digital™

A Structure from Motion Approach using Constrained Deformable Models and Appearance Prediction

Sing Bing Kang

Cambridge Research Laboratory

Cambridge Research Laboratory
Technical Report Series

CRL 97/6

October 1997

Cambridge Research Laboratory

The Cambridge Research Laboratory was founded in 1987 to advance the state of the art in both core computing and human-computer interaction, and to use the knowledge so gained to support the Company's corporate objectives. We believe this is best accomplished through interconnected pursuits in technology creation, advanced systems engineering, and business development. We are actively investigating scalable computing; mobile computing; vision-based human and scene sensing; speech interaction; computer-animated synthetic persona; intelligent information appliances; and the capture, coding, storage, indexing, retrieval, decoding, and rendering of multimedia data. We recognize and embrace a technology creation model which is characterized by three major phases:

Freedom: The life blood of the Laboratory comes from the observations and imaginations of our research staff. It is here that challenging research problems are uncovered (through discussions with customers, through interactions with others in the Corporation, through other professional interactions, through reading, and the like) or that new ideas are born. For any such problem or idea, this phase culminates in the nucleation of a project team around a well articulated central research question and the outlining of a research plan.

Focus: Once a team is formed, we aggressively pursue the creation of new technology based on the plan. This may involve direct collaboration with other technical professionals inside and outside the Corporation. This phase culminates in the demonstrable creation of new technology which may take any of a number of forms - a journal article, a technical talk, a working prototype, a patent application, or some combination of these. The research team is typically augmented with other resident professionals—engineering and business development—who work as integral members of the core team to prepare preliminary plans for how best to leverage this new knowledge, either through internal transfer of technology or through other means.

Follow-through: We actively pursue taking the best technologies to the marketplace. For those opportunities which are not immediately transferred internally and where the team has identified a significant opportunity, the business development and engineering staff will lead early-stage commercial development, often in conjunction with members of the research staff. While the value to the Corporation of taking these new ideas to the market is clear, it also has a significant positive impact on our future research work by providing the means to understand intimately the problems and opportunities in the market and to more fully exercise our ideas and concepts in real-world settings.

Throughout this process, communicating our understanding is a critical part of what we do, and participating in the larger technical community—through the publication of refereed journal articles and the presentation of our ideas at conferences–is essential. Our technical report series supports and facilitates broad and early dissemination of our work. We welcome your feedback on its effectiveness.

Robert A. Iannucci, Ph.D.
Director

A Structure from Motion Approach using Constrained Deformable Models and Appearance Prediction

Sing Bing Kang

October 1997

Abstract

In this technical report, we address the problem of recovering 3-D models from sequences of uncalibrated images with unknown correspondence. To that end, we integrate tracking, structure from motion with geometric constraints, and use of deformable 3-D models in a single framework. The key to making the proposed approach work is the use of appearance-based model matching and refinement.

This *appearance-based constrained structure from motion* (AbCSfm) approach is especially useful in recovering shapes of objects whose general structure is known but which may have little discernable texture in significant parts of their surfaces. We applied the proposed approach to 3-D face modeling from multiple images to create new 3-D faces for DECface, a synthetic talking head developed at Cambridge Research Laboratory, Digital Equipment Corporation. The DECface model comprises a collection of 3-D triangular and rectangular facets, with nodes as vertices. In recovering the DECface model, we assume that the sequence of images is taken with a camera with unknown camera focal length and extrinsic parameters (i.e., camera pose). Results of this approach show its good convergence properties and its robustness against cluttered backgrounds.

CRL Technical reports are available on the CRL's web page at
http://www.crl.research.digital.com.

Digital Equipment Corporation
Cambridge Research Laboratory
One Kendall Square, Building 700
Cambridge, Massachusetts 02139

Contents

List of Figures

1 Introduction

The classical approach to recovering 3-D structure from a sequence of images is to calibrate the camera, track the features across the sequence, and then apply stereo techniques using the tracked features. More recent techniques allow 3-D structures to be recovered without explicit camera calibration. Nevertheless, the processes of feature tracking and structure from motion are almost always separate.

In this technical report, we propose an approach that integrates tracking, structure from motion with geometric constraints, and use of deformable 3-D models in a single framework. The input image sequence is assumed uncalibrated, and the image correspondences are also assumed not known. The key to making the proposed approach work is the use of appearance-based model matching and refinement. Another distinguishing feature of this approach is that feature correspondences are not *statically determined*; they may "drift" over time according to how well they satisfy both local image similarity and 3-D geometric constraints.

This *appearance-based constrained structure from motion* (AbCSfm) approach is especially useful in recovering shapes of objects whose general structure is known but which may have little discernable texture in significant parts of their surfaces. A good example of such an object is the human face, where there is usually a significant amount of relatively untextured regions (especially if there is little facial hair) and where the facial structure is known. We applied the proposed approach to 3-D face modeling from multiple images to create new 3-D faces for DECface, a synthetic talking head developed at Cambridge Research Laboratory, Digital Equipment Corporation. The DECface model comprises a collection of 3-D triangular and rectangular facets, with nodes as vertices. In recovering the DECface model, we assume that the sequence of images are taken with a camera with unknown camera focal length and extrinsic parameters (i.e., camera pose).

In our current implementation, we use the frontal shot of the face as the reference image and impose a line-of-sight constraint of 3-D facial nodes using this reference image. We also constrained 3-D model deformation by minimizing an objective function that trade-off minimal change in local curvature and node position with fit to predicted point correspondences and face appearance.

1.1 Prior work

There is a large body of work on the recovery of raw 3-D data from multiple images; they include multibaseline stereo [14], trinocular stereo that combines constant brightness constraint with trilinear tensor (small displacements, only three images) [19], stereo with interpolation [4], and shape from rotation [21, 30]. In a work that unifies image matching with stereo, Xu and Zhang [29] use initially extracted correspondence to estimate the epipolar geometry using a robust estimator. The computed epipolar geometry is then used to recover more correspondences as in classical stereo matching.

Virtually all stereo approaches assume fixed disparity throughout once it has been established, e.g., through a separate feature tracker or image registration technique. Most techniques assume that the camera parameters, intrinsic and extrinsic, are known. Our proposed method integrates the tracker with structure and motion recovery, and does not assume that the focal length is known. In theory, for general camera motion with constant intrinsic parameters, three views are sufficient to recover structure, camera motion, and all five camera intrinsic parameters [7, 20]. For algorithmic stability, we assume only one unknown intrinsic camera parameter, namely the focal length. The aspect ratio is assumed to be unity, the image skew to be insignificant, and the principal point to be coincident with the center of the image.

