Framework of Integrating 2D Points and Curves for Tracking of 3D Nonrigid Motion and Structure

Min C. Shin, Ramprasad Balasubramanian, Dmitry Goldgof, and Carlos Kim
Department of Computer Science & Engineering
University of South Florida, Tampa, FL 33620
shin, balasubr, goldgof, or ckim2@csee.usf.edu

Abstract

In this paper, we present a method for 3D nonrigid motion tracking and structure reconstruction from 2D points and curve segments from a sequence of perspective images. The 3D locations of features in the first frame are known. The 3D affine motion model is used to describe the nonrigid motion. The results from synthetic and real data are presented. The applications include lip tracking, MPEG4 face player, and burn scar assessment. The results show (1) curve segments are more robust under noise (observed from synthetic data with different Gaussian noise level), and (2) using both feature yields a significant performance gain in real data.

1. Introduction

The area of motion analysis has been receiving a significant amount of attention due to a wide area of application including scene analysis, robot navigation, and object recognition [1]. Recently, the application of the motion analysis has been expanded to the world of the internet. With a high bandwidth requirement of transmitting video data, the research on data compression and 3D reconstruction has been growing. The nonrigid motion is able to describe more naturally occurring motions by eliminating the shape conservation constraint of rigid motion. By describing the change of image content using nonrigid motion model, the video can be compressed to (1) the initial scene information, and (2) the mathematical motion description of the subsequent images [4]. Also, it has been applied to the medical field such as the burn scar assessment study [7].

In this paper, the algorithm for recovering the 3D structure and the motion of object undergoing nonrigid motion from 2D perspective images is developed and analyzed. The integration of two types of features brings an additional applicability where not enough point features is available or the curve features are more naturally available such as lips. To the point feature based algorithm [2], we have added (1) the usage of curve features, and (2) more motion and structure constraint. The Bezier curve representation is chosen to enable an easy integration with point features since a Bezier curve can be described by a set of control points [5]. The significant application of this algorithm is 3D lip tracking [6], the MPEG-4 compression [4], and the burn scar assessment [7].

The 3D structure of the first frame is assumed to be known. This paper attempts to reduce the necessity of having range images of "all" frames, enabling accurate real-time 3D acquisition by recovering 3D information from the subsequent 2D images. The results show that a significant performance gain is observed by using both point and curve features.

2. Problem Formulation

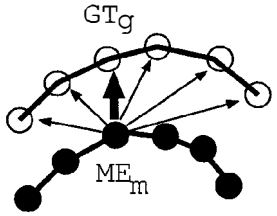

Figure 1. Error between two discrete curves. GT_g is closest to ME_m

Bezier Curve Features The curve features used in this work are described by Bezier curves with 3 control points [5]. They describe quadratic curves. Rather than polynomial representation of the curve, the Bezier curve is easier to interpret geometrically. More importantly, (1) the perspective projection of the 3D Bezier curve was simple projection of 3D control points and change of weights [5], (2) the Bezier curve is affine invariant [3], and (3) the curve is transformed by affine model by applying the affine model to the control points. So the motion of the curve is described by the motion of the control points. The motion of curve segments can be integrated with the points by using 3 control "points" as point features leaving the problem to be motion of points.

A Bezier curve with 3 control points C is described by

$$C(u) = (1-u)^2 \cdot w_0 \cdot T_0 + 2u(1-u) \cdot w_1 \cdot T_1 + u^2 \cdot w_2 \cdot T_2 \quad (1)$$

0-7695-0750-6/00 $10.00 © 2000 IEEE

where $T_i(i = 0, 1, 2)$ are the control points (2D or 3D), $w_i(i = 0, 1, 2)$ are weights for each control point, u parametric direction [0,1]. T_0 and T_2 are the end points of the curve which are observed from an image, while T_1 is estimated by fitting a curve on a set of points describing the curve.

For a 3D curve, weight values (w) are fixed to be 1.0. Its perspective 2D projection is described by (1) 2D control points being perspective projection of 3D control points, and (2) weights being depth/z-value of the corresponding 3D control point [5]. The points along the curves were detected by edge detection or manually, then they were fit to a Bezier curve by using a least-squared method.

Error Measurement In order to compute the error between two curves, we have used Hausdorff-like distance between two curves. The 3D Ground Truth curve (GT) is a set of 3D points corresponding to the chain of 2D points observed from the intensity images. The 3D curve points are obtained from range images. Then, a 3D Machine Estimated (ME) curve is constructed from the three control points (which are estimated by this algorithm). by sampling u at the rate of 1/G where G = number of curve points in GT.

