Michele Covell and Chris Bregler, "Eigen-points," *Proceedings of 3rd IEEE International Conference on Image Processing*. Lausanne, Switzerland 16-19, 1996. pp. 471-4 vol. 3.

EIGEN-POINTS

*Michele Covell and Christoph Bregler**
covell@interval.com bregler@interval.com

Interval Research Corporation
1801 Page Mill Road, Bldg. C, Palo Alto, CA 94304, USA

ABSTRACT

Eigen-points places control points onto unmarked images. The control points are the image locations corresponding to fiduciary points on an object. For example, we might designate ten points on the outside boundary of the lip as fiduciary points on a face. Then, the control points mark the image locations where those points on the outside lip boundary appear. The control-point locations are estimated using a coupled manifold model, which describes the joint variation of the image appearance and the control-point location.

This paper first discusses why this problem is interesting and then reviews previous approaches to placing control points on images of deformable objects. The next section of the paper outlines our approach to placing control points automatically. Finally, some results from our analysis are presented.

1 MOTIVATION

Being able to annotate images with control points placed at fiduciary locations is useful in many application areas. This section discusses automatic image morphing. Other applications include image segmentation, video compositing, interactive video, bootstrapping annotated databases, and in-betweening for animation.

Automatic morphing [1] can be done using eigen-points by matching each of the images separately against the coupled manifold model and using those matches to locate fiduciary points in the images. The corresponding fiduciary points in the two images are then used as constraints in morphing. Figure 1 provides an example of such a morph.

2 CURRENT APPROACHES to point location on images of deformable objects

Active contour models (snakes) [2][3] can easily be thought of as finding point locations at the nodal points of the contour model. However, there is no direct link between the image appearance (the external-energy term) and the shape constraints (the internal-energy term). This makes the discovery of "correct" energy functional an error-prone process.

Shape-plus-texture models [4][5] describe the appearance of an object using shape descriptions (e.g. contour locations or multiple point locations) plus a texture description (e.g. the expected grayscale values at specified offsets relative to the shape-description points). Unfortunately, they are forced to rely on iterative solutions, since they need an estimate of the unknown shape parameters in order to process the image data. Furthermore, the shape– and texture–models do not explicitly take advantage of the coupling between shape and the image data.

When deriving models to estimate unknown parameters, what should be captured is the coupling between observable parameters (like image grayscale values) and the unknown parameters, not the independent descriptions of the unknown parameters and of the normalized known parameters. This is similar to the difference between reconstructive models (models that allow data to be reconstructed with minimum error) and discriminant models (models that allow unknown classification data to be estimated with minimum error). We are not interested in the optimal description of shape or texture, individually. Instead we are interested in the optimal description of how to discriminate different shapes based on the observed image data. The next section describes our approach to this coupled-description problem.

3 EIGEN-POINTS APPROACH to placing control points

Using eigen-points, the problem of locating fiduciary points on an unmarked image using the information from previously marked images is solved in two stages. The first stage is to locate the feature of interest—for example, the actors' lips. This can be done using template- or model-based matching. For example, eigenfeatures [6] can be used to locate each feature using an affine manifold model.

The second stage is to then place the control points around the feature, marking the same fiduciary points as were marked in the

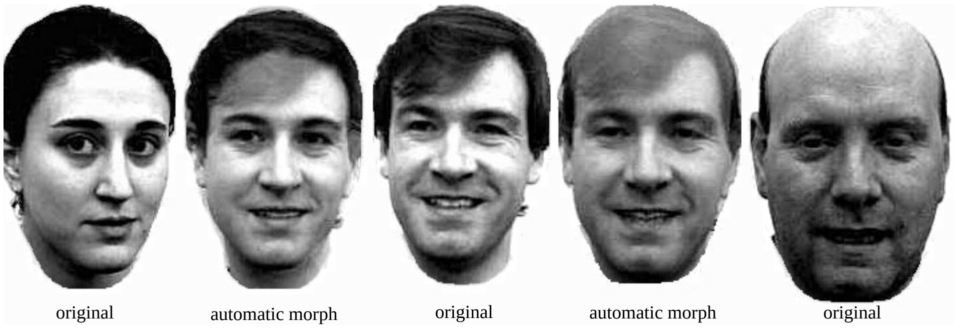


Figure 1: Examples of image morphs using automatically placed correspondence points. The control-point locations used in these morphs were estimated automatically by eigen-points. Constraints were placed around eyes, nose, mouth, chin and ears. No constraints were placed on the hair, neck or shoulders.