The approaches specific to face modeling can be partitioned into two categories based on the input, namely range and image data, and images only. In an approach that uses both range and image data, Lee *et al.* [11] use dense 3-D data from Cyberware Color Digitizer$^{\mathrm{TM}}$, and apply 3-D feature-based matching (for facial features such as the nose, chin, ears, eyes) to initialize their 3-D adaptable facial mesh. This facial mesh is subsequently augmented with a dynamic model of facial tissue controlled by facial muscles. Kang *et al.* [10] use as input both range image and corresponding color image of the face. They use color-based 2-D facial feature detection methods to locate the eyes, eyebrows, and mouth. The feature detection involve computing edges in color space followed by contour extraction and smoothing by dilation and shrinking.

The simpliest case of techniques using only images as input involves only two orthogonal views (namely, the front and side views) of the face. Extraction of 3-D face model would then entail profile analysis, identification of facial features from contours, and adjustment of a 3-D face template through interpolation [1, 8].

Lengagne *et al.* use a calibrated stereo pair and use the dense disparity map computed through an interpolation technique [4]. In their approach, the 3-D deformation of the face model is guided by differential features that have high curvature values (such as the nose and eye orbits).

Two representative work that use as input a sequence of face images to refine a 3-D face model are those of DeCarlo and Metaxas [2] and Jebara and Pentland [9]. The first method uses optical flow in an image sequence to move and deform the face model [2] for expression tracking. Facial anthropometric data is used to limit facial model deformations in the initialization and during tracking. The focal length of the camera is assumed to be known approximately. In the second method, the eyes, nose and mouth are tracked, and the structure and motion of the face is estimated using recursive Kalman filtering [9]. The deformation of the face shape is constrained by linear subspace of eigenvectors as a result of Singular Value Decomposition (SVD) of sample face shapes. In this case, the whole face is not tracked. In a more general approach, Fua and Leclerc [5] reconstruct both shape and reflectance properties of surfaces from multiple images. The surface shape is initialized by conventional stereo, and is deformed while minimizing an objective function that is a weighted sum of stereo, shading, and smoothness constraints.

1.2 Organization

In section 2, we describe in detail our approach which we call *appearance-based constrained structure from motion* (AbCSfm). This approach enables 3-D models to be extracted from multiple images despite initially unknown feature correspondences. It is based on image-based registration that is guided by predicted 3-D image appearance and a structure from motion algorithm. To illustrate the proposed approach, we then describe an application that uses AbCSfm to recover 3-D facial models from multiple images in section 3.1. Discussion of the method and a possible variant of it is given in section 4, with a summary subsequently provided.

2 General approach

We have developed an approach that allows us to recover a 3-D model from initially unknown point correspondence and an approximate 3-D template. We call this approach *appearance-based constrained structure from motion* (AbCSfm). The components of AbCSfm, as shown in Figure 1, are

- Image registration (spline-based registration in our case [22]
- Structure from motion (iterative Levenberg-Marquardt batch approach [23]
- Appearance prediction (simple texture resampling [28]). The predicted appearance is computed based on current image point correspondences and structure from motion estimates, and is used to refine image registration.

In this approach, initialization is first done by performing pair-wise spline-based registration using one frame as a reference, with every other frame. This establishes a set of gross point correspondences across the image sequence, from which the camera parameters and model shape are extracted. Subsequently, it iterates over three major steps:

1. *Appearance prediction*
   In this step, for each image other than the reference image, the appearance of the 3-D model given the camera pose and intrinsic parameters is computed and projected onto a new image.

2. *Spline-based image registration*
   The predicted image is registered with the actual image to refine the point correspondences.

3. *Structure from motion*
   Using the refined point correspondences, estimate the new (usually better) estimates of the camera pose and intrinsic parameters, as well as 3-D model shape.

The use of appearance-based strategy is important as it accounts for not only occlusions, but also perspective distortion due to changes in object pose. In contrast to Lowe's approach [13] which uses edges, we use whole predicted images.

2.1 Tracking by spline-based registration

In the spline-based registration framework [22, 24], a new image I_2 is registered to an initial *base image* I_1 using a sum of squared differences formula

$$E(\{u_i, v_i\}) = \sum_i [I_2(x_i + u_i, y_i + v_i) - I_1(x_i, y_i)]^2, \quad (1)$$

where the $\{u_i, v_i\}$'s are the per-pixel *flow* estimates.

In this registration technique, the flow estimates $\{u_i, v_i\}$ are represented using two-dimensional *splines* controlled by a smaller number of displacement estimates $\hat{u}_j$ and $\hat{v}_j$ which lie on a coarser *spline control grid* (Figure 2). This is in contrast to representing them as completely independent quantities (and thus having an underconstrained problem). The value for the displacement at a pixel i can be written as

$$\begin{pmatrix} u(x_i, y_i) \\ v(x_i, y_i) \end{pmatrix} = \sum_j B_j(x_i, y_i) \begin{pmatrix} \hat{u}_j \\ \hat{v}_j \end{pmatrix} \quad \text{or} \quad \begin{pmatrix} u_i \\ v_i \end{pmatrix} = \sum_j w_{ij} \begin{pmatrix} \hat{u}_j \\ \hat{v}_j \end{pmatrix}, \quad (2)$$

where the $B_j(x, y)$ are called the *basis functions* and are only non-zero over a small interval (*finite support*). The $w_{ij} = B_j(x_i, y_i)$ are called *weights* to emphasize that the (u_i, v_i) are known linear combinations of the $(\hat{u}_j, \hat{v}_j)$.

