Structure and Motion from Images of Smooth Textureless Objects

Yasutaka Furukawa[1], Amit Sethi[1], Jean Ponce[1], and David Kriegman[2]

[1] Beckman Institute, University of Illinois at Urbana-Champaign
{yfurukaw,asethi}@uiuc.edu, ponce@cs.uiuc.edu
[2] Dept. of Computer Science, University of California at San Diego
kriegman@ucsd.edu

Abstract. This paper addresses the problem of estimating the 3D shape of a smooth textureless solid from multiple images acquired under orthographic projection from unknown and unconstrained viewpoints. In this setting, the only reliable image features are the object's silhouettes, and the only true stereo correspondence between pairs of silhouettes are the *frontier points* where two viewing rays intersect in the tangent plane of the surface. An algorithm for identifying geometrically-consistent frontier points candidates while estimating the cameras' projection matrices is presented. This algorithm uses the *signature* representation of the dual of image silhouettes to identify promising correspondences, and it exploits the redundancy of multiple epipolar geometries to retain the consistent ones. The visual hull of the observed solid is finally reconstructed from the recovered viewpoints. The proposed approach has been implemented, and experiments with six real image sequences are presented, including a comparison between ground-truth and recovered camera configurations, and sample visual hulls computed by the algorithm.

1 Introduction

Structure and motion estimation algorithms typically assume that correspondences between *viewpoint-independent* image features such as interest points or surface markings have been established via tracking or some other mechanism (e.g., [4,21,23]). Several effective techniques for computing a projective, affine, or Euclidean scene representation from these correspondences while estimating the corresponding projection matrices are now available (see, for example [8,9, 13] for extensive discussions of such methods). For objects with little texture and few surface markings, silhouettes are the most reliable image features. The silhouette of a smooth solid is the projection of a surface curve, the *occluding contour*, where the viewing cone grazes the surface. Establishing correspondences between these *viewpoint-dependent* features is difficult: In fact, there is only a finite number of true stereo correspondences between any two silhouettes, namely the *frontier points* where the two occluding contours and the corresponding viewing rays intersect in the tangent plane of the surface [10].

For image sequences taken by a camera with known motion, it is possible to estimate the second-order structure of a surface along its occluding contour,

T. Pajdla and J. Matas (Eds.): ECCV 2004, LNCS 3022, pp. 287–298, 2004.

as first shown by Giblin and Weiss in the orthographic projection case [12] (see, for example, [5,7,20] for extensions to perspective projection). Methods for recovering both the surface structure and the camera motion using a trinocular rig have also been proposed [14,25]. The single-camera case is more difficult, and all approaches proposed so far have either been limited to circular motions [11, 18,28], required a reasonable guess to bootstrap an iterative estimation process [2,6], or been limited to synthetic data [26]. Likewise, all published methods for computing visual hulls [16] from image silhouettes, dating back to Baumgart's 1974 thesis [3], have assumed that the camera configurations were known a priori.

This paper presents an integrated approach to the problem of estimating both structure and motion for smooth textureless solids observed by orthographic cameras with unknown and unconstrained viewpoints. An algorithm for identifying geometrically-consistent frontier point candidates while estimating the cameras' projection matrices is presented. This algorithm uses the *signature* representation of the dual of image silhouettes, proposed in [1] in the object recognition context, to identify promising correspondences, and it exploits the redundancy of multiple epipolar geometries [17] to retain the consistent ones. The visual hull [3,16] of the observed solid is finally reconstructed from the recovered viewpoints. We have implemented this algorithm, and tested it on six real image sequences.

2 Proposed Approach

As mentioned in the previous section, the only true stereo correspondences between two silhouettes of a smooth solid are a finite number of *frontier points*, where two viewing rays intersect as they graze the surface along the same tangent plane (Figure 1). Equivalently, the frontier points are the intersections of the corresponding occluding contours on the surface.

As will be shown in Section 2.2, it is a relatively simple matter to estimate the projection matrices associated with m views of a smooth surface when a sufficient number of *true* frontier points are available for a sufficient number of image pairs. Conversely, it is easy to find the frontier points associated with a pair of images once the corresponding projection matrices are known since the corresponding tangent lines run parallel to the epipolar lines. This suggests the following algorithm for robustly estimating the projection matrices while identifying correct matches between silhouette pairs. It is similar in spirit to the RANSAC-based approach to weak calibration proposed in [22].

