^* = \sum_{i=1}^{t} c_i U_i , \tag{2}$$

[3] Other approaches have been described in the literature (cf. [10]).

Sample images from the training set:

First few principle components:

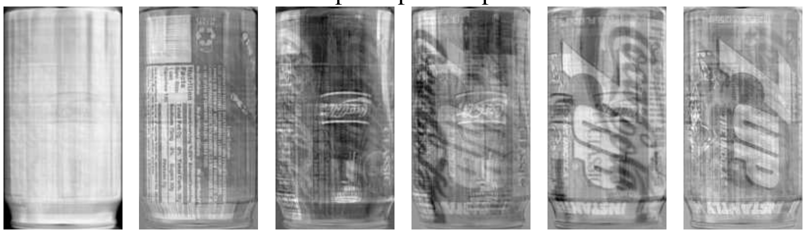


Fig. 1. Example that will be used to illustrate ideas throughout the paper.

where the c_i are scalar values that can be computed by taking the dot product of **e** and the column U_i. This amounts to a projection of the input image, e, onto the subspace defined by the t basis vectors.

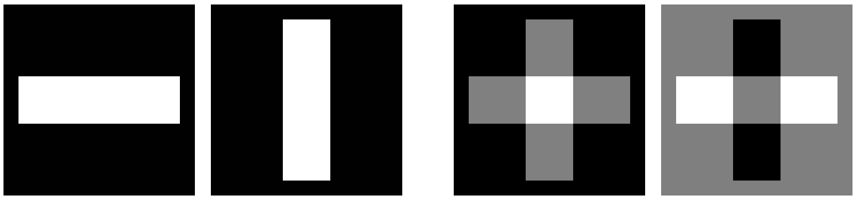
For illustration we constructed an eigenspace representation for soda cans. Figure 1 (top row) shows some example soda can images in the training set which contained 200 images of Coke and 7UP cans viewed from the side. The eigenspace was constructed as described above and the first few principle components are shown in the bottom row of Figure 1.[4] For the experiments in the remainder of the paper, 50 principle components were used for reconstruction. While fewer components could be used for recognition, EigenTracking will require a more accurate reconstruction.

4 Robust Matching

The approximation of an image region by a linear combination of basis vectors can be thought of as "matching" between the eigenspace and the image. This section describes how this matching process can be made robust.

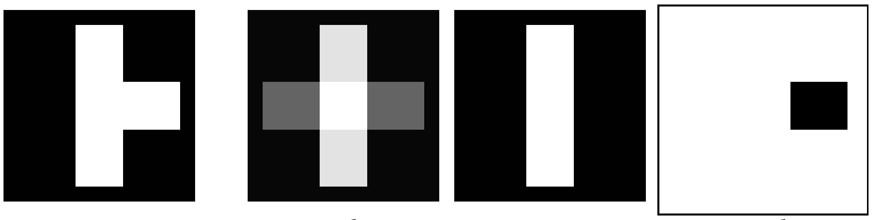
Let **e** be an input image region, written as a $nm \times 1$ vector, that we wish to match to the eigenspace. For the standard approximation $\mathbf{e}^*$ of **e** in Equation (2), the coefficients c_i are computed by taking the dot product of **e** with the U_i. This approximation corresponds to the least-squares estimate of the c_i [10]. In other words, the c_i are those that give a reconstructed image that minimizes the squared error $E(\mathbf{c})$ between **e** and $\mathbf{e}^*$ summed

[4] In this example we did not subtract the mean image from the training images before computing the eigenspace. The mean image corresponds to the first principle component resulting in one extra eigenimage.



a *b* *c* *d*

Fig. 2. A simple example. *(a, b):* Training images. *(c, d):* Eigenspace basis images.



a *b* *c* *d*

Fig. 3. Reconstruction. *(a):* New test image. *(b):* Least-squares reconstruction. *(c):* Robust reconstruction. *(d):* Outliers (shown as black pixels).