To evaluate a ME curve against a GT curve, each point (ME_m) where m = 1, 2, ..., M is compared against all points of the GT to find the closest point (GT_g) where g = 1, 2, ..., G. Then the error of ME is determined by the mean of the error distance for all points of ME (refer to Figure 1.)

The error of point features is computed by taking an Euclidean distance between ME point location (estimated from this algorithm) and GT point location.

3. Recovery Algorithm

The algorithm uses (1) 2D pixel coordinates of point and curve features, and (2) the depth values of the point and curve features from the first frame, to extract (1) motion (affine motion model), and (2) structure (3D location of point & curve features.) The motion model assumes a *small motion in a local path* which is *constant* throughout the frames. This motion model is suitable for image sequences taken at a fast acquisition rate such as video sequences. In this work, the correspondence information is assumed to be given. In many applications, 2D real-time motion tracking is successfully achieved [6].

Consider an image sequence of I images ($i : 1, ..., I$), K points ($k : 1, ..., K$), and L curves ($l : 1, ..., L$). The the Bezier curve is described by 3 control points and the motion of the curve is defined by the motion of the control points. We simply combine the $3L$ control points and K points to create $3L + K = N$ point features ($n : 1, ..., N$). Given 2D pixel points $P_n^i = [X_n^i, Y_n^i]^T$, we estimate the 3D structure of the point features ($p_n^i = [x_n^i, y_n^i, z_n^i]^T$) and nonrigid motion M. M is an affine motion model where

$$M = \begin{pmatrix} m_{00} & m_{01} & m_{02} & m_{03} \\ m_{10} & m_{11} & m_{12} & m_{13} \\ m_{20} & m_{21} & m_{22} & m_{23} \end{pmatrix} \quad (2)$$

The last column of the motion matrix is fixed to be $[1, 1, 1]^T$ to avoid any trivial solution. It has been shown that second order motion model can be used to describe more complex motion [2].

Let's assume that $\bar{P}_n^i = [U_n^i, V_n^i]$ is a 2D point on image plane where focal length is 1.0 without loosing any generality. Note that $\bar{P}_n^i$ can be computed from P_n^i using the intrinsic parameters of the camera. Using the perspective equation, we can estimate the 3D location $\bar{p}_n^i$ if we know 2D location $\bar{P}_n^i$ and the z_n^i of p_n^i.

$$\bar{p}_n^i = \begin{bmatrix} U_n^i z_n^i & V_n^i z_n^i & z_n^i \end{bmatrix}^T \quad (3)$$

Therefore, the unknowns of the 3D structure is only z_n^i (depth). We can also estimate the 3D location $\hat{p}_n^i$ using estimated M and $\bar{p}_n^{(i-1)}$.

$$\hat{p}_n^i = M\bar{p}_n^{(i-1)} \quad (4)$$

The error function E is

$$E = \sum_{i=2}^{I} \sum_{n=1}^{N} |\bar{p}_n^i - \hat{p}_n^i| \quad (5)$$

Note that the structure of the first frame is known.

The least squares estimation of Marquette-Levenburg is used to estimate structure (z_n^i) and motion (M) by minimizing E.

The motion is *small and constant*, therefore (1) the motion matrix is close to an identity matrix and (2) the depth change between two consecutive frames (Δz) is small. We have empirically found these settings to work well. We impose the following motion and structure constraints.

1. the diagonal elements (m_{11}, m_{22}, m_{33})
   $0.75 \leq m_{ii} \leq 1.25$
2. the off-diagonal elements (m_{ij} where $i \neq j$)
   $-0.25 \leq m_{ij} \leq 0.25$
3. $\Delta z \leq 30mm$.

There are "9 (motion)" + "$(I - 1) \cdot N$ (structure)" unknowns. The algorithm requires ($(I-1) \cdot N \geq 4.5$). For instance, for 3 frames, it (theoretically) requires only 1 curve or 3 points.

The motion initial guess is assigned as

$$\begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$

and the structure initial guess is set by assigning 3D locations of the first frame to all frames.

4. Dataset

We have tested our algorithm on one synthetic and four real image sequences. The dataset is attempted to meet the

motion model principle of this framework: (1) motion is fairly constant and small, and (2) the curves features are planar (Figure 2.) Note that each curve can be planar in any orientation allowing the patch to be non-planar.