1. For each image pair, select a set of promising frontier points candidates. Each candidate will be referred to as a *match* between the two images in the sequel.
2. Find a minimal set of images and geometrically-consistent matches, and estimate the corresponding pairwise epipolar geometries and the individual projection matrices;
3. Add the remaining images one by one, using matches that are geometrically consistent with the current set of images to estimate the corresponding projection matrices.

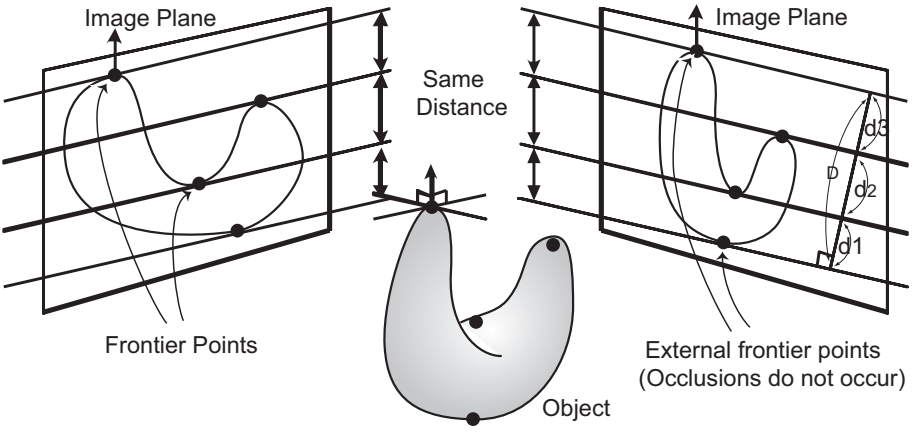


Fig. 1. Frontier points. See text for details.

Three main ingredients play a role in the successful implementation of this algorithm—namely, effective techniques for (1) selecting promising matches between pairs of images; (2) estimating the projection matrices from these matches; and (3) rejecting matches that are not consistent with all available geometric information. These ingredients are detailed in the following sections.

2.1 Selecting Frontier Point Candidates

A fundamental property of frontier points under orthographic projection is that the tangent lines at these points are parallel to each other, and the distances between successive tangents are the same in the two images. This property was used in [1] as the basis for a 3D object recognition algorithm. Briefly, the *signature* of a planar curve Γ is defined by mapping every unit vector $\boldsymbol{n}$ in the plane onto the tuple formed by the successive distances between the tangent lines to Γ perpendicular to $\boldsymbol{n}$ (Figure 1), taken in the order in which they are traversed by that vector. Formally, the signature can be thought of as a representation of the set of tangent lines—or *dual*—of Γ by a family of curves embedded in subspaces of $\mathbb{R}^d$ of various dimensions, where d is the maximum number of parallel tangents of Γ [1]. In the structure-from-motion context, this interpretation is not necessary. Instead, it is sufficient to note that the signatures of two silhouettes intersect at the corresponding frontier points, which affords a simple mechanism for selecting potential pairs of frontier points.

To account for the possibility of self occlusion, we follow the robust matching approach of [1,24] to determine the "distance" between two signature points $\boldsymbol{d} = (d_1, \ldots, d_k)$ and $\boldsymbol{d}' = (d'_1, \ldots, d'_l)$, where k may not equal l. Assuming that $d_{ij} = |d_i - d'_j|$ obeys a normal distribution with variance σ for matching entries, and a uniform distribution for all others, the discrepancy between individual entries in $\boldsymbol{d}$ and $\boldsymbol{d}'$ is the Lorentzian $L_\sigma = \sigma^2/(d_{ij}^2 + \sigma^2)$, whose value is 1 for a perfect match but is close to zero for large mismatches. To respect the natural ordering of the tangent lines, the final score is found by using dynamic

programming to maximize the sum of the Lorentzians among all paths with non-decreasing function $j(i)$, and dividing the maximum by the number of matched signature points.

This approach provides a guide for selecting promising matches. We also use a number of filters for rejecting incorrect ones: First, the object should lie on the same side of matching tangents in both images. Second, the curvatures at matching frontier points should have the same sign [15]. In practice, we exhaustively search each pair of silhouettes for potential sets of frontier points,[1] and retain the t most promising ones, where t is a fixed constant ($t = 10$ in our implementation).