*currently at U.C. Berkeley, Berkeley CA 94720 training data—for example, the outer boundary of the lips. Once the feature locations are estimated, the control-point placement is completed by extending the affine model approach to include estimation of “hidden dimensions”. These hidden data dimensions are the locations of the control points associated with the feature. To estimate the values along the hidden dimensions, a feature model that captures the coupling between the observable dimensions (the grayscale values) and the hidden dimensions (the control-point locations) is used.

Control-point placement around a feature location is done in eigen-points in three steps. First, at each estimated feature location, the observed variations (the variations of the grayscale values from their expected values) are projected onto the grayscale subspace of the coupled manifold, giving the strength for each principal component in that pattern of variation. These strengths are then scaled according to the coupling ratios between the grayscale and the control-point subspaces. This re-scaling of the dimensions allows us to then take the projected (grayscale subspace) location on the coupled manifold and reconstruct the hidden data values (the control–point locations). The variations in control-point locations (their deviations from the expected locations) are retrieved by projecting the scaled manifold location back into control-point $(\Delta x, \Delta y)$ coordinates, using the principal components of the control-point locations.

3.1 Training on coupled control-point and feature image data

The coupled grayscale/control-point models are computed using the training database. The training data includes both feature images and (x, y) locations for the control-points associated with that feature, relative to the “origin” defined by the feature location.

The initial processing to derive this coupled manifold model is similar to that for eigen-features [6]. Feature subimages are analyzed to get $\bar{f}$, the $N_x N_y$-length vector of expected image values, and F , an unbiased matrix of image data. Similarly, the L control-point locations given with each image are analyzed to get $\bar{p}$, the $2L$ -length vector of expected control-point locations, and P , an unbiased matrix of control-point locations from the training data. These two matrices are combined into a coupled image/control-point matrix $\left[\mathbf{F}^T \ \ \mathbf{P}^T\right]^T_,$ with each image column of F aligned with the corresponding control-point column of P . The most significant left and right singular vectors and the corresponding singular values of this matrix are computed. For simplicity of explanation, we will describe the process using an SVD of the coupling matrix itself. Using the SVD,

$$\begin{bmatrix} \mathbf{F} \\ \mathbf{P} \end{bmatrix} = \left[\begin{array}{c|c} \mathbf{U_F} & \\ & \mathbf{U}_\perp \\ \mathbf{U_P} & \end{array}\right] \begin{bmatrix} \Sigma_K & \mathbf{0} \\ \mathbf{0} & \Sigma_\perp \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{V} & \mathbf{V}_\perp \end{bmatrix}^T \qquad 1$$

where the first K components of the decomposition are considered significant and the remaining are treated as noise dimensions and where $\mathbf{U_F}$ and $\mathbf{U_P}$ are the $N_x N_y \times K$ and the $2L \times K$ matrices corresponding to the image and the control-point subspaces within the K -dimensional coupled manifold, respectively.

3.2 Estimating a feature’s control-point locations

Control-point placement takes the subimage from the area surrounding the estimated feature location in the new image and projects that onto the coupled manifold:

$$\hat{x} = \Sigma_K^{-1} \mathbf{U_F}^{-1} (f - \bar{f}) \qquad 2$$

and then projects that coupled manifold location into the control-point subspace:

$$\hat{p} = \mathbf{U_P} \Sigma_K \hat{x} + \bar{p} = \mathbf{U_P} \mathbf{U_F}^{-1} (f - \bar{f}) + \bar{p} \qquad 3$$

This is the general form for estimating control point locations from (unlabeled) image data. The next two sections present alternative implementations of this estimation equation. Section 3.2.1 provides an approximate implementation, with reduced computational requirements. Section 3.2.2 provides an exact implementation, with corrections for common noise sources.