In the current implementation, the spline control grid is a regular subsampling of the pixel grid, $\hat{x}_j = mx_i$, $\hat{y}_j = my_i$, so that each set of $m \times m$ pixels corresponds to a single spline patch. We use

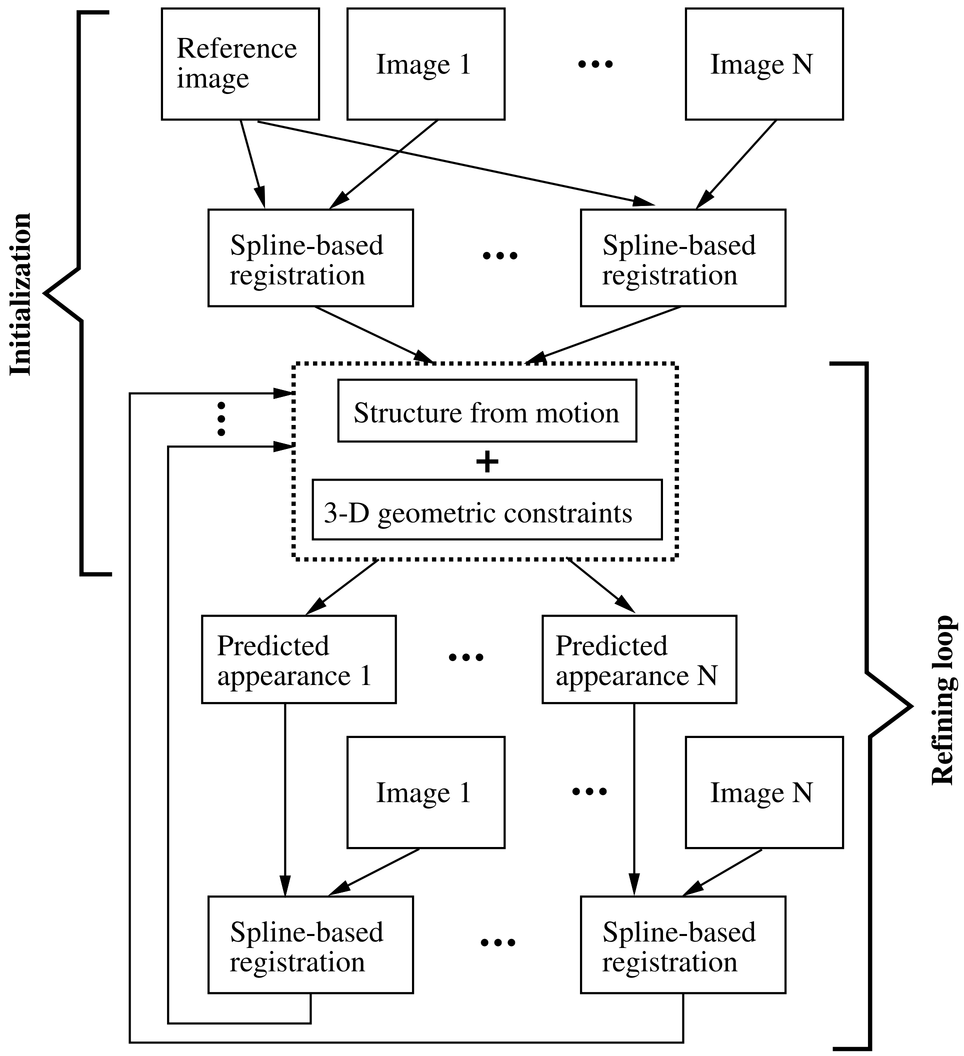

Figure 1: General approach of appearance-based constrained structure from motion (AbCSfm).

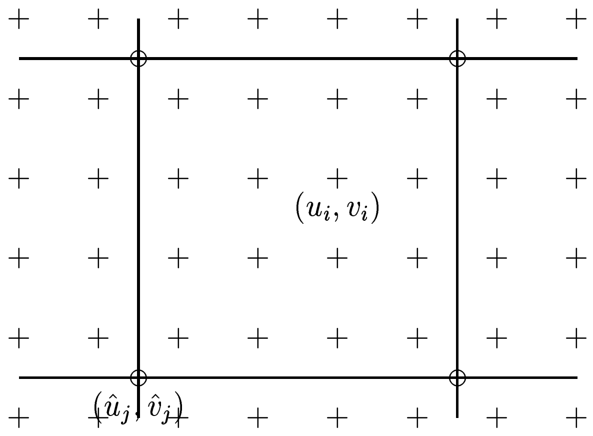

Figure 2: Displacement spline: the spline control vertices $\{(\hat{u}_j, \hat{v}_j)\}$ are shown as circles ($\circ$) and the pixel displacements $\{(u_i, v_i)\}$ are shown as pluses ($+$) [22].

bilinear basis functions, i.e., $B_j(x, y) = \max((1 - |x - \hat{x}_j|/m)(1 - |y - \hat{y}_j|/m), 0)$ (see [22] for a discussion of other possible bases). The local spline-based flow parameters are recovered using a variant of the Levenberg-Marquardt iterative non-linear minimization technique [17].

We also modified (2) to include the weights m_{ij} associated with a *mask* as follows:

$$\begin{pmatrix} u_i \\ v_i \end{pmatrix} = \sum_j m_{ij} w_{ij} \begin{pmatrix} \hat{u}_j \\ \hat{v}_j \end{pmatrix}, \tag{3}$$

where $m_{ij} = 1$ or 0 if the corresponding pixel is in the object or background area respectively. This is necessary to prevent registration of the background areas influencing registration of the projected model areas across images. m_{ij} can also assume values between 0 and 1, especially during the hierarchical search where the images are subsampled and the intensities averaged.

2.2 General structure from motion

The formulation of recovering structure from motion is based on that of [23]. Essentially, we are trying to recover a set of 3-D structure parameters $\mathbf{p}_i$ and time-varying motion parameters T_j from a set of observed image features $\mathbf{u}_{ij}$. The general equation linking a 2D image feature location $\mathbf{u}_{ij}$ in frame j to its 3-D position $\mathbf{p}_i$ (i is the track index) is

$$\mathbf{u}_{ij} = \mathcal{P}\left(T_j^{(K)} \ldots T_j^{(1)} \mathbf{p}_i\right) \tag{4}$$

where the perspective projection transformation $\mathcal{P}()$ is applied to a cascaded series of rigid transformation $T_j^{(k)}$. Each transformation is in turn defined by

$$T_j^{(k)}\mathbf{x} = \mathbf{R}_j^{(k)}\mathbf{x} + \mathbf{t}_j^{(k)} \tag{5}$$

where $\mathbf{R}^{(k)}$ is a rotation matrix and $\mathbf{t}_j^{(k)}$ is a translation applied after the rotation. Within each of the cascaded transforms, the motion parameters may be time-varying (the j subscript is present) or fixed (the subscript is dropped).