2.2 Estimating Projection Matrices from Frontier Points

We assume an affine (orthographic, weak-perspective, or para-perspective) projection model, and show in this section how to estimate the projection matrices associated with a set of cameras from the corresponding object silhouettes and their pairwise frontier points. Contrary to the typical situation in structure from motion, where many point features are visible in many images, a (relatively) small set of frontier points is associated with each pair of images, and it is only visible there. Therefore, a different approach to motion estimation is called for. We proceed in three steps as described below.

Affine motion from a pair of images. Exploiting the affine ambiguity of affine structure from motion allows us to write the projection matrices associated with two images I and I' in the *canonical form* (see [9] for example):

$$\hat{\mathcal{M}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad \hat{\mathcal{M}}' = \begin{bmatrix} 0 & 0 & 1 & 0 \\ a & b & c & d \end{bmatrix}. \tag{1}$$

Assuming there are n frontier points with three-dimensional coordinates (x_j, y_j, z_j) and image coordinates (u_j, v_j) and (u'_j, v'_j) $(i = 1, \ldots, n)$, it follows immediately that

$$au_j + bv_j + cu'_j - v'_j + d = 0 \text{ for } j = 1, \ldots, n. \tag{2}$$

This is of course equivalent to the affine epipolar constraint $\alpha u_j + \beta v_j + \alpha' u'_j + \beta' v'_j + \delta = 0$, where the coefficients a, b, c, and d are related to the parameters α, β, α', β', and δ by $a : \alpha = b : \beta = c : \alpha' = -1 : \beta' = d : \delta$. Given the images of n frontier points, the parameters a, b, c, and d can be computed by using linear least squares to solve the over-constrained system of linear equations (2) in these unknowns.

[1] We could of course use some hashing technique—based, say, on the diameter D of the object in the direction of interest—to improve the efficiency of the search for promising matches, but this is far from being the most costly part of our algorithm.

Affine motion from multiple images. This section shows how to recover the m projection matrices $\mathcal{M}_i$ $(i = 1, \ldots, m)$ in some global affine coordinate system once the pairwise epipolar geometries are known, or, equivalently, once the projection matrices are known in the canonical coordinate systems attached to each camera pair.

Suppose that the values $(a_{kl}, b_{kl}, c_{kl}, d_{kl})$ associated with two images I_k and I_l have been computed from (2). There must exit some affine transformation $\mathcal{A}$ mapping the canonical form (1) onto $\mathcal{M}_k$ and $\mathcal{M}_l$, i.e.,

$$\begin{bmatrix} \mathcal{M}_k \\ \mathcal{M}_l \end{bmatrix} = \begin{bmatrix} \hat{\mathcal{M}}_k \\ \hat{\mathcal{M}}_l \end{bmatrix} \mathcal{A}. \tag{3}$$

If we write the two projection matrices $\mathcal{M}_k$ and $\mathcal{M}_l$ as

$$\mathcal{M}_k = \begin{bmatrix} \boldsymbol{p}_k^T \\ \boldsymbol{q}_k^T \end{bmatrix} \quad \text{and} \quad \mathcal{M}_l = \begin{bmatrix} \boldsymbol{p}_l^T \\ \boldsymbol{q}_l^T \end{bmatrix},$$

it is a simple matter to eliminate the unknown entries of $\mathcal{A}$ in Eq. (3) and show that

$$\boldsymbol{q}_l = \begin{bmatrix} \boldsymbol{p}_k & \boldsymbol{q}_k & \boldsymbol{p}_l & \mathbf{0} \end{bmatrix} \boldsymbol{e}_{kl},$$

where $\mathbf{0} = (0, 0, 0, 1)^T$, and $\boldsymbol{e}_{kl} = (a_{kl}, b_{kl}, c_{kl}, d_{kl})^T$. In other words, we have four linear constraints on the entries of the matrices $\mathcal{M}_k$ and $\mathcal{M}_l$. By combining the equations associated with all image pairs, we obtain a linear system of $2m(m-1)$ linear equations in the $8m$ entries of the m projection matrices, whose solutions are only defined up to an arbitrary affine transformation. We remove this ambiguity by fixing two projection matrices to their canonical form given by (1). The solution of the remaining $p = 2m(m-1) - 4$ linear equations in $q = 8(m-2)$ unknowns is again computed by using linear least squares. Three images are sufficient to compute a single solution, and four images yield redundant equations that can be used for consistency checks as explained in the next section.