3.2.1 Approximate control-point location

Equation 3 can be solved approximately, without computing a matrix inverse, as long as $\mathbf{U_F}$ is nearly orthogonal. This will be the case, in situations where the signal-to-noise ratio of the image subspace is much higher than that of the control-point subspace. Typically this happens when low-resolution images with control-point locations marked at integer-pixel positions are used in training. The computational savings of this approach are increased when features are located using a manifold model derived from the image subspace of the coupled model.

When $\mathbf{U_F}$ is nearly orthogonal, Equation 3 can be approximated by:

$$\hat{p} = \mathbf{U_P} \mathrm{L_F}^{-2} \mathbf{U_F}^T (f - \bar{f}) + \bar{p} \qquad 4$$

where $\mathrm{L_F}$ is a diagonal matrix with the *i*’th entry equal to the vector length of $\mathbf{U}_{\mathbf{F}i}$.

This approximation can be combined with feature location using the image subspace of the coupled model for further computational savings. If the features are located using the manifold model implied by $\{\mathrm{L_F}^{-1} \mathbf{U_F}^T, \Sigma_K \mathrm{L_F}^{-1}\}$ (instead of by the optimal manifold model reported in [6]), these two steps can share the computations that involve $\mathrm{L_F}^{-2} \mathbf{U_F}^T$. The optimal feature model is very close to $\{\mathrm{L_F}^{-1} \mathbf{U_F}^T, \Sigma_K \mathrm{L_F}^{-1}\}$ under the same conditions described above: namely, much higher signal-to-noise ratios in the training set’s image subspace than in its control-point subspace.

3.2.2 Exact control-point location

When high accuracy is required, the true inverse to $\mathbf{U_F}$ should be used and corrections should be included for expected sources of labeling noise.

When the exact inverse of $\mathbf{U_F}$ is used, the computational noise from multiplying by $\mathbf{U_P} \mathbf{U_F}^{-1}$ can be reduced. This is possible due to the joint structure of $\mathbf{U_F}$ and $\mathbf{U_P}$: namely, $\mathbf{U_F}^T \mathbf{U_F} + \mathbf{U_P}^T \mathbf{U_P} = \mathbf{I}$. With this constraint, a C-S decomposition [7] can be used to find SVDs for $\mathbf{U_F}$ and $\mathbf{U_P}$ with identical right singular vectors. The control point locations can then be estimated in a new image from:

$$\hat{p} = \mathbf{Q_P} (\Sigma_\mathbf{P} \Sigma_\mathbf{F}^{-1}) \mathbf{Q_F}^T (f - \bar{f}) + \bar{p} \qquad 5$$

where $\Sigma_\mathbf{F}$ and $\Sigma_\mathbf{P}$ are the non-zero singular values of $\mathbf{U_F}$ and $\mathbf{U_P}$, $\mathbf{Q_F}$ and $\mathbf{Q_P}$ are the corresponding left singular vectors, and $\mathbf{V_{FP}}$ is the shared right singular vectors (which is used below).

The estimation process is now a simple sequence of: orthonormal projection (onto the manifold), component scaling, and orthonormal projection (into the control-point space). This combination of steps will have lower computational noise than using $\mathbf{U_P} \mathbf{U_F}^{-1}$ directly. Even when component scaling is replaced by a full matrix multiply, it is best completed in the manifold subspace, in order to reduce the dimensionality of the full matrix multiply.