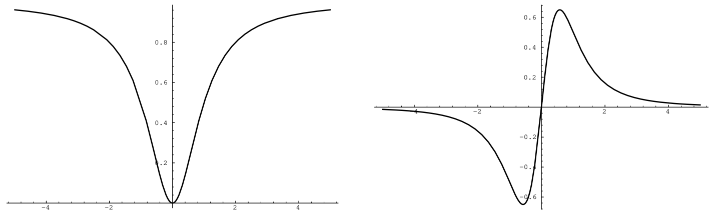
over the entire image:

$$E(\mathbf{c}) = \sum_{j=1}^{n \times m} (\mathbf{e}_j - \mathbf{e}_j^*)^2 = \sum_{j=1}^{n \times m} \left(\mathbf{e}_j - \left(\sum_{i=1}^{t} c_i U_{i,j} \right) \right)^2 . \tag{3}$$

This least-squares approximation works well when the input images have clearly segmented objects that look roughly like those used to build the eigenspace. But it is commonly known that least-squares is sensitive to gross errors, or "outliers" [8], and it is easy to construct situations in which the standard eigenspace reconstruction is a poor approximation to the input data. In particular, if the input image contains structured noise (eg. from the background) that can be represented by the eigenspace then there may be multiple possible matches between the image and the eigenspace and the least-squares solution will return some combination of these views.

For example consider the very simple training set in Figure 2 (*a* and *b*). The basis vectors in the eigenspace are shown in Figure 2 (*c, d*).[5] Now, consider the test image in Figure 3*a* which does not look the same as either of the training images. The least-squares reconstruction shown in Figure 3*b* attempts to account for all the data but this cannot be done using a linear combination of the basis images. The robust formulation described below recovers the dominant feature which is the vertical bar (Figure 3*c*) and to do so, treats the data to the right as outliers (black region in Figure 3*d*).

[5] We subtracted the mean from each of Figure 2 *a* and *b* and included the constant image in the expansion basis.



$\rho(x, \sigma)$ $\psi(x, \sigma)$

Fig. 4. Robust error norm (ρ) and its derivative (ψ).

To robustly estimate the coefficients **c** we replace the quadratic error norm in Equation (3) with a robust error norm, ρ, and minimize

$$E(\mathbf{c}) = \sum_{j=1}^{n \times m} \rho \left(\left(\mathbf{e}_j - \left(\sum_{i=1}^{t} c_i U_{i,j} \right) \right), \sigma \right). \tag{4}$$

where σ is a scale parameter. For the experiments in this paper we take ρ to be

$$\rho(x, \sigma) = \frac{x^2}{\sigma + x^2}, \qquad \frac{\partial}{\partial x}\rho(x, \sigma) = \psi(x, \sigma) = \frac{2x\sigma^2}{(\sigma^2 + x^2)^2}, \tag{5}$$

which is a robust error norm that has been used extensively for optical flow estimation [3, 4]. The shape of the function, as shown in Figure 4, is such that it “rejects”, or downweights, large residual errors. The function $\psi(x, \sigma)$, also shown in Figure 4, is the derivative of ρ and characterizes the influence of the residuals. As the magnitudes of residuals $(\mathbf{e}_j - \mathbf{e}_j^*)$ grow beyond a point their influence on the solution begins to decrease and the value of $\rho(\cdot)$ approaches a constant.

The value σ is a scale parameter that affects the point at which the influence of outliers begins to decrease. By examining the ψ-function we see that this “outlier rejection” begins where the second derivative of ρ is zero. For the error norm used here, this means that those residuals where $|(\mathbf{e}_j - \mathbf{e}_j^*)| > \sigma/\sqrt{3}$ can be viewed as outliers.

The computation of the coefficients **c** involves the minimization of the non-linear function in Equation (4). We perform this minimization using a simple gradient descent scheme with a continuation method that begins with a high value for σ and lowers it during the minimization (see [2, 3, 4] for details). The effect of this procedure is that initially no data are rejected as outliers then gradually the influence of outliers is reduced. In our experiments we have observed that the robust estimates can tolerate roughly $35 - 45\%$ of the data being outliers.

4.1 Outliers and Multiple Matches

As we saw in Figure 3 it is possible for an input image to contain a brightness pattern that is not well represented by any single “view”. Given a robust match that recovers the

Image Least Squares Robust 1 Outliers 1 Robust 2 Outliers 2

Fig. 5. Robust matching with structured noise.