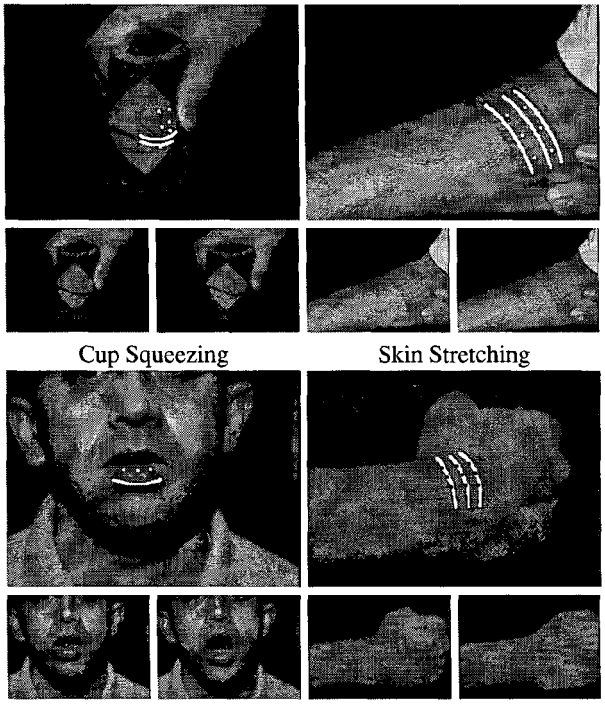

Figure 2. Real Image Sequences. The 2D projection of the recovered 3D point and curve features are marked white on the first image of each sequence

The real data includes noise from the following sources. (1)*range scanner.* K^2T range scanner could capture the range up to the accuracy of 0.5mm. (2) *pixel quantization error.* When a point is projected to 2D pixel coordinate, the location had to be converted to integer value. (3) *inconsistent motion.* The motion between two consecutive frames in real data was not perfectly constant. The results show that the data was constant enough for the algorithm to perform well. Also, the "delta function" [2] can be introduced to accommodate the inconsistency. (4) *feature extraction.* The chaincode of a curve is extracted by hand or first performing edge detection then finding the chaincode to gather points along the curve. The control points of the curve is fitted using the least squares method. The error due to fitting of the Bezier curve was small and did not affect the performance of the algorithm.

5. Results

The results are divided into 2 sections: synthetic data and real data. Error is presented in 4 categories: P_only, P_both, C_only, C_both. "P" indicates that the error is for point features and "C" indicates the curves. "Only" means that only one feature is used for the motion estimation while "both" means that both point and curve features are used. For instance, "C_only" means that only curve features are used during the motion estimation and the error is for the curve features. "P_both" means that both features are used during the motion estimation and the error is for the point features.

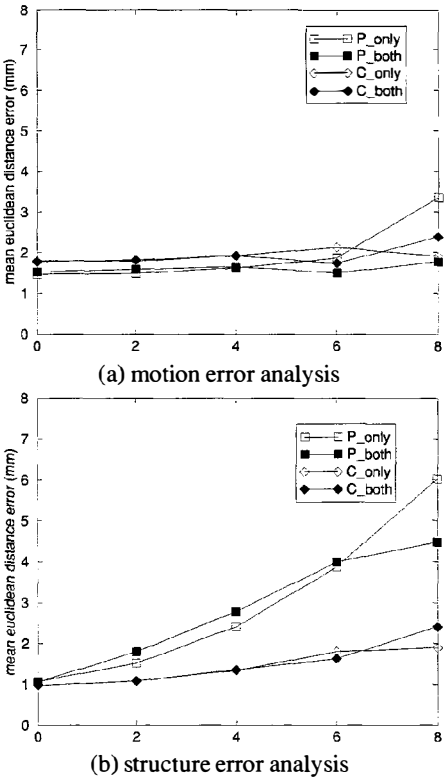

Figure 3. Change of error due to the noise of the 2D input.

5.1. Synthetic Data

In this section, the synthetic data consisting of 3 frames are considered. Three 3D curves and nine 3D points on a plane nearly 1.5m away from the camera has been stretched in x-direction resulting in average motion of 10mm/frame. The 3D curves and points are projected to 2D pixel plane therefore naturally including quantization error.

The experiment on synthetic data carries two purposes. First, the algorithm is to be validated under a simple setting. Second, the effect of noise on performance is investigated. The 2D pixel coordinate of the features have been added with Gaussian noise in x and y direction where $\sigma = (0, 2, 4, 6, 8)$.

First, under no additive noise (aside from the quantization noise), the performance of P_only, P_both, C_only, and C_both are within a small range at a very reasonable level of (1.47mm - 1.80mm for motion) and (0.98mm - 1.07mm for structure) (refer to Figure 3.) Second, the motion estimation seem to be more robust under noise than the structure estimation. Third, the none of four settings (described above) had any significant advantage in motion error. However, the structure error with curve segments were significantly lower than error with point features. Fourth, using both features did not seem to improve the results except some regions with high noise level.