Inaccuracies are introduced into the labeling process by the choice of the dimension of the coupled manifold model itself. The selection of K is a difficult and, in some sense, arbitrary choice. The first K component directions are all treated as if they are determined solely by the signal component of the coupling data, while the other component directions are treated as if they contain no information about the coupling data. This problem can be corrected by replacing this hard classification with a gradual roll-off across the coupling components. This correction results in the replacement of $\Sigma_\mathbf{P}\Sigma_\mathbf{F}^{-1}$ with a general matrix, which combines a gradual roll-off across the principle components of the coupled dataset with the scale changes dictated by $\Sigma_\mathbf{P}\Sigma_\mathbf{F}^{-1}$. In particular:

$$\hat{p} = \mathbf{Q}_\mathbf{P}(\Sigma_\mathbf{P}\mathbf{X}_{QQ}\Sigma_\mathbf{F}^{-1})\mathbf{Q}_\mathbf{F}^{T}(f - \bar{f}) + \bar{p} \qquad 6$$

where $\mathbf{X}_{QQ} = \mathbf{V}_\mathbf{FP}^{T}\Sigma_K^{-2}(\Sigma_K^{2} - \sigma_{cn}^{2}\mathbf{I})\mathbf{V}_\mathbf{FP}$ and σ_{cn}^{2} is our best estimate for the component noise in the coupled training data. Including $\mathbf{X}_{QQ}$ in the computation provides a gradual roll-off across the coupling components, de-emphasizing the parts of the coupled data which are likely to come from noisy components.

Noise in the new, unlabeled image will introduce another type of error. Input image noise will inflate our estimates of the observed variations and will result in incorrect control-point estimates. Regularizing the inverse of $\Sigma_\mathbf{F}$ can reduce the effects of this input noise.

Equation 6, along with noise-level normalization prior to the analysis [7] and regularization to account for the expected input noise, is what is used to get the results shown in Figure 1 and the some of the results discussed in the next section.

4 RESULTS

A training database was formed from images of seven people, starting with five original images of each person. These images were marked with 235 control points: 56 around the outline of the head, face, and ears; 29 each around the left and right eyes, irises and eyebrows; 31 around the nose and nostrils; and 90 around the boundaries of the lips, teeth and gums, and on the smile lines. These control points were grouped into 18 "features" (using K-means clustering on the average and variance of their separation distances). For each of these clusters in each of the images, the corresponding feature location was taken to be the median (x,y) values of the control point locations in the cluster. For the sake of simplicity, the dimensions of each feature's subimage were specified manually: this required only 18 subimage specifications (one for each feature cluster). The original training database of 35 images was extended using automatic morphing to create one in-between image for each same-person pair in the original database. These in-between images did not require any additional manual labeling, since their control point locations can be computed directly from the originals. All the images were also flipped horizontally and added to the database. In this way, a database of 2*(7*(5*4)+35)=350 images was formed.

Noise-level normalization was completed prior to the principal components analysis on the coupled data. The noise level in each image dimension was estimated from its average value, assuming that the noise variance is proportional to the average pixel value, above some lower bound. The noise variance in the control point locations was assumed to be one to four pixels, depending on how directly visible the point was in most images (four pixels for the top of the head and the gum lines; two pixels for the top and bottom of the iris and the top of the lower teeth; one pixel for everything else).

The performance in placing control points, given the correct feature location, was tested, on the same set of images as were in the training database. The feature location was, again, the median control-point (x,y) values for the feature cluster. The average error in control point location (relative to the marked location) was less than one pixel when using either approach: the error with the approach from Section 3.2.1 (0.9 pixels) was not significantly different from that with the approach from Section 3.2.2 (0.8 pixels). The maximum error in both cases was about 16 pixels, as shown in Figure 2.

The performance in locating features and placing control points was tested on a separate set of images, showing new shots of the people that were used in the training set. Using the approach from Section 3.2.1 (with the image subspace of the coupled manifold for feature location), the average error was 1.5 pixels in feature location and 3.0 pixels in control-point location. Using the approach from Section 3.2.2 (with eigen-features [6] for feature location), the average error was 1.0 pixel in feature location and 1.5 pixels in control point location. Examples are shown in Figure 3.