Euclidean motion. Let us write the affine projection matrices recovered in the previous section as $\mathcal{M}_i = \begin{bmatrix} \mathcal{A}_i & \boldsymbol{b}_i \end{bmatrix}$ $(i = 1, \ldots, m)$. As shown in [19] for example, once the affine projection matrices are known, there exists an affine transformation, or *Euclidean upgrade*,

$$\mathcal{Q} = \begin{bmatrix} \mathcal{C} & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix} \text{ such that } \mathcal{M}_i \mathcal{Q} = \begin{bmatrix} \mathcal{R}_i & \boldsymbol{b}_i \end{bmatrix},$$

where the 2×3 matrix $\mathcal{R}_i$ is the top part of a 3×3 rotation matrix and, this time, $\mathbf{0} = (0, 0, 0)^T$. It follows that $\mathcal{A}_i(\mathcal{C}\mathcal{C}^T)\mathcal{A}_i^T = \mathcal{A}_i \mathcal{S} \mathcal{A}_i^T = \mathrm{Id}_2$, where $\mathcal{S} = \mathcal{C}\mathcal{C}^T$, and Id_2 is the 2×2 identity matrix. The m instances of this equation provide $3m$ constraints on the 6 independent entries of the symmetric matrix $\mathcal{S}$, allowing its

recovery via linear least squares. Once $\mathcal{S}$ is known, the matrix $\mathcal{C}$ can be recovered using Cholesky factorization for example.[2]

2.3 Enforcing Geometric Consistency

As shown in [17] for example, the pairwise epipolar constraints among a set of images are redundant. We propose in this section to exploit this redundancy by enforcing the corresponding geometric constraints during matching.

Geometric consistency constraints. The following simple tests can be used to check whether a set of matches and the corresponding projection matrices are geometrically consistent:

1. *Motion estimation residuals.* As shown in Section 2.2, the recovery of the affine projection matrices from a set of frontier points can be formulated as a linear least-squares problem. The size of the corresponding residual gives a first measure of consistency. The same is true of the residual of the linear system associated with the corresponding Euclidean upgrade. We use both measures in our implementation as simple filters for rejecting incorrect matches.
2. *Unmatched external frontier points.* Suppose the projection matrices associated with m images have been estimated, but matches of some image pairs (I_k, I_l) have *not* been used in the estimation process (this is a typical situation because of the epipolar constraints' redundancy). The affine fundamental matrix associated with I_k and I_l is easily computed from the corresponding projection matrices, and it can be used to predict the frontier points' projections in both images. Due to noise, discretization errors, occlusions, etc., some of the predicted points in one image may not have matches in the other one. Still, the two outermost—or *external*—frontier points are normally visible in each image (Figure 1), even in the presence of self occlusion, and they can be used as a second consistency filter. Of course, the distance between these points should be the same in the two images, i.e., the diameters of the two silhouettes in the direction orthogonal to the epipolar lines should be the same. But one can go further and compute the distance separating each external frontier point from the epipolar line associated with its match. This test, that computes four images distances instead of a single diameter difference, has proven much more discriminative in our experiments.
3. *Matched frontier points.* Assuming as before that the projection matrices are known, the 3D positions of all matched frontier points are easily reconstructed via triangulation. Our third consistency check is to project these frontier points into every other image and see if they lie outside the corresponding silhouette. Sum of distances of outlying frontier points to the closest point on each silhouette becomes the measure.

[2] This assumes that $\mathcal{S}$ is positive definite, which may not be the case in the presence of noise. See [21] for another approach based on non-linear least squares and avoiding this assumption.

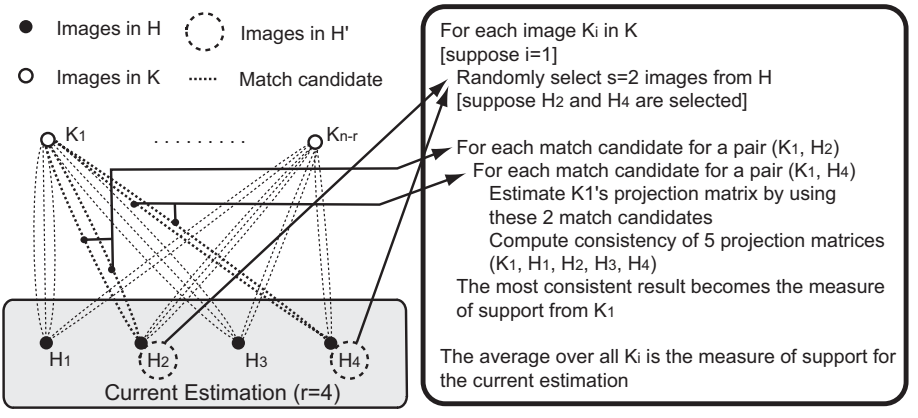


Fig. 2. A procedure for estimating how well r projection matrices are supported by all the other images in the bootstrapping process.