The general camera-centered perspective projection equation is

$$\begin{pmatrix} u \\ v \end{pmatrix} = \mathcal{P}_1 \begin{pmatrix} x \\ y \\ z \end{pmatrix} \equiv \begin{pmatrix} \frac{fx+\sigma y}{z} + u_0 \\ \frac{rfy}{z} + v_0 \end{pmatrix} \tag{6}$$

where f is a product of the focal length of the camera and the pixel array scale factor, r is the image aspect ratio, σ is the image skew, and (u_0, v_0) is the principal point. In theory, for general camera motion with constant intrinsic parameters, three views are sufficient to recover structure, camera motion, and all five camera intrinsic parameters [7, 20]. For stability, we assume only one intrinsic camera parameters matter, namely the focal length (the aspect ratio is assumed to be unity).

An alternative object-centered formulation (a more general version of [23]) which we use is

$$\begin{pmatrix} u \\ v \end{pmatrix} = \mathcal{P}_2 \begin{pmatrix} x \\ y \\ z \end{pmatrix} \equiv \begin{pmatrix} \frac{sx+\eta\sigma y}{1+\eta z} + u_0 \\ \frac{rsy}{1+\eta z} + v_0 \end{pmatrix} = \begin{pmatrix} \frac{sx}{1+\eta z} \\ \frac{rsy}{1+\eta z} \end{pmatrix} \tag{7}$$

with the reasonable assumption that $\sigma = 0$ and $(u_0, v_0) = (0, 0)$. Here, we assume that the (x, y, z) coordinates before projection are with respect to a reference frame that has been displaced away from the camera by a distance t_z along the optical axis,[1] with $s = f/t_z$ and $\eta = 1/t_z$. The projection parameter s can be interpreted as a *scale factor* and η as a *perspective distortion factor*. Our alternative perspective formulation results in a more robust recovery of camera parameters under weak perspective, where $\eta \ll 1$, and assuming $(u_0, v_0) \approx (0, 0)$ and $\sigma \approx 0$, we have $\mathcal{P}(x, y, z)^T \approx (sx, rsy)^T$. This is because s and rs can be much more reliably recovered than η, in comparison with the old formulation where f and t_z are very highly correlated.

[1] If we wish, we can view t_z as the z component of the original global translation which is absorbed into the projection equation, and then set the third component of $\mathbf{t}$ to zero.

2.3 Least-squares minimization with geometric constraints

The Levenberg-Marquardt algorithm [17] is used to solve for the structure and motion parameters. Without the geometric constraints, formulation is exactly that of [23]. We are, instead, trying to minimize

$$\mathcal{E}_{\text{all}}(\mathbf{a}) = \mathcal{E}_{\text{sfm}}(\mathbf{a}) + \mathcal{E}_{\text{geom}}(\mathbf{a}) \tag{8}$$

where

$$\mathcal{E}_{\text{sfm}}(\mathbf{a}) = \sum_i \sum_j c_{ij} \left|\mathbf{u}_{ij} - \mathcal{P}(\mathbf{a}_{ij})\right|^2 \tag{9}$$

is the usual structure from motion objective function that minimizes deviation from observed point feature positions. $\mathcal{P}()$ is given in (4), and

$$\mathbf{a}_{ij} = \left(\mathbf{p}_i^T, \mathbf{m}_j^T, \mathbf{m}_g^T\right)^T \tag{10}$$

is the vector of structure and motion parameters which determine the image of point i in frame j. The vector $\mathbf{a}$ contains all of the unknown structure and motion parameters, including the 3-D points $\mathbf{p}_i$, the time-dependent motion parameters $\mathbf{m}_j$, and the global motion/calibration parameters $\mathbf{m}_g$. The weight c_{ij} in (9) describes our confidence in measurement $\mathbf{u}_{ij}$, and is normally set to the inverse variance σ_{ij}^{-2}. Implementational details are given in [23]. In our case, we set c_{ij} to be a value proportional to the least amount of local texture indicated by the minimum eigenvalue of the local Hessian. The local Hessian H is given by

$$H = \begin{bmatrix} \sum_{\mathcal{W}} I_x^2 & \sum_{\mathcal{W}} I_x I_y \\ \sum_{\mathcal{W}} I_x I_y & \sum_{\mathcal{W}} I_y^2 \end{bmatrix} \tag{11}$$

$\mathcal{W}$ being the local window centered at (x, y) and (I_x, I_y) is the intensity gradient at (x, y). If $e_{\min,ij}$ is the minimum eigenvalue at point i in frame j, then

$$c_{ij} = \frac{e_{\min,ij}}{\max_{ij} e_{\min,ij}} \tag{12}$$

This is particularly important in the case of face model recovery because of the possible lack of texture on parts of the face, such as the cheeks and forehead areas. Using this metric for c_{ij} downplays the importance of points on these relatively untextured areas (see, for example, [18, 24]). To account for occlusions, c_{ij} is set to zero if the corresponding point is predicted to be hidden.

The other term in (8) is

$$\mathcal{E}_{\text{geom}}(\mathbf{a}) = \sum_i \left(\alpha_i |h_i - h_i^0|^2 + \beta_i |\mathbf{p}_i - \mathbf{p}_i^0|^2\right), \tag{13}$$

which is the additional geometric constraints that reduces the deformation of the template or reference 3-D model. The quantities with the superscript 0 refers to the reference 3-D model that is to be deformed. h_i is the perpendicular distance of point $\mathbf{p}_i$ to the plane passing through its nearest neighbors (three in our case). In other words, if Π_i is the best fit plane of the neighbor points of $\mathbf{p}_i$, and $\mathbf{p} \cdot \hat{\mathbf{n}}_i = d_i$ is the equation of Π_i, then

$$h_i = \mathbf{p}_i \cdot \hat{\mathbf{n}}_i - d_i \tag{14}$$

α_i is the weight associated to the preservation of local height (in a sense, preserving curvature), and β_i is the weight associated with the preservation of the reference 3-D position. The weights can be made to vary from node to node, or made constant across all nodes, as in our case.