5 CONCLUSIONS

Eigen-points provides explicit estimates of fiduciary point locations, which are useful in applications such as image morphing, lip-synching, in-betweening, interactive video, image segmentation and video compositing. In eigen-points, multiple control points are associated with an image feature. Feature locations are then estimated and used with coupled affine manifolds to estimate the control-point locations around the feature. The eigen-point training data can also be used to group control-points and image regions into features and their associated control-points. This grouping process can use the training control-point information to define the "correct" alignment of a feature across example images and to minimize the internal deformations within a single image feature. This capability is useful both for eigen-points and for other model-based matching algorithms, such as eigen-features.

REFERENCES

[1] M. Covell, M. Withgott, "Spanning the gap between motion estimation and morphing," *Proc International Conference on Acoustics, Speech, and Signal Processing*, 1994

[2] M. Kass, A. Witkin, D. Terzopoulous, "Snakes, Active Contour Models." *Proc International Conference on Computer Vision*, 1987.

[3] C. Bregler, S. Omohundro, "Surface Learning with Applications to Lipreading," *Neural Information Processing Systems*, 1994.

[4] D. Beymer, "Vectorizing Face Images by Interleaving Shape and Texture Computations," *MIT AI Memo 1537*, 1995.

[5] A. Lanitis, C.J. Taylor, T.F. Cootes, "A Unified Approach to Coding and Interpreting Face Images," *Proc International Conference on Computer Vision*, 1995.

[6] B. Moghaddam, A. Pentland, "Maximum likelihood detection of faces and hands," *Proc International Workshop on Automatic Face and Gesture Recognition*, 1995.

[7] G. Golub, C. Van Loan, *Matrix Computations*, Johns Hopkins University Press, 1989.

ACKNOWLEDGEMENTS

We would like to thank Malcolm Slaney of Interval Research Corporation for his comments both on this paper and on the underlying research and A. Lanitis of University of Manchester for the use of his database of face images.

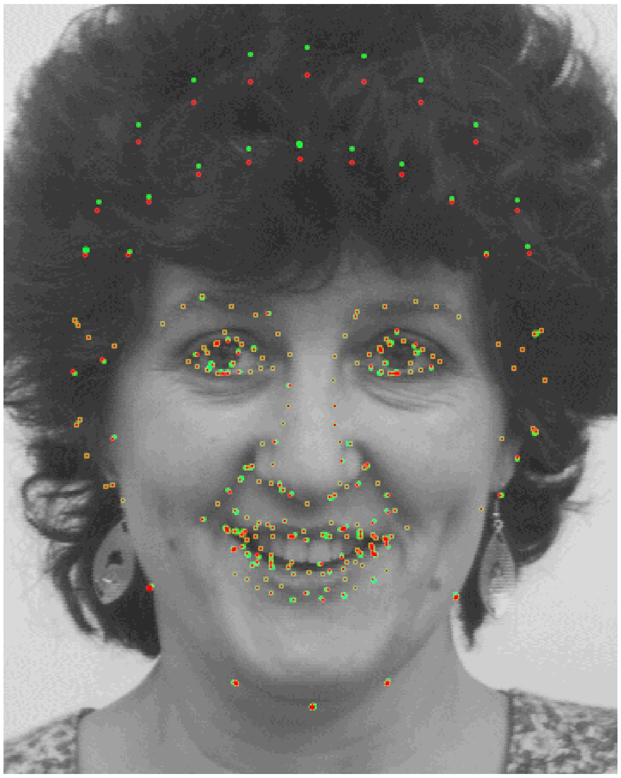
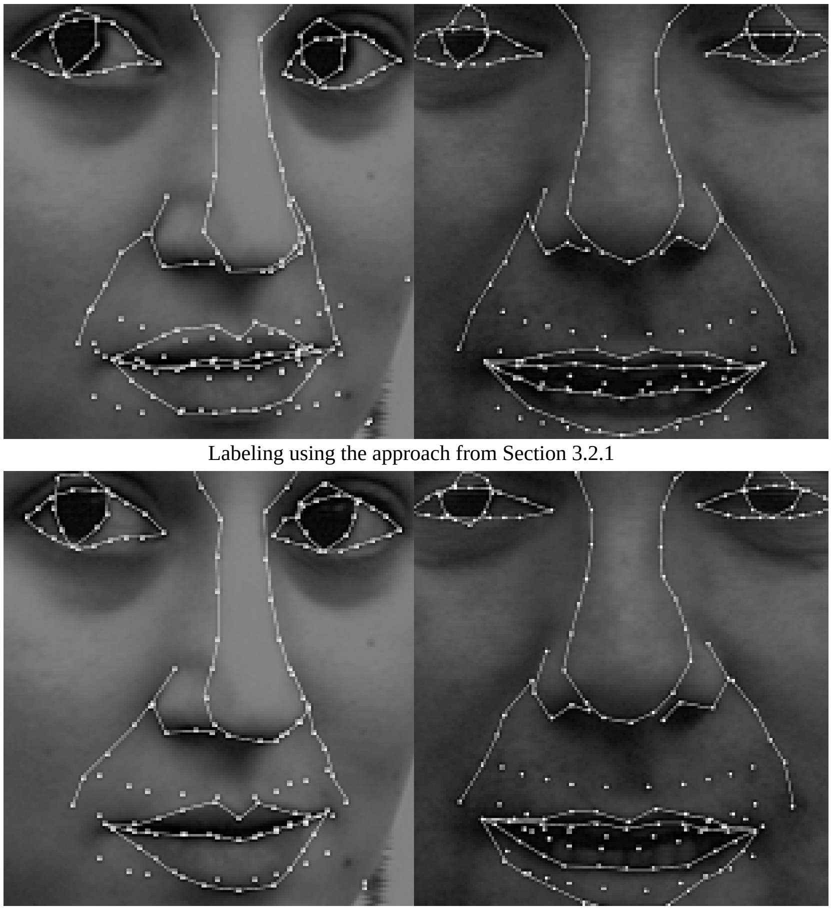


