M.I.T. Media Laboratory Perceptual Computing Section Technical Report No. 512

Luxomatic: Computer Vision for Puppeteering

Andrew D. Wilson
Vision and Modeling Group
MIT Media Laboratory
20 Ames St., Cambridge, MA 02139
(drew@media.mit.edu)

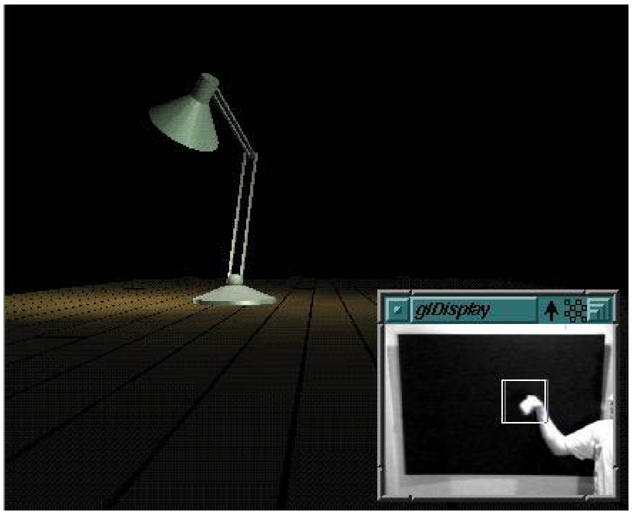

Figure 1: *Luxomatic* is a performance animation system that uses information derived from computer vision processes to drive the animation of a graphical character (a Luxo lamp).

Abstract

In this tech note we describe the *Luxomatic* performance animation system, a computer vision system that tracks a puppeteer's hand to map hand position and shape to an animated graphical model. An eigenvector decomposition of typical hand images are used to both track the hand and extract two-dimensional pose information in real time. *Luxomatic* was constructed as a class project in 1997.

1 Introduction

In this paper we describe a realtime performance animation system that is driven by computer vision input. The goal of the system is to drive a computer graphics animation by the movement and configuration of the user's hand. Inspired by the *Luxo Jr.* short by John Lasseter and Pixar [2], we decided to map the user's hand to an animated Luxo lamp model. Figure 1 illustrates the input and the graphical output. Animating a graphical character via a like human performance is called *performance animation*. A video of the system running located at `http://www.media.mit.edu/~drew/movies`.

The choice of mapping the hand to the Luxo lamp is motivated by two observations. First, Luxo Jr. demonstrates that even a desk lamp may be animated to convey convincing human movement. Much of this impression is due to the precise movement of the head of the lamp, which mimics the movement of a human head. Second, the configuration of the head can be specified by a small number of parameters that may be extracted using computer vision techniques. Extraction of more than a few parameters relating to the coarse pose of the hand is beyond the state of the art in computer vision, particularly for realtime systems driven by low resolution images (though see [5]).

The following sections detail the algorithm used to map the user's hand shape and hand position to the graphical Luxo model. The Luxomatic system illustrates the power of representations that encode constraints in a way that permits simultaneous recognition and parameter-extraction processes.

1.1 Eigenvector decomposition of images

The Luxomatic system must encode the appearance of the hand under various poses for later recognition. In particular, the algorithm must know if an image under question looks like a hand image, and furthermore, determine the pose of the hand. Rather than attempt a geometric model-based recovery of the hand parameters (for example, joint angles), we opted for a strictly appearance-based approach.

One appearance-based approach would be store some number of hand images and compare the image in question to each of the hand prototype images in turn. The drawback of this approach is that it does not take into account the fact that most hand images are similar to one another. Thus a potentially large number of hand images may be required to cover the space of all hand images. See [1] for a related approach.

The approach used in Luxomatic is to compute some small number of *eigenimages* (eigenvectors) from a collection of hand images, as in [6]. These eigenimages span a low-dimensional linear subspace of the total image space. Because the collection of training hand images are similar to one another, only a small number of eigenimages are necessary to closely approximate any one of the training hand images.