5.2. Real Data

Four real data sequences have been used: (1) cup squeezing, (2) skin stretching, (3) mouth opening, and (4) wrist bending (Figure 2.) The 2D projection of the recovered 3D point and curve features are marked white on the first image of each sequence for presentation. With the "Wrist Bending" sequence, the point features on the curves are extracted. The points are marked "black" for easier visualization. This dataset is to show that the algorithm can incorporate even the point features on the curves.

The statistics of the dataset is given in Table 1. Except for the face dataset, the number of features was far more than the minimum requirement of 3 (for 3 frames). The dataset consisting of 3 frames were used. However, the framework is easily expandable to the longer image sequences [2].

The Table 2 shows the results which are divided into four categories: P_only, P_both, C_only, C_both. The results between using one type of feature and two types are shown between (P_only & P_both) and (C_only & C_both). There were 16 such instances = 2 (curve and point) x 4 datasets x 2 (motion & structure).

First, the results (under P_both and C_both) show that the algorithm was able to incorporate using both features. The error was less than 2.5 mm (in average) in 13 out of 16 times. The average absolute error of four datasets was around 2mm. Considering that the range camera's accuracy is only up to 0.5mm, the error seems to be reasonable. The mouth data showed a greater error rate possibly due to a small number of features being used. Also, the wrist data contains some inconsistency on the range data due to the hair on the skin. Second, when both features are used, the results were more stable. In 6 out of 16 instances, the results were better using both features with the average improvement of 54%. In 6 other instances, the error increased by average of 8%. This indicates that the usage of both features improved the results when there was a large error.

6. Conclusions

This work presents a framework to integrate point and curve features for nonrigid motion recovery and 3D structure reconstruction. The results of synthetic data show that the addition of the curve features significantly improved the structure performance. The results from real data show that

	2D	3D	# of points	# of curves
cup	6.6	2.6	6	2
skin	23.4	7.4	6	3
face	31.8	11.1	3	1
wrist	13.1	7.0	9	3

Table 1. Real Data Description. Average Movement for each frame is shown. 2D in pixels and 3D in mm.

	P_only	P_both	C_only	C_both
cup	0.7	0.5	3.3	1.4
skin	4.0	1.5	1.4	1.5
face	1.9	2.1	3.8	3.9
wrist	1.9	2.4	2.1	2.4
mean	2.1	1.6	2.7	2.3

Motion Results (Units in mm)

cup	0.8	0.5	2.2	0.8
skin	6.6	2.0	1.1	1.1
face	4.1	4.1	2.2	2.2
wrist	2.9	3.0	1.6	1.6
mean	3.6	2.4	1.8	1.5

Structure Results (Units in mm)

Table 2. Results of Real Data. (Units in mm)

(1) the algorithm was able to incorporate the usage of both features, (2) the addition of the curve features reduced the error significantly when there was a large error.

7. Acknowledgments

This work was supported by the NSF grants IRI-9619240, CDA-9724422.

References

[1] J. K. Aggarwal and Q. Cai, "Human motion analysis: A review", IEEE Nonrigid and Articulated Motion Workshop, pg 90-102, June 1997.

[2] R. Balasubramanian, D. Goldgof, C. Kambhamettu, "Tracking of nonrigid motion and 3D structure from 2D image sequences without correspondences." *Proceedings International Confererence on Image Processing, Chicago, USA* 1:933-937, October 1998.

[3] F. Hill, *Computer Graphics*. Macmillan Publishing Company, New York, 1990.

[4] Eric Petajan, "Facial Animation Coding Unofficial Derivative of MPEG-4 Standardization," ISO/IEC/JTC1/SC29/WG11 Face and Body Animation Ad Hoc Groupo, Sept 1997.

[5] L. Piegl and W. Tiller, *The NURBS Book*. Springer-Verlag, New York, second edition, 1997.

[6] H. Tao and T. S. Huang, "Explanation-based facial motion tracking using a piecewise Bezier volume deformation model", *Proceedings of Computer Vision and Pattern Recognition, Fort Collins, CO*, I:611-617, 1999.

[7] L. Tsap, D. Goldgof, S. Sarkar, P. Powers, "A Vision-Based Technique for Objective Assessment of Burn Scars", *IEEE Transactions on Medical Imaging*, 17(4), pp. 620-633, 1998.