Figure 2: Examples of re-labeled training data. The images that were used as training data were themselves re-labeled, using the coupled models which were derived from them. The new estimates are shown in green; the original training data in red. The approaches described in Section 3.2.1 and in Section 3.2.2 resulted in nearly identical relabelings, with the labeling errors in the same directions and in nearly the same amplitude at each of the control points. The difference in the average error between the two approaches on this set of inputs was not significant (0.9 pixels vs. 0.8 pixels).

This image shows the worst errors, with offsets around 16 pixels at the top of the head. Other locations where errors tended to occur were at the gum lines and at the top of the forehead. It is not clear how much of this error is due to poor training data (i.e. inconsistent original labeling) and how much is due to the reduced “compliance” of the manifold models at these control points. The compliance was effectively reduced by the increased noise variance estimates for these locations. Another possible source of error would be an incorrect model of the way in which the image noise varies with amplitude.



Labeling using the approach from Section 3.2.1

Labeling using the approach from Section 3.2.2

Figure 3: Labeling images which were not in the training data base. These images, along with others, where used as a disjoint testing set: none were included in the training database. (Other images of these same people were included in the training database.) labeling using the approach from Section 3.2.1 with the image subspace of the coupled manifold model for feature location (top row) averaged about 1.5 pixels in feature-location error and 3 pixels in control-point-location error. labeling using the approach from Section 3.2.2 with eigen-features for location (bottom row) averaged about 1 pixel in feature-location error and 1.5 pixels in control-point-location error.

The differences in the performance in these two approaches can be seen most clearly in:

– the control-point locations around the irises in the left images (the approach from Section 3.2.1 does not follow the right edges of the irises well);

– the control point locations along the upper boundaries of the top lip in the left images (both approaches exaggerate the cleft in the top edge of the top lip; the approach from Section 3.2.1 does a poor job of following the left half of the top edge);

– the control point locations on the boundaries between the upper and lower lips in the left images (the approach from Section 3.2.1 does not align the boundary at the bottom of the upper lip with the boundary at the top of the lower lip);

– the control point locations around the left irises of the right images (the approach from Section 3.2.2 does not trace out the desired circle for the iris); and

– the control point locations along the upper boundaries of the bottom lip in the right images (the approach from Section 3.2.1 places the control points above the top of the bottom lip, near the top of the bottom teeth).