4. *Smooth camera motion.* When the input images are part of a video sequence, it is possible to exploit the continuity of the camera motion. In particular, we require the angle between the viewing directions associated with images number k and l to be less than $|k-l|$ times some predefined threshold d. We use $d = 10$ [degrees] in our experiments.

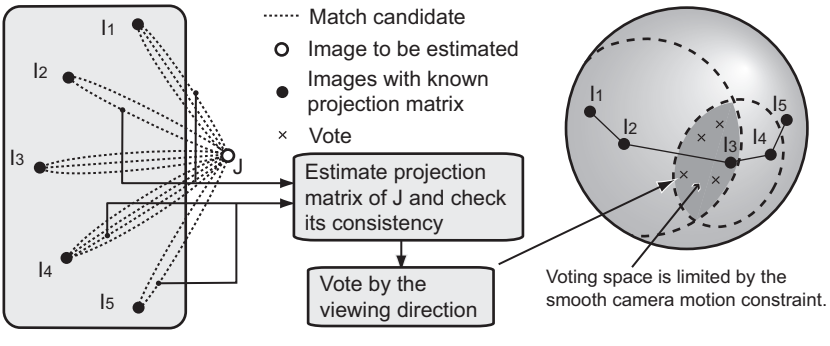


Fig. 3. Voting method to estimate a new projection matrix. Two match candidates are selected to cast a vote. When a camera motion is known to be smooth, the third consistency check method is applied and the voting space is limited to the intersection of circles.

Selecting consistent matches while estimating motion parameters. Let us show how to find geometrically consistent matches between image pairs while estimating the corresponding epipolar geometries as well as the individual projection matrices. As noted in Section 2.2, bootstrapping the process requires

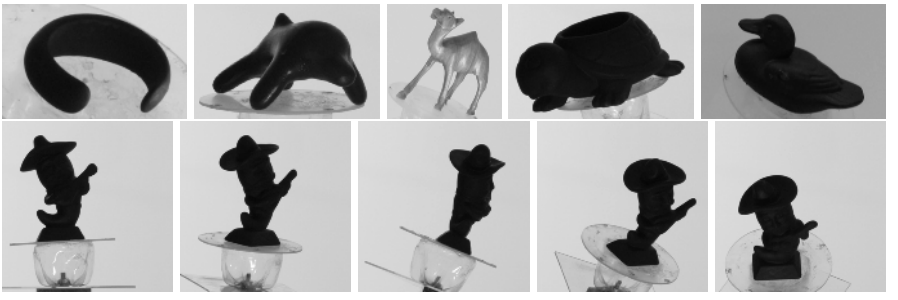


Fig. 4. Sample images of objects. The top row shows an image of a bracelet, a toy dolphin, a toy camel, a toy turtle, and a toy duck. The bottom row shows five images of a Mexican doll.

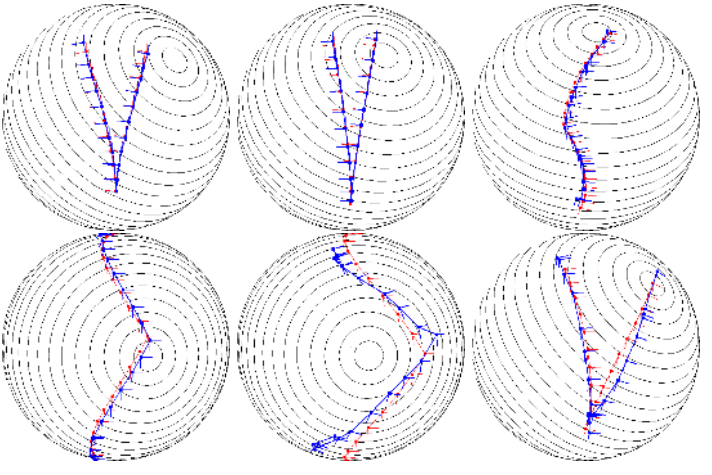


Fig. 5. In all the figures, thin lines represent ground truth data and thick lines represent our estimations. Top: recovered camera trajectories of bracelet, dolphin, and camel. Bottom: recovered camera trajectories of turtle, duck, and Mexican doll.