"dominant" structure in the input image, we can detect those points that were treated as outliers. We define an outlier vector, or "mask", $\mathbf{m}$ as

$$m_j = \begin{cases} 0 & |(\mathbf{e}_j - \mathbf{e}_j^*)| \leq \sigma/\sqrt{3} \\ 1 & \text{otherwise.} \end{cases}$$

If a robust match results in a significant number of outliers, then additional matches can be found by minimizing

$$E(\mathbf{c}) = \sum_{j=1}^{n\times m} m_j \; \rho\left(\left(\mathbf{e}_j - \left(\sum_{i=1}^{t} c_i U_{i,j}\right)\right), \sigma\right). \tag{6}$$

4.2 Robust Matching Examples

An example will help illustrate the problems with the least-squares solution and the effect of robust estimation. Figure 5 shows an artificial image constructed from two images that were present in the training data (the image is $2/3$ Coke can and $1/3$ 7UP can). It is impossible to reconstruct the entire input image accurately with the eigenspace despite the fact that both parts of the image can be represented independently. The least-squares solution recovers a single view that contains elements of both possible views. The robust estimation of the linear coefficients results in a much more accurate reconstruction of the dominant view (Figure 5, Robust 1). Moreover, we can detect those points in the image that did not match the reconstruction very well and were treated as outliers (black points in Figure 5, Outliers 1) Equation (6) can be used to recover the view corresponding to the outliers (Figure 5, Robust 2) and even with very little data, the reconstructed image reasonably approximates the view of the 7UP can.

5 EigenSpaces and Parametric Transformations

The previous section showed how robust estimation can improve the reconstruction of an image that is already aligned with the eigenspace. In this section we consider how to achieve this alignment in the first place. It is impractical to represent all possible views of an object at all possible scales and all orientations. One must be able to recognize a familiar object in a previously unseen pose and hence we would like to represent a small set of views and recover a transformation that maps an image into the eigenspace. In the previous section we formulated the matching problem as an explicit non-linear parameter estimation problem. In this section we will simply extend this problem formulation with the addition of a few more parameters representing the transformation between the image and the eigenspace.

To extend eigenspace methods to allow matching under some parametric transformation we to formalize a notion of "brightness constancy" between the eigenspace and the image. This is a generalization of the notion of brightness constancy used in optical flow which states that the brightness of a pixel remains constant between frames but that its location may change. For eigenspaces we wish to say that there is a view of the object, as represented by some linear combination of the basis vectors, U_i, such that pixels in the reconstruction are the same brightness as pixels in the image given the appropriate transformation. We call this the *subspace constancy assumption*.

Let $U = [U_1, U_2, \ldots U_t]$, $\mathbf{c} = [c_1, c_2, \ldots c_t]^T$, and

$$U\mathbf{c} = \sum_{i=1}^{t} c_i U_i, \tag{7}$$

where $U\mathbf{c}$ is the approximated image for a particular set of coefficients, $\mathbf{c}$. While $U\mathbf{c}$ is a $nm \times 1$ vector we can index into it as though it were an $n \times m$ image. We define $(U\mathbf{c})(\mathbf{x})$ to be the value of $U\mathbf{c}$ at the position associated with pixel location $\mathbf{x} = (x, y)$.

Then the robust matching problem from the previous section can be written as

$$E(\mathbf{c}) = \sum_{\mathbf{x}} \rho(I(\mathbf{x}) - (U\mathbf{c})(\mathbf{x}), \sigma), \tag{8}$$

where I is an $n \times m$ sub-image of some larger image. Pentland *et al.* [11] call the residual error $I - U\mathbf{c}$ the distance-from-feature-space (DFFS) and note that this error could be used for localization and detection by performing a global search in an image for the best matching sub-image. Moghaddam and Pentland extend this to search over scale by constructing multiple input images at various scales and searching over all of them simultaneously [9]. We take a different approach in the spirit of parameterized optical flow estimation. First we define the subspace constancy assumption by parameterizing the input image as follows