The Levenberg-Marquardt algorithm first forms the approximate Hessian matrix

$$\mathbf{A} = \sum_i \left[\sum_j c_{ij} \left(\frac{\partial \mathcal{P}(\mathbf{a}_{ij})}{\partial \mathbf{a}} \right)^{\mathrm{T}} \frac{\partial \mathcal{P}(\mathbf{a}_{ij})}{\partial \mathbf{a}} + \mathbf{B}(\beta_i) \right] \tag{15}$$

where $\mathbf{B}(\beta_i)$ is a matrix which is zero everywhere except at the diagonal entries corresponding to the ith 3-D point. The weighted gradient vector is

$$\mathbf{b} = \sum_i \left[- \sum_j c_{ij} \left(\frac{\partial \mathcal{P}(\mathbf{a}_{ij})}{\partial \mathbf{a}} \right)^{\mathrm{T}} \mathbf{e}_{ij} + \mathbf{g}_i \right], \tag{16}$$

where $\mathbf{g}_i = (0 \ldots \mathbf{p}_i'^{\mathrm{T}} \ldots 0)^{\mathrm{T}}$, and

$$\begin{aligned} \mathbf{p}_i' &= \alpha_i (h_i - h_i^0) \left(\frac{\partial h_i}{\partial \mathbf{p}_i} \right)^{\mathrm{T}} + \beta_i (\mathbf{p}_i - \mathbf{p}_i^0) \\ &= \alpha_i (h_i - h_i^0) \hat{\mathbf{n}}_i + \beta_i (\mathbf{p}_i - \mathbf{p}_i^0), \end{aligned} \tag{17}$$

from (14) and using the simplifying assumption that each node position is independent of its neighbors (not strictly true). $\mathbf{e}_{ij} = \mathbf{u}_{ij} - \mathcal{P}(\mathbf{a}_{ij})$ is the image plane error of point i in frame j.

Given a current estimate of $\mathbf{a}$, it computes an increment $\delta\mathbf{a}$ towards the local minimum by solving

$$(\mathbf{A} + \lambda \mathbf{I}) \delta \mathbf{a} = -\mathbf{b}, \tag{18}$$

where λ is a stabilizing factor which varies over time [17].

We also impose the line-of-sight constraint on the recovered 3-D point with respect to the reference image.

2.4 Generating predicted appearance

It is relatively easy to render the model given the 3-D surface model (with its facets and vertices) and its position and orientation. The object facets are sorted in order of decreasing depth relative to the camera, and then rendered by texture-mapping the facets in the same order. The rendering technique used in our work is a standard technique in computer graphics, and can be found in [28].

A 3-D model that is a good candidate for our proposed approach is the human face model. Its structure is known and using conventional stereo techniques are not very reliable because the human face usually has significant portions of relatively untextured regions.

3 Application: Mapping new faces to 3-D DECface

3.1 DECface

DECface is a system that facilitates the development of applications requiring a real-time lip-synchronized synthetic face [26]. Originally based on the X Window System and the audio facilities of DECtalk and AF [12], DECface has been built with a simple interface protocol to support the development of face-related applications. The fundamental components of DECface are software speech synthesis, AF (AudioFile), and face modeling.

Of particular importance to us is the face modeling component. It involves texture-mapping frontal view face images (synthetic or real) onto a correctly-shaped wireframe.

Topologies for facial synthesis are typically created from explicit 3D polygons [15]. For simplicity, we construct a simple 2D representation of the full frontal view because, for the most part, personal interactions occur face-to-face. This model consists of 200 polygons of which 50 represent the mouth and an additional 20 represent the teeth (Figure 3). The jaw nodes are moved vertically as a function of displacement of the corners of the mouth [3]. The lower teeth are displaced along with the lower jaw. Eyelids are created from a double set of nodes describing the upper lid, such that as they move, the lids close.

The canonical representation is originally mapped onto the individual's image mostly by hand. This requires the careful placement of key nodes to certain locations, as illustrated in Figure 3 in particular, the corners of the lips and eyes, the placement of the chin and eyebrows, as well as the overall margins of the face.

To generate facial expressions within DECface, two primary muscle types were implemented: linear and sheet. When orchestrated together, these muscles can create universally recognized facial expressions such as anger, fear, surprise, disgust, sadness and happiness. These muscle

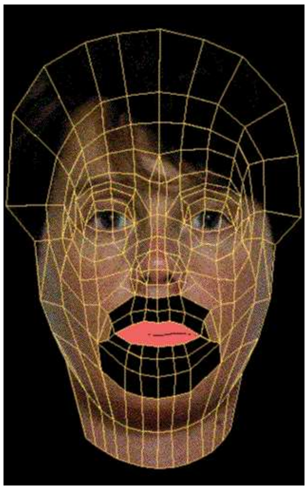

Figure 3: Reconfigured facial geometry on the face image. Notice the close alignment of the nodes around the eyes, mouth, chin and face margins.

types can be described as a geometric deformation function of which the linear muscle has the simplest derivation (for more details see [25]).

DECface is currently being used as a visual and audio feedback mechanism for the Smart Kiosk project at Cambridge Research Lab, Digital Equipment Corp. [27]. The Smart Kiosk can be considered as an enhanced version of the Automatic Teller Machine, with the added capability of being able to interact with the user through body tracking, and gesture and speech recognition. DECface is used to personalize the interaction between the Smark Kiosk and the user. This objective is achieved partly by its ability to communicate its focus of attention to the user population through the gaze behavior of eye contact.

3.2 Mapping faces using one input image

As mentioned in the previous section, mapping new faces to DECface involves texture-mapping frontal view face images (synthetic or real) onto a correctly-shaped wireframe. The original method to generate DECface with a new face is to manually adjust every node, which is a very tedious process. A "generic" separate face (whose DECface topology and 3-D distribution is known) is used as a reference during the process of moving each node within the new face image. This node-moving process is equivalent to the transfer of z information from the "generic" face to the new face. We have investigated methods to automate this process by using templates of

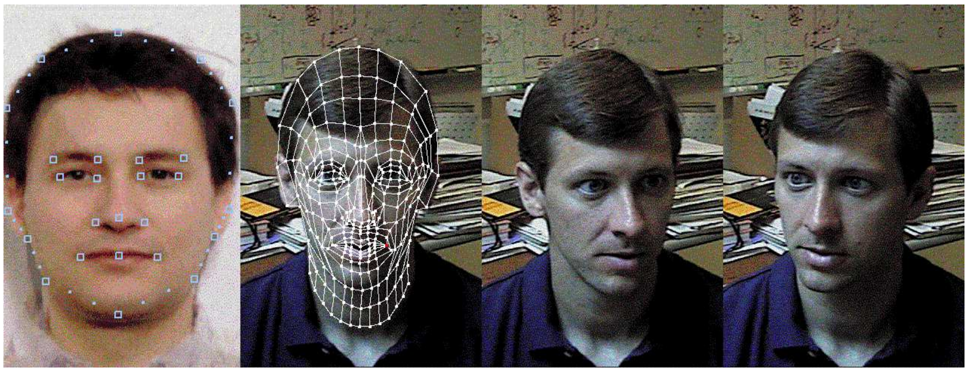

Figure 4: Initial state. The generic face whose DECface topology is known is shown at the left most. The other three images are the input images, with the reference image being the second image from the left.

facial features such as the eyes, mouth, and face profile.