We can use the eigenvectors to test if a given image is among the set of hand images without checking against all images by simply projecting the test image into the linear subspace to produce a set of eigenvector coefficients, combining these coefficients with the eigenvectors to produce a reconstruction of the image, and lastly, comparing this reconstructed image against the original. If the reconstruction is a close facsimile of the input image, the input likely

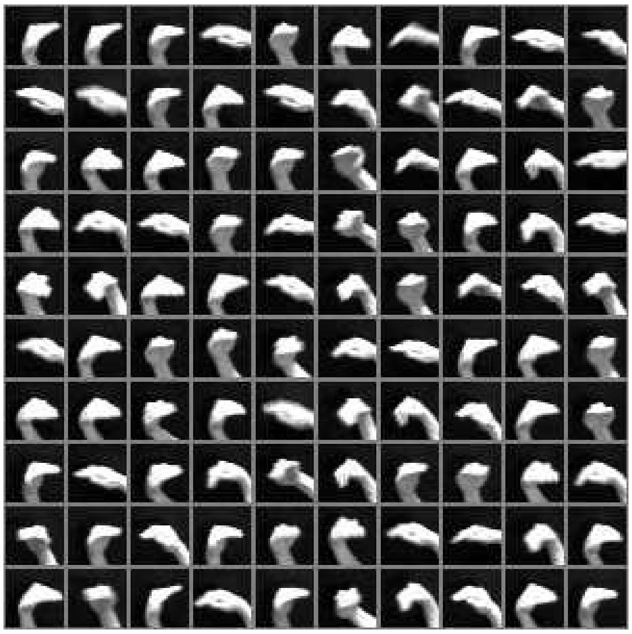

Figure 2: The set of training images of the hand used in computing a basis set for hand images. Each image is 30 x 30 pixels.

belongs to the set of hand images. We similarly use an eigenvector decomposition to match hand images in [7].

For Luxomatic, 100 images of the hand were segmented manually from an image sequence. These images were used in the computation of an eigenvector basis set; the five eigenvectors accounting for most of the variance were kept. The training images are shown in Figure 2.

1.2 Radial Basis Functions for mapping

The set of eigenvector projection coefficients calculated in the reconstruction process encode information about hand pose. In Luxomatic, the projection coefficients are mapped to meaningful hand pose parameters by a radial basis function (RBF) function approximation [4] which is trained offline by the designer.

In our case, the function x to be approximated maps eigenvector coefficients x to two-dimensional pose parameters $f(x)$. The designer supplies a number of pairs $(x_i, f(x_i) = y_)$ to train the approximation of $f(x)$. RBF approximations take the form:

$$f(x) = \sum_i c_i G(||x - x_i||) \quad (1)$$

where x is a vector of eigenvector projection coefficients, x_i are the example projection coefficients used in training the mapping, $G(r)$ is a radial function, and the c_i are derived via a least squares procedure. In this work we use the linear function $G(r) = r$.

A simple linear RBF approximation of the mapping from eigenspace coefficients to pose parameters is appropriate when the pose parameters are continuous in the space of eigenvector coefficients. With the images and pose parameters in Luxomatic, we find this continuity. Figure 4 shows

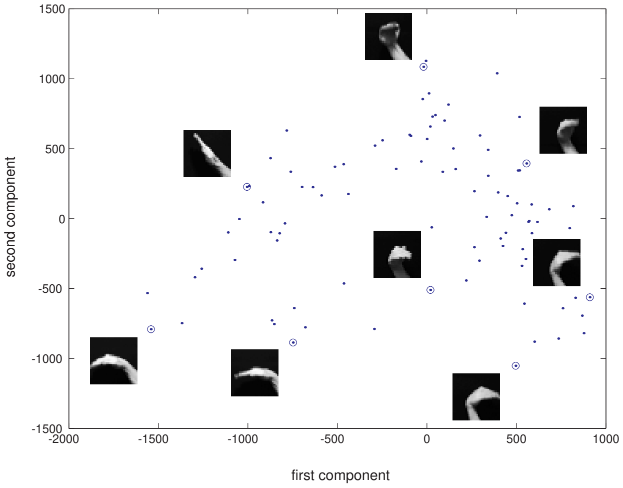

Figure 4: The manifold of hand images in eigenspace, projected onto the first two eigenvectors. The X examples used in computing the mapping from eigenspace coefficients to pose parameters are indicated by circles. The full image associated with each example is also shown. Note how continuous changes in eigenspace correspond to continuous changes in pose.

the examples used in forming the RBF mapping, their position in the space of the first two eigenvector coefficients, and the source image associated with each.