$$I(\mathbf{x} + \mathbf{u}(\mathbf{x}, \mathbf{a})) = (U\mathbf{c})(\mathbf{x}), \quad \forall \mathbf{x}, \tag{9}$$

where $\mathbf{u}(\mathbf{x}, \mathbf{a}) = (u(\mathbf{x}, \mathbf{a}), v(\mathbf{x}, \mathbf{a}))$ represents an image transformation (or motion), u and v represent the horizontal and vertical displacements at a pixel, and the parameters $\mathbf{a}$ are to be estimated. For example we may take $\mathbf{u}$ to be the affine transformation

$$u(\mathbf{x}, \mathbf{a}) = a_0 + a_1 x + a_2 y$$
$$v(\mathbf{x}, \mathbf{a}) = a_3 + a_4 x + a_5 y$$

where x and y are defined with respect to the image center. Equation (9) states that there should be some transformation, $\mathbf{u}(\mathbf{x}, \mathbf{a})$, that, when applied to image region I, makes I

look like some image reconstructed using the eigenspace. This transformation can be thought of as *warping* the input image into the coordinate frame of the training data.

Our goal is then to simultaneously find the $\mathbf{c}$ and $\mathbf{a}$ that minimize

$$E(\mathbf{c}, \mathbf{a}) = \sum_{\mathbf{x}} \rho(I(\mathbf{x} + \mathbf{u}(\mathbf{x}, \mathbf{a})) - (U\mathbf{c})(\mathbf{x}), \sigma). \tag{10}$$

As opposed to the exhaustive search techniques used by previous approaches [9, 13], we derive and solve a continuous optimization problem.

First we rewrite the left hand side of Equation (9) using a first order Taylor series expansion

$$I(\mathbf{x}) + I_x(\mathbf{x})u(\mathbf{x}, \mathbf{a}) + I_y(\mathbf{x})v(\mathbf{x}, \mathbf{a}) = (U\mathbf{c})(\mathbf{x})$$

where I_x and I_y are partial derivatives of the image in the x and y directions respectively. Reorganizing terms gives

$$I_x(\mathbf{x})u(\mathbf{x}, \mathbf{a}) + I_y(\mathbf{x})v(\mathbf{x}, \mathbf{a}) + (I(\mathbf{x}) - (U\mathbf{c})(\mathbf{x})) = 0. \tag{11}$$

This is very similar to the standard optical flow constraint equation where the $U\mathbf{c}$ has replaced $I(\mathbf{x}, t-1)$ and $(I - U\mathbf{c})$ takes the place of the "temporal derivative".

To recover the coefficients of the reconstruction as well as the transformation we combine the constraints over the entire image region and minimize

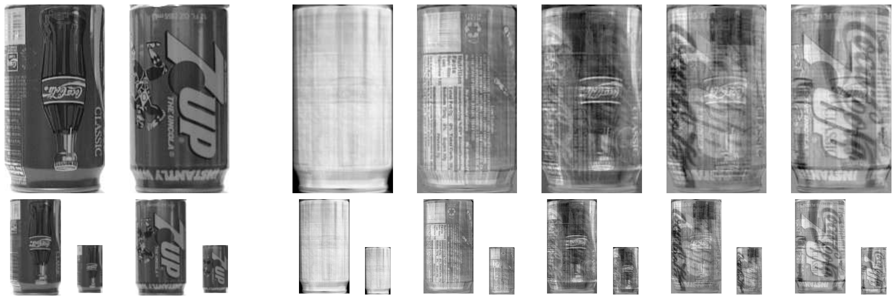
$$E(\mathbf{c}, \mathbf{a}) = \sum_{\mathbf{x}} \rho(I_x(\mathbf{x})u(\mathbf{x}, \mathbf{a}) + I_y(\mathbf{x})v(\mathbf{x}, \mathbf{a}) + (I(\mathbf{x}) - (U\mathbf{c})(\mathbf{x})), \sigma) \tag{12}$$

with respect to $\mathbf{c}$ and $\mathbf{a}$. As in the previous section, this minimization is performed using a simple gradient descent scheme with a continuation method that gradually lowers σ. As better estimates of $\mathbf{a}$ are available, the input image is warped by the transformation $\mathbf{u}(\mathbf{x}, \mathbf{a})$ and this warped image is used in the optimization. As this warping registers the image and the eigenspace, the approximation $U\mathbf{c}$ gets better and better. This minimization and warping continues until convergence. The entire non-linear optimization scheme is described in greater detail in [2].