Because only one face input image is used, to generate the appropriate 3-D version of DECface, the canonical height distribution is preserved. This is, however, not always desirable, especially since many human faces have significantly different facial shapes. As a result, to preserve as much as possible the correct shape, we use three input images, each showing a different pose of the face, with one showing the frontal face pose. It is possible, of course, to use two or more than three images to achieve the same goal.

3.3 Mapping faces using three input images

In our work, we use three images of the face at different orientations, with one of them at a frontal pose and used as the reference image. As before, we assume all camera parameters, intrinsic and extrinsic, not known (except that the aspect ratio is one, the image skew is zero, and the principal point is at the image center). We also assume that the point correspondences between the generic face and the reference face has been done as in described in the previous section. This is the same as assuming that the reference shape of the model has been initialized. Note, however, that the point correspondences across the image sequence *are not known*.

We set both α_i and β_i in (13) to 0.25. As mentioned before, the feature track finetuning step involves using the spline-based tracker on the predicted appearance and actual image. However, because the prediction does not involve the background, only the predicted face image portion of the image is involved; the weights associated with the background are set to zero in the spline-based tracker.

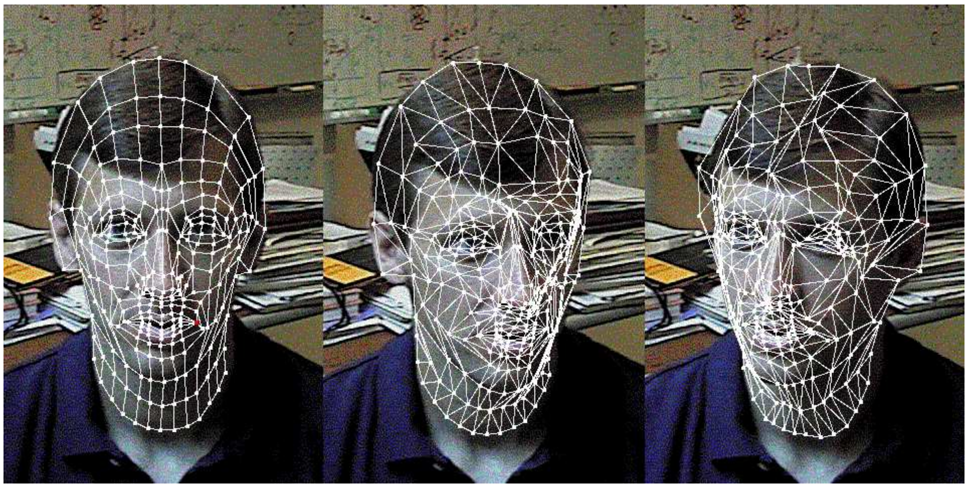

Figure 5: State immediately after performing spline-based registration for the second and third images in the sequence.

An example is shown in Figures 4-8. A comparison between the original 3-D face model and the deformed 3-D face model is shown in Figure 9. As can be seen, the resulting 3-D face has been horizontally stretched somewhat. If the geometric constraints are not imposed (except for just the simple line-of-sight constraint), then the resulting 3-D face model is quite badly deformed, as seen from Figure 10.

The input images of another face is shown in Figure 11. The resulting face model rendered at three different viewpoints is displayed in Figure 12. As can be seen from the side-by-side visual comparison of the 3-D face models prior to and after deformation (Figure 13), the 3-D model has been again stretched horizontally. In addition, the shape of the forehead is made rounder.

4 Discussion

The algorithm may easily fail if the change in object appearance across image sequence is too drastic from one frame to another. In our application of 3-D face modeling, it tolerates face rotation up to about 15°.

A variant of the method would involve the direct incorporation of the optic flow term into the objective function (8) to give

$$\mathcal{E}_{\mathrm{all}}(\mathbf{a}) = \mathcal{E}_{\mathrm{sfm}}(\mathbf{a}) + \mathcal{E}_{\mathrm{geom}}(\mathbf{a}) + \mathcal{E}_{\mathrm{flow}}(\mathbf{u}) \tag{19}$$

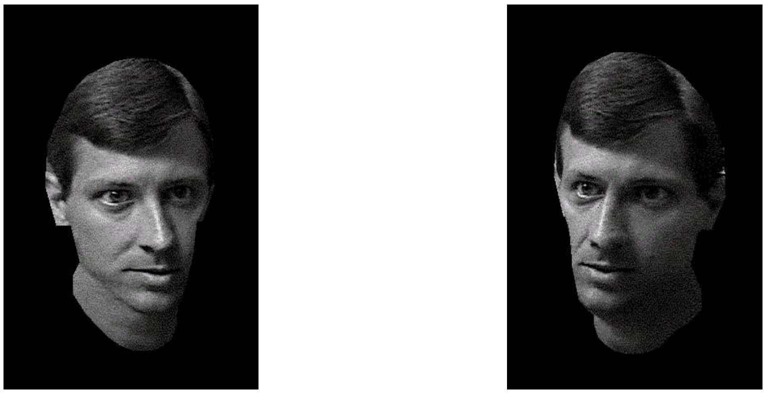

Figure 6: Intermediate predicted facial appearance for the two non-reference images.