selecting $r \geq 3$ images from a total of n images and one match candidate for each one of the $\binom{r}{2} \geq 3$ corresponding image pairs. First, we randomly select r images $H = \{H_1, \ldots, H_r\}$ and try all promising matches among them to estimate r projection matrices. Second, we measure how well these estimates are supported by the other images $K = \{K_1, \ldots, K_{n-r}\}$. After repeating this process a fixed number of times, we finally report the set H of r images with maximum support as the winner.

Our measure of support is defined as follows (Figure 2): Suppose for a moment that $\binom{r}{2}$ match candidates have been used to estimate the projection matrices associated with the r images in H. For each image K_i in K, $s \geq 2$ images are randomly selected from H to estimate the projection matrix of K_i. Note that since the projection matrices associated with the elements of H are known, we only need to match K_i with $s \geq 2$ elements H' of H to estimate its projection

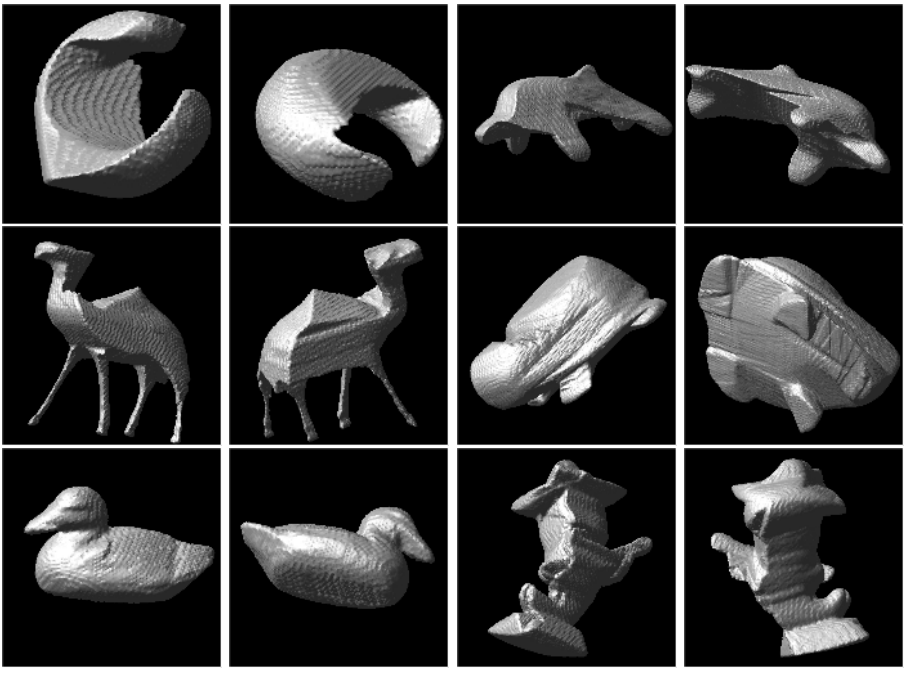
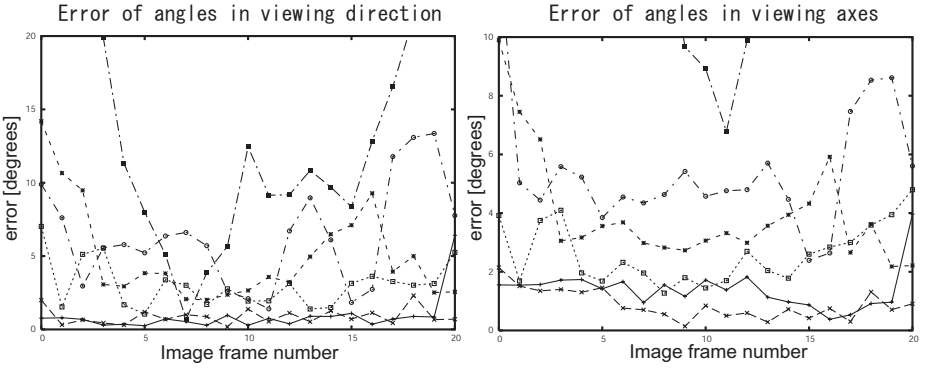