Note that this optimization scheme will not perform a global search to "find" the image region that matches the stored representation. Rather, given an initial guess, it will refine the pose and reconstruction. While the initial guess can be fairly coarse as described below, the approach described here does not obviate the need for global search techniques but rather compliments them. In particular, the method will be useful for tracking an object where a reasonable initial guess is typically available.

EigenPyramids. As in the case of optical flow, the constraint equation (11) is only valid for small transformations. The recovery of transformations that result in large pixel differences necessitates a coarse-to-fine strategy. For every image in the training set we construct a pyramid of images by spatial filtering and sub-sampling (Figure 6). The images at each level in the pyramid form distinct training sets and at each level SVD is used to construct an eigenspace description of that level.

The input image is similarly smoothed and subsampled. The coarse-level input image is then matched against the coarse-level eigenspace and the values of $\mathbf{c}$ and $\mathbf{a}$ are



a *b*

Fig. 6. Example of EigenPyramids. *a:* Sample images from the training set. *b:* First few principle components in the EigenPyramid.

estimated at this level. The new values of **a** are then projected to the next level (in the case of the affine transformation the values of a_0 and a_3 are multiplied by 2). This **a** is then used to warp the input image towards the eigenspace and the value of **c** is estimated and the a_i are refined. The process continues to the finest level.

6 EigenTracking

The robust parameterized matching scheme described in the previous section can be used to track objects undergoing changes in viewpoint or changes in structure. As an object moves and the view of the object changes, we recover both the current view of the object and the transformation between the current view and the image. It is important to note that no "image motion" is being used to "track" the objects in this section. The tracking is achieved entirely by the parameterized matching between the eigenspace and the image. We call this *EigenTracking* to emphasize that a view-based representation is being used to track an object over time.

For the experiments here a three-level pyramid was used and the value of σ started at $65\sqrt{3}$ and was lowered to a minimum of $15\sqrt{3}$ by a factor of 0.85 at each of 15 stages. The values of **c** and **a** were updated using 15 iterations of the descent scheme at each stage, and each pyramid level. The minimization was terminated if a convergence criterion was met. The algorithm was given a rough initial guess of the transformation between the first image and the eigenspace. From then on the algorithm automatically tracked the object by estimating **c** and **a** for each frame. No prediction scheme was used and the motion ranged from 0 to about 4 pixels per frame. For these experiments we restricted the transformation to translation, rotation, and scale.

6.1 Pickup Sequence

First we consider a simple example in which a hand picks up a soda can. The can undergoes translation and rotation in the image plane (Figure 7). The region corresponding to the eigenspace is displayed as white box in the image. This box is generated by projecting the region corresponding to the eigenspace onto the image using the inverse of the