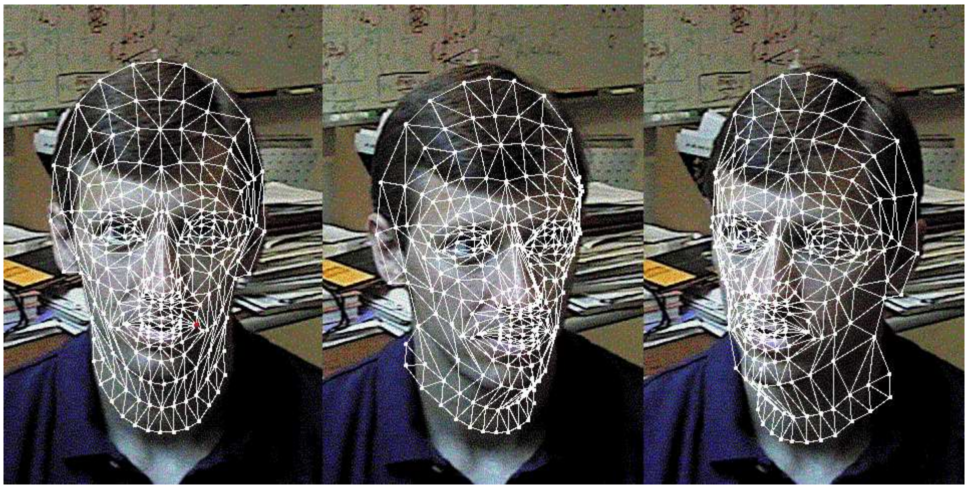

Figure 7: Final state.

Figure 8: Appearance of final 3-D face model at various poses.

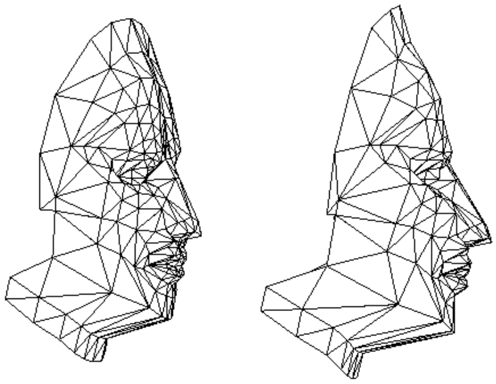

Figure 9: Side views of original (left) and deformed (right) 3-D meshes for the face in Figure 4.

Figure 10: Appearance of final 3-D face model at various poses (with no geometric constraints, apart from line-of-sight).

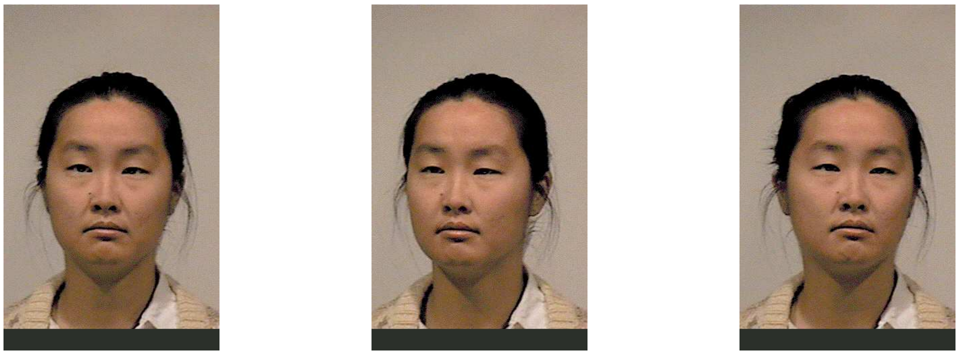

Figure 11: Input images of another face.

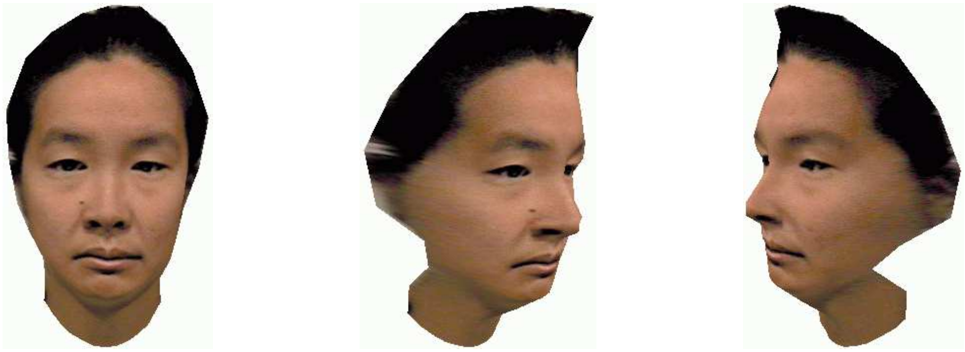

Figure 12: Appearance of final 3-D face model at various poses (from input images shown in Figure 11.

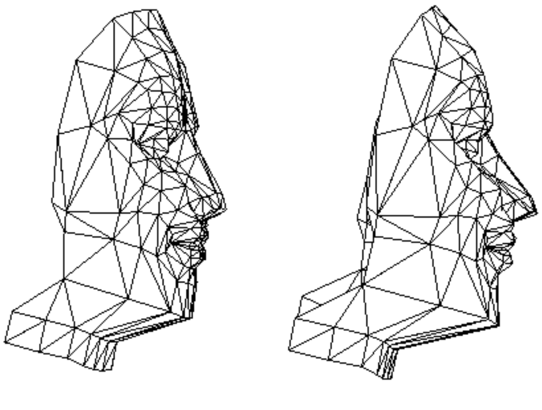

Figure 13: Side views of original (left) and deformed (right) 3-D meshes for the face in Figure 11.

where

$$\mathcal{E}_{\text{flow}}(\mathbf{u}) = \sum_i \sum_{j>1} \gamma_{ij} |I_1(\mathbf{u}_{i1}) - I_j(\mathbf{u}_{ij})|^2 \tag{20}$$

with $I_j(\mathbf{u}_{ij})$ being the intensity (or color) at $\mathbf{u}_i$ on frame j, and γ_{ij} is the weight associated with the point $\mathbf{u}_{ij}$. Note that in our particular application of facial model recovery, since the first frame is the reference frame, $\mathbf{u}_{i1}$ is kept constant throughout.

One problem with directly embedding this term in the structure from motion module is that the flow error term is local and thus unable to account for large motions. It would either require that the initial model pose be quite close to the true model pose, or the addition of a hierarchical scheme similar to that implemented in the spline-based registration method. Otherwise, the system is likely to have better convergence properties if the tracking is performed outside the structure from motion loop. In the current implementation, while having the small perturbations of the model pose would be desirable from the computational point of view (but not from the accuracy point of view), this is not a requirement.