Fig. 6. Visual hull models constructed using the recovered camera projections.

matrix. For each image K_i and each element of H', we select one match candidate, estimate the projection matrix of K_i, and compute a consistency score by using the geometric constraints described above. This process is repeated for all tuples of match candidates between K_i and H', and we take the maximum consistency score as the measure of support $S(K_i)$ of the image K_i for H. The overall measure of support for H is computed as the average of the individual measures, or $\sum_{i=1}^{n-r} S(K_i)/(n-r)$. Next, we will describe how to estimate all the other $(n-r)$ projection matrices starting from the estimation of r projection matrices that has been just computed.

Let us assume from now on that the projection matrices associated with $m \geq r$ images $I = \{I_1, \ldots, I_m\}$ have been computed, and consider the problem of adding one more image J to I (Figure 3). We use a voting scheme to improve the matching reliability: We tessellate the unit sphere and represent each projection matrix by its viewing direction on the sphere. For all tuples I' of size s of images in I (again for the same reason as above, we need to match J with only $s \geq 2$ other elements for the estimation), we exhaustively choose a match candidate between J and each image in I', then estimate the projection matrix for J. Its consistency is checked by enforcing the four geometric constraints given above, and we cast a vote. The cell receiving the largest number of votes is declared as a winner and simply an average is taken in that cell to estimate the projection matrix of J. Note that the motion smoothness constraint can be incorporated in this scheme by limiting the voting space as an intersection of circles, centered



Mean and standard deviation of error of angles

Sequence		bracelet	dolphin	camel	turtle	duck	mexican
Error in viewing direction [degrees]	Mean	0.91	0.88	3.05	5.04	17.0	6.39
	Standard Deviation	1.26	0.53	1.53	3.24	14.0	3.46
Error in viewing axes [degrees]	Mean	1.40	0.89	2.61	3.98	26.0	5.50
	Standard Deviation	0.70	0.49	1.01	1.87	13.6	2.22

Fig. 7. Quantitative experimental results. Orientation errors in viewing directions and viewing axes are plotted for all the sequences. The mean and the standard deviation of these errors are also shown in the bottom table.

at viewing directions of each I_i, as shown in Figure 3. All images are added one by one to the set I by using this simple voting strategy repeatedly.

3 Implementation Details and Experimental Results

Six objects (a bracelet, a toy dolphin, a toy camel, a toy turtle, a toy duck, and a Mexican doll) have been used in our experiments. Each sequence consists of 21 images, which are acquired using a pan-tilt head providing ground truth for the viewing angles. Figure 4 shows one sample image for the first five objects, and five images for the Mexican doll to illustrate its complex shape.

Image contours are extracted with sub-pixel localization using B-spline snakes and gradient vector flow [27], while detecting corners. As discussed in the previous section, our algorithm first finds a set of r geometrically-consistent projection matrices by examining a subset of all the image tuples. The size of this subset has been set to 50 for all the examples. All other projection matrices are then estimated one by one. We exploit the smooth camera motion constraint for all the objects, using values of $r = 4$ and $s = 2$ in all cases.

Figure 5 compares the camera trajectories recovered by our algorithm to the ground-truth data from the pan-tilt head. In each case, the corresponding camera coordinate frames are first registered by a similarity transformation before being plotted on the unit sphere. As can be seen from the figure, estimated trajectories

are quite accurate, especially for the first four objects. As shown by Figure 6, the objects' visual hulls [3,16] are also recovered quite well. In fact, most inaccuracies are not so much due to errors in the recovered projection matrices as to the fact that a limited set of camera positions was used to construct each model.

Some quantitative results are given in Figure 7. The top two graphs show that errors tend to decrease in the middle of image sequences, which corresponds to intuition. As shown by the bottom table, rather large errors are obtained for the duck sequence. This is due do a few erroneous projection matrices at the beginning and the end of the sequence, with accurate estimates in its middle part.