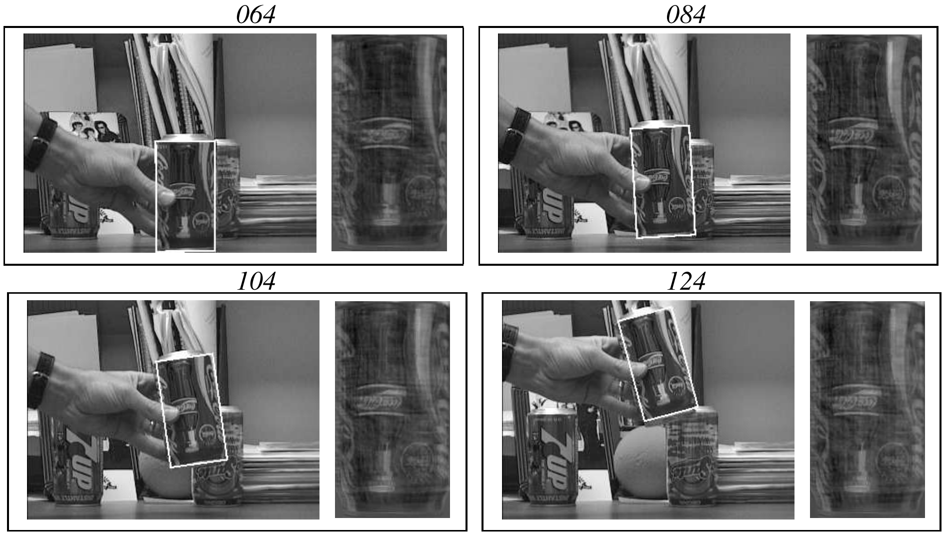


Fig. 7. Pickup Sequence. EigenTracking with translation and rotation in the image plane. Every 20 frames in the 75 frame sequence are shown.

estimated transformation between the image and the eigenspace. This projection serves to illustrate the accuracy of the recovered transformation. Beside each image is shown the robust reconstruction of the image region within the box.

6.2 Tracking a Rotating Object

Figure 8 shows the tracking of a soda can that translates left and right while moving in depth over 200 frames. While the can is changing position relative to the camera it is also undergoing rotations about its major axis. What this means is that the traditional brightness constancy assumption of optical flow will not track the "can" but rather the "texture" on the can. The subspace constancy assumption, on the other hand, means that we will recover the transformation between our eigenspace representation of the can and the image. Hence, it is the "can" that is tracked rather than "texture".

More details are provided to the right of the images. On the left of each box is the "stabilized" image which shows how the original image is "warped" into the coordinate frame of the eigenspace. Notice that the background differs over time as does the view of the can, but that the can itself is in the same position and at the same scale. The middle image in each box is the robust reconstruction of the image region being tracked. On the right of each box (in black) are the "outliers" where the observed image and the reconstruction differed by more than $\sigma/\sqrt{3}$.

6.3 Articulated Motion and Gesture Recognition

A final example considers the problem of recognizing hand gestures in video sequences in which the hand is moving. We define a simple set of four hand gestures illustrated in

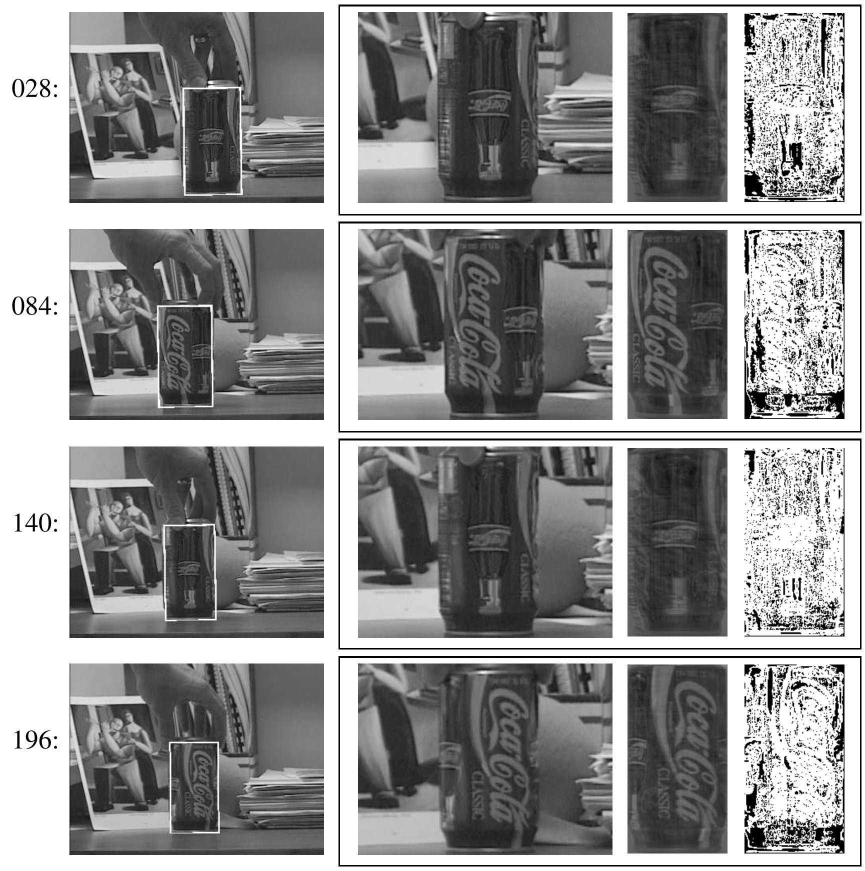


Fig. 8. EigenTracking with translation and divergence over 200 frames. The soda can rotates about its major axis while moving relative to the camera.