The RBF mapping is also useful in modeling the nonlinearity in the image-to-pose mapping. Any linear transformation, such as an eigenvector projection, will not be able to capture this nonlinearity. The combination of an eigenvector projection for dimensionality dimension and subsequent RBF mapping is computationally efficient, while capturing the nonlinearity in mapping to pose variables. The alternative approach of mapping the images themselves directly to pose parameters using only an RBF would involve a great deal more of computation than the combination of eigenvector projection and subsequent RBF mapping.

We find that with a very limited number of examples an RBF is able to model the nonlinear mapping from eigenvector projection coefficients to the two-dimensional pose parameters of hand yaw and pitch. Figure 4 illustrates the manifold of hand images in eigenspace, and the images used in mapping from eigenspace coefficients to hand pose parameters.

1.3 The Luxomatic Algorithm

Once the combined eigenimage and RBF mapping is computed from training data, Luxomatic uses this mapping to find the hand and extract the pose parameters. The location and pose parameters are then passed to the computer graphics system that complete the performance animation loop.

The hand is located in the image by computing the (x, y) location in the image which minimize the reconstruction error given the eigenvectors computed in training. This test is performed only in the areas of the image that exhibit motion, as determined by a simple image differencing. Once localized, the pose parameters of the hand are computed by the eigenvector to pose RBF approximation.

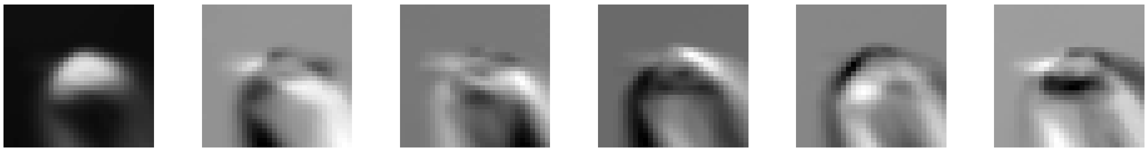

Figure 3: The mean image (leftmost) and top five eigenimages (left to right) computed from the training images.

On an R4400 Indy, this process runs near full frame rate (30Hz) when there is minimal motion in the image.

2 Manifolds as a Constraint on Search

Together, the eigenvectors and RBF mapping approximate the manifold of hand images under a simple pose parameterization. The search and classification stages are relatively efficient, easily running in realtime on a modest processor, for two reasons. First, the manifold serves to constrain the search to the subspace of hands, and secondly, the extraction of the pose parameters occurs simultaneously with search. The Luxomatic system is similar to the object recognition system in Murase and Nayar [3], which uses a manifold representation in eigenspace to recognize objects under various viewing angle and illumination conditions.

References

[1] T.J. Darrell and A.P. Pentland. Space-time gestures. *Proc. Comp. Vis. and Pattern Rec.*, pages 335–340, 1993.

[2] J. Lasseter. Luxo jr. 1986.

[3] H. Murase and S. Nayar. Visual learning and recognition of 3-D objects from appearance. *Int. J. of Comp. Vis.*, 14:5–24, 1995.

[4] T. Poggio and F. Girosi. Networks for approximation and learning. *Proc. IEEE*, 78(9), 1990.

[5] J. Rehg. *Visual analysis of high DOF articulated objects with application to hand tracking*. PhD thesis, Caregie Mellon University, Dept. of Electrical and Computer Engineering, April 1995.

[6] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[7] A. D. Wilson and A. F. Bobick. Learning visual behavior for gesture analysis. In *Proc. IEEE Int'l. Symp. on Comp. Vis.*, Coral Gables, Florida, November 1995.