In addition, using the flow error term directly may not be efficient from the computational point of view. This is because at every iteration and incremental step, a new predicted appearance has to be computed. This operation is rather computationally expensive, especially if the size of the projected model is large. Having the tracking module only loosely coupled with structure from motion results in fewer number of iterations in computing the predicted object appearance. Finally, there is the non-trivial question of assigning the weights γ_{ij} relative to the structure from motion and geometric constraint related weights.

Geometric constraints on the face deformation in other forms can also be used. An example would be to use the most dominant few deformation vectors based on SVD analysis of multiple training 3-D faces [9]. A similar approach would be to apply nodal analysis on the multiple training 3-D faces [16, 6] to extract common and permissible deformations in terms of nonrigid modes.

5 Summary

We have described an algorithm called *appearance-based constrained structure from motion* (AbCSfm) that allows 3-D models to be extracted directly from a sequence of uncalibrated images. It is not necessary to precompute feature correspondences across the image sequence. The algorithm dynamically determines the feature correspondences, estimates the structure and camera motion, and uses them to predict the object appearance in order to refine the feature correspondences.

We have used the algorithm to model 3-D faces from a small number of input images, and results have shown the algorithm to be robust and have good convergence properties.

Acknowledgments

I would like to thank Michael Jones and Rebecca Hwa for "lending" their faces.

References

[1] T. Akimoto, Y. Suenaga, and R. S. Wallace. Automatic creation of 3D facial models. *IEEE Computer Graphics and Applications*, pages 16–22, September 1993.

[2] D. DeCarlo and D. Metaxas. The integration of optical flow and deformable models with applications to human face shape and motion estimation. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'96)*, pages 231–238, San Francisco, CA, June 1996. IEEE Computer Society.

[3] V. Fromkin. Lip positions in American English vowels. *Language and Speech*, 7(3):215–225, 1964.

[4] P. Fua. A parallel stereo algorithm that produces dense depth maps and preserves image features. *Machine Vision and Applications*, 6:35–49, 1993.

[5] P. Fua and Y. G. Leclerc. Object-centered surface reconstruction: Combining multi-image stereo and shading. *International Journal of Computer Vision*, 16(1):35–56, September 1995.

[6] J. P. Gourret and J. Khamlichi. A model for compression and classification of face data structures. *Computers and Graphics*, 20(6):863–879, 1996.

[7] A. Heyden and K. Astrom. Euclidean reconstruction from constant intrinsic parameters. In *Proc.s 13th International Conference on Pattern Recognition*, pages 339–343, 1996.

[8] Horace H. S. Ip and L. Yin. Constructing a 3D individualized head model from two orthogonal views. *The Visual Computer*, 12(5):254–266, 1996.

[9] T. S. Jebara and A. Pentland. Parameterized structure from motion for 3D adaptive feedback tracking of faces. In *Conference on Computer Vision and Pattern Recognition*, pages 144–150, San Juan, Puerto Rico, June 1997.

[10] C.-Y. Kang, Y.-S. Chen, and W.-H. Hsu. Automatic approach to mapping a lifelike 2.5D human face. *Image and Vision Computing*, 12(1):5–14, Jan./Feb. 1994.

[11] Y. Lee, D. Terzopoulos, and K. Waters. Realistic modeling for facial animation. *Computer Graphics (SIGGRAPH'95)*, pages 55–62, August 1995.

[12] T. M. Levergood, A. C. Payne, J. Gettys, G. W. Treese, and L. C. Steward. AudioFile: A network-transparent system for distributed audio applications. Technical Report 93/8, Digital Equipment Corporation, Cambridge Research Lab, 1993.

[13] D. G. Lowe. *Perceptual Organization and Visual Recognition*. Kluwer Academic Publishers, Boston, Massachusetts, 1985.

[14] M. Okutomi and T. Kanade. A multiple baseline stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(4):353–363, April 1993.

[15] F.I. Parke. State of the art in facial animation. *ACM SIGGRAPH Course Notes*, 26, 1990.

[16] A. Pentland and S. Sclaroff. Closed-form solutions for physically-based shape modeling and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(7):715–729, July 1991.

[17] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, Cambridge, England, second edition, 1992.

[18] J. Shi and C. Tomasi. Good features to track. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'94)*, pages 593–600, Seattle, Washington, June 1994. IEEE Computer Society.

[19] G. Stein. Model-based brightness constraints: On direct estimation of structure and motion. In *Conference on Computer Vision and Pattern Recognition*, pages 400–406, San Juan, Puerto Rico, June 1997.

[20] P. Sturm. Critical motion sequences for monocular self-calibration and uncalibrated Euclidean reconstruction. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'97)*, pages 1100–1105, Puerto Rico, June 1997. IEEE Computer Society.

[21] R. Szeliski. Shape from rotation. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'91)*, pages 625–630, Maui, Hawaii, June 1991. IEEE Computer Society Press.

[22] R. Szeliski and J. Coughlan. Hierarchical spline-based image registration. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'94)*, pages 194–201, Seattle, Washington, June 1994. IEEE Computer Society.

[23] R. Szeliski and S. B. Kang. Recovering 3D shape and motion from image streams using non-linear least squares. *Journal of Visual Communication and Image Representation*, 5(1):10–28, March 1994.

[24] R. Szeliski, S. B. Kang, and H.-Y. Shum. A parallel feature tracker for extended image sequences. In *IEEE International Symposium on Computer Vision*, pages 241–246, Coral Gables, Florida, November 1995.

[25] K. Waters. A muscle model for animating three-dimensional facial expressions. *Computer Graphics (SIGGRAPH'87)*, 21(4):17–24, July 1987.

[26] K. Waters and T. Levergood. DECface: A system for synthetic face applications. *Multimedia Tools and Applications*, 1:349–366, 1995.

[27] K. Waters, J. Rehg, M. Loughlin, S. B. Kang, and D. Terzopoulos. Visual sensing of humans for active public interfaces. In *Workshop on Computer Vision in Man-machine Interfaces*, Cambridge, UK, April 1996.

[28] A. Watt. *Fundamentals of Three-dimensional Computer Graphics*. Addison Wesley, 1990.

[29] G. Xu and Z. Zhang. *Epipolar Geometry in Stereo, Motion and Object Recognition*. Kluwer Academic Publishers, September 1996.

[30] Y. Z. Zheng. Acquiring 3-D models from sequences of contours. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(2):163–178, February 1994.

A Structure from Motion Approach using Constrained Deformable Models and Appearance Prediction

Sing Bing Kang

CRL 97/6

October 1997