Figure 9. A 100 image training set was collected by fixing the wrist position and recording a hand alternating between these four gestures. The eigenspace was constructed and 25 basis vectors were used for reconstruction. In our preliminary experiments we have found brightness images to provide sufficient information for both recognition and tracking of hand gestures (cf. [9]).

Figure 10 shows the tracking algorithm applied to a 100 image test sequence in which a moving hand executed the four gestures. The motion in this sequence was large (as much as 15 pixels per frame) and the hand moved while changing gestures. The figure shows the backprojected box corresponding to the eigenspace model and, to the right, on top, the reconstructed image. Below the reconstructed image is the "closest" image in the original training set (taken to be the smallest Euclidean distance in the space of coefficients). While more work must be done, this example illustrates how eigenspace approaches might provide a view-based representation of articulated objects. By allowing parameterized transformations we can use this representation to track and recognize

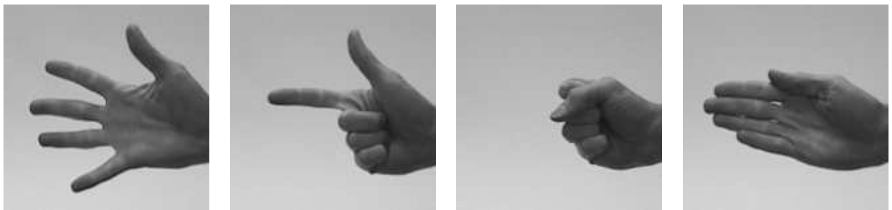


Fig. 9. Examples of the four hand gestures used to construct the eigenspace.

human gestures.

7 Conclusions

This paper has described robust eigenspace matching, the recovery of parameterized transformations between an image region and an eigenspace representation, and the application of these ideas to EigenTracking and gesture recognition. These ideas extend the useful applications of eigenspace approaches and provide a new form of tracking for previously viewed objects. In particular, the robust formulation of the subspace matching problem extends eigenspace methods to situations involving occlusion, background clutter, noise, etc. Currently these problems pose serious limitations to the usefulness of the eigenspace approach. Furthermore, the recovery of parameterized transformations in a continuous optimization framework provides an implementation of a *views+transformation* model for object recognition. In this model a small number of views are represented and the transformation between the image and the nearest view is recovered. Finally, the experiments in the paper have demonstrated how a view-based representation can be used to track objects, such as human hands, undergoing both changes in viewpoint and and changes in pose.

References

1. A. Baumberg and D. Hogg. Learning flexible models from image sequences. In J. Eklundh, editor, *ECCV-94*, vol. 800 of *LNCS-Series*, pp. 299–308, Stockholm, 1994.
2. M. J. Black and A. D. Jepson. EigenTracking: Robust matching and tracking of articulated objects using a view-based representation. Tech. Report T95-00515, Xerox PARC, Dec. 1995.
3. M. Black and P. Anandan. The robust estimation of multiple motions: Affine and piecewise-smooth flow fields. *Computer Vision and Image Understanding,* in press. Also Tech. Report P93-00104, Xerox PARC, Dec.1993.
4. M. J. Black and P. Anandan. A framework for the robust estimation of optical flow. In *ICCV-93*, pp. 231–236, Berlin, May 1993.
5. A. Blake, M. Isard, and D. Reynard. Learning to track curves in motion. In *Proceedings of the IEEE Conf. Decision Theory and Control*, pp. 3788–3793, 1994.
6. A. F. Bobick and A. D. Wilson. A state-based technique for the summarization and recognition of gesture. In *ICCV-95* , pp. 382–388, Boston, June 1995.