References

1. Amit Sethi, David Renaudie, David Kriegman, and Jean Ponce. Curve and Surface Duals and the Recognition of Curved 3D Objects from their Silhouette. *Int. J. of Comp. Vision*, 58(1), 2004.
2. Kalle Åström and Fredrik Kahl. Motion estimation in image sequences using the deformation of apparent contours. *IEEE Trans. Patt. Anal. Mach. Intell*, 21(2):114–127, 1999.
3. B.G. Baumgart. Geometric modeling for computer vision. Technical Report AIM-249, Stanford University, 1974. Ph.D. Thesis. Department of Computer Science.
4. S. Birchfield. KLT: An implementation of the Kanade-Lucas-Tomasi feature tracker.
5. Edmond Boyer and Marie Odile Berger. 3d surface reconstruction using occluding contours. *Int. J. of Comp. Vision*, 22(3):219–233, 1997.
6. Roberto Cipolla, Kalle E. Åström, and Peter J. Giblin. Motion from the frontier of curved surfaces. In *Proc. Int. Conf. Comp. Vision*, pages 269–275, 1995.
7. Roberto Cipolla and Andrew Blake. Surface shape from the deformation of apparent contours. *Int. J. of Comp. Vision*, 9(2):83–112, 1992.
8. O. Faugeras, Q.-T. Luong, and T. Papadopoulo. *The Geometry of Multiple Images*. MIT Press, 2001.
9. D.A. Forsyth and J. Ponce. *Computer Vision: A Modern Approach*. Prentice-Hall, 2002.
10. P. Giblin and R Weiss. Epipolar curves on surfaces. *Image and Vision Computing*, 13(1):33–44, 1995.
11. Peter Giblin, Frank E. Pollick, and J. E. Rycroft. Recovery of an unknown axis of rotation from the profiles of a rotating surface. *Journal of Optical Society America*, pages 1976–1984, 1994.
12. Peter Giblin and Richard Weiss. Reconstruction of surface from profiles. In *Proc. Int. Conf. Comp. Vision*, pages 136–144, 1987.
13. R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, 2000.
14. Tanuja Joshi, Narendra Ahuja, and Jean Ponce. Structure and motion estimation from dynamic silhouettes under perspective projection. In *Proc. Int. Conf. Comp. Vision*, pages 290–295, 1995.
15. J.J. Koenderink. What does the occluding contour tell us about solid shape? *Perception*, 13:321–330, 1984.
16. A. Laurentini. How far 3D shapes can be understood from 2D silhouettes. *IEEE Trans. Patt. Anal. Mach. Intell.*, 17(2):188–194, February 1995.

17. Noam Levi and Michael Werman. The viewing graph. In *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pages 518–522, 2003.
18. Paulo Mendonca, Kwan-Yee K. Wong, and Robert Cipolla. Camera pose estimation and reconstruction from image profiles under circular motion. In *Proc. Euro. Conf. Comp. Vision*, pages 864–877, 2000.
19. C.J. Poelman and T. Kanade. A paraperspective factorization method for shape and motion recovery. *IEEE Trans. Patt. Anal. Mach. Intell.*, 19(3):206–218, March 1997.
20. Richard Szeliski and Richard Weiss. Robust shape recovery from occluding contours using a linear smoother. *Int. J. of Comp. Vision*, 28(1):27–44, 1998.
21. C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. *Int. J. of Comp. Vision*, 9(2):137–154, 1992.
22. P. Torr and D. Murray. The development and comparison of robust methods for estimating the fundamental matrix. *Int. J. of Comp. Vision*, 24(3), 1997.
23. P.H. Torr, A. Zisserman, and S.J. Maybank. Robust detection of degenerate configurations for the fundamental matrix. In *Proc. Int. Conf. Comp. Vision*, pages 1037–1042, Boston, MA, 1995.
24. P.H.S. Torr and A. Zisserman. Mlesac: A new robust estimator with application to estimating image geometry. *CVIU*, 78(1):138–156, 2000.
25. Régis Vaillant and Olivier D. Faugeras. Using extremal boundaries for 3-d object modeling. *IEEE Trans. Patt. Anal. Mach. Intell*, 14(2):157–173, 1992.
26. B. Vijayakumar, David J. Kriegman, and Jean Ponce. Structure and motion of curved 3d objects from monocular silhouettes. In *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pages 327–334, 1996.
27. Yue Wang, Eam Khwang Teoh, and Dinggang Shen. Structure-adaptive b-snake for segmenting complex objects. In *IEEE International Conference on Image Processing*, 2001.
28. Kwan-Yee K. Wong and Robert Cipolla. Structure and motion from silhouettes. In *Proc. Int. Conf. Comp. Vision*, pages 217–222, 2001.