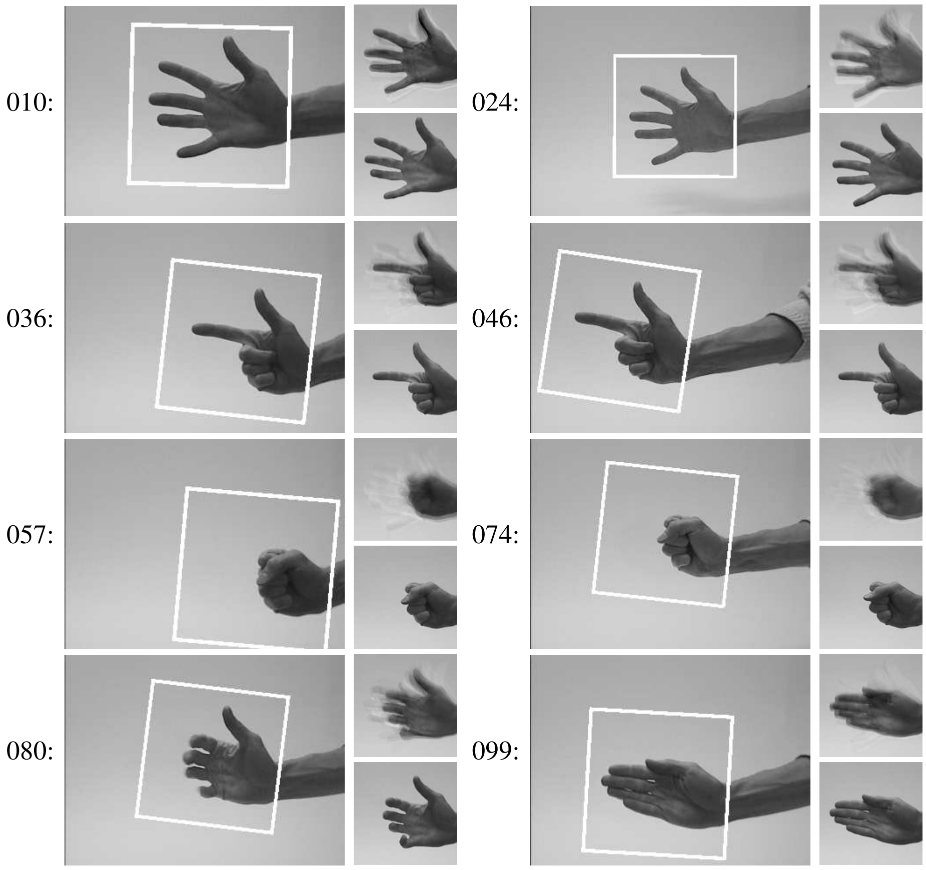


Fig. 10. Tracking and recognizing hand gestures in video.

7. C. Bregler and S. M. Omohundro. Surface learning with applications to lip reading. *Advances in Neural Information Processing Systems 6*, pp. 43–50, San Francisco, 1994.
8. F. R. Hampel, E. M. Ronchetti, P. J. Rousseeuw, and W. A. Stahel. *Robust Statistics: The Approach Based on Influence Functions*. John Wiley and Sons, New York, NY, 1986.
9. B. Moghaddam and A. Pentland. Probabilistic visual learning for object detection. In *ICCV-95*, pp. 786–793, Boston., June 1995.
10. H. Murase and S. Nayar. Visual learning and recognition of 3-D objects from appearance. *International Journal of Computer Vision*, 14:5–24, 1995.
11. A. Pentland, B. Moghaddam, and T. Starner. View-based and modular eigenspaces for face recognition. In *CVPR-94*, pp. 84–91, Seattle, June 1994.
12. M. J. Tarr and S. Pinker. Mental rotation and orientation-dependence in shape recognition. *Cognitive Psychology*, 21:233–282, 1989.
13. M. Turk and A. Pentland. Face recognition using eigenfaces. In *CVPR-91*, pp. 586–591, Maui, June 1991.

This article was processed using the LaTeX macro package with ECCV'96 style