3D Photography from Photographs and Video Clips

Jean Ponce[(1)]

Frederick Rothganger[(1)], Svetlana Lazebnik[(1)], Kenton McHenry[(1)]

Cordelia Schmid[(2)], Shyjan Mahamud[(3)], Martial Hebert[(3)]

(1) Beckman Institute, University of Illinois, Urbana, IL 61801, USA

(2) INRIA Rhône-Alpes, 665 Avenue de l'Europe, 38330 Montbonnot, France

(3) Robotics Institute, Carnegie-Mellon University, Pittsburgh, PA 15213, USA

e-mail: {jponce,rothgang,slazebni,mchenry}@uiuc.edu
Cordelia.Schmid@inrialpes.fr
{mahamud,hebert}@cs.cmu.edu

Abstract

This paper addresses the problem of acquiring realistic visual models of the shape and appearance of complex three-dimensional (3D) scenes from collections of images, a process dubbed *3D photography*. We focus on three instances of this problem: (1) the image-based construction of *projective visual hulls* of complex surfaces from weakly-calibrated photographs; (2) the automated matching and registration of photographs of textured surfaces using affine-invariant patches and their geometric relationships; and (3) an approach to projective motion analysis and self-calibration explicitly accounting for natural camera constraints such as zero skew and capable of handling large numbers of images in an efficient and uniform manner. We also briefly discuss some related applications of oriented differential projective geometry to computer vision problems, including the determination of the ordering of rim segments in projective visual hull computation, and a purely projective proof of Koenderink's famous characterization of the local shape of visual contours.

1 Introduction

Recent advances in motion analysis and image-based modeling and rendering have led to a convergence between computer vision and computer graphics, and, to a limited extent, to the industrial deployment of this technology: For example, the Façade [24] approach to environmental modeling developed at UC Berkeley has been used in producing "The Matrix"; the virtualized-reality technology developed at Carnegie-Mellon University [53] has been used to enhance the broadcast of the 2000 edition of the Superbowl with Matrix-like effects; and computer vision techniques are now routinely used in film production. Several key ingredients are still missing, however, before 3D photography fulfills its potential

in film, game, and Web-content production, TV advertising and sport-event broadcasts, electronic commerce, teleconferencing, human-computer interaction, and architectural and archaeological walkthroughs. Truly flexible modeling environments, capable of acquiring realistic models of dynamic scenes under general illumination patterns without cumbersome devices for geometric and photometric calibration must be developed.

The output of today's approaches to structure from motion (SFM) is essentially an unstructured cloud of geometric features such as points or lines [28, 32, 46, 117]. This is appropriate for rendering scene models from a limited set of viewpoints (near-frontal views where the Delaunay triangulation of an image's features are used to construct a surface mesh [94]) but not for 360° viewing. In addition, current SFM techniques may fail on surfaces with little or no texture, and they are limited by their computational coast to relatively small numbers of images and features [120]. On the other hand, 3D photography methods using as input a small set of *registered* pictures output polyhedral and/or volumetric models that are appropriate for rendering [22, 53, 58, 76, 77, 109, 122], but they require carefully setting up and calibrating the cameras, and they fail to record fine surface detail when changes between successive viewpoints are too large for correlation-based stereo [25, 35, 54, 86] to be effective. Neither class of approaches is particularly appropriate for modeling articulated or deformable objects.

We focus in the rest of this paper on three instances of the 3D photography problem: [1] the image-based construction of topological mesh models of complex surfaces from weakly-calibrated photographs (Section 2); [2] the automated matching and registration of photographs of textured surfaces taken from very different viewpoints (Section 3); and [3] the development of projective and Euclidean structure-from-motion techniques capable of handling large numbers of images in an efficient and uniform manner (Section 4). The proposed approach is based on several key ideas, including: [a] a novel image-based representation of object shape, topology and photometry in terms of *projective visual hulls* [63, 65]; [b] a novel representation for rigid and articulated textured surfaces in terms of *affine-invariant patches* and their spatial relationships [99]; and [c] an efficient, provably-convergent approach to projective SFM and self-calibration explicitly accounting for natural camera constraints such as zero skew [75, 95]. Our practical goal is to integrate these ideas into a system capable of acquiring realistic visual models of rigid and articulated objects from photographs and video sequences in controlled (blue screen) and uncontrolled (natural) environments. From a more theoretical point of view, our work on projective visual hulls has also led us to investigate the role of *oriented differential projective geometry* [60, 107] in computer vision and computer graphics. We will briefly discuss preliminary results in Section 2.

1.1 Related Work

A number of methods are available for constructing polyhedral meshes and deformable surfaces from range images [12, 22, 31, 115, 114, 122], the main challenge in this case being to fuse and register data extracted from multiple images [8, 50, 122]. Parametric object models defined in terms of superquadrics [3, 38, 90], algebraic surfaces [55, 110, 111, 112], and assemblies of simple volumetric primitives [17, 24] can be constructed from range images or registered photographs. Stereo pairs or triples of images can also be used to acquire polyhedral object models. In this context, the main difficulty is to establish correspondences between pictures: Feature-based techniques [2, 37, 88] and correlation-based approaches [25, 54, 35, 86] work well when the separation between the cameras is small. The wide-baseline case is more difficult [96], but prior shape models can help: For example, the Façade system of Debevec *et al.* [24] uses an approach—dubbed *stereo from shape* from now on—where the large disparities corresponding to gross surface structure are essentially zeroed by intersecting the visual rays from an *offset* image with the model's surface, then projecting the intersections into a *reference* image. The remaining disparities typically correspond to the fine structure of the observed scene, allowing once again the use of correlation techniques. Various types of local viewpoint invariants have also recently been proposed to establish correspondences in wide-baseline stereo [80, 113, 123]. We will revisit those as well as stereo from shape in latter parts of this paper. When the input pictures are registered, an alternative to conventional stereopsis is to delineate the outline of the object of interest in each image, and use the registered image contours to reconstruct an approximation of its surface, known as the *visual hull* [61] and formed by intersecting the viewing cones formed by the rays passing through the optical centers of the cameras and the corresponding image silhouettes. Algorithms for constructing visual hulls from images date back to the mid-70s and Baumgart's PhD thesis [5], and variants include [18, 76, 77, 85, 106]. We have used the fact that the viewing cones should be tangent to the surface to construct smooth surface models from visual hulls in [109]; an alternative that does not require full camera calibration will be proposed in Section 2 (see [76, 77] for related work). The *space-carving* approach proposed by Kutulakos and Seitz [58] is related to both stereo and visual-hull algorithms, and it uses registered photographs to construct discrete volumetric models, whose boundary voxels record color information. When continuous image sequences are available, other techniques can be used as well [10, 15, 16, 21, 52, 109, 125]. In the absence of registration information, SFM techniques [27, 44] estimate both the shape of the observed object and the motion of the camera observing it. Popular approaches include the affine factorization method of Tomasi and Kanade [117] and the projective reconstruction techniques pioneered by Faugeras [28] and

Hartley *et al.* [46]. The latter typically rely on the multilinear constraints associated with the *fundamental matrix* [71, 130] or the *trifocal tensor* [42, 103] to reconstruct the scene up to a projective transformation (see also [74, 108] for projective factorization methods that will be discussed in more detail later in this paper). This reconstruction is then refined using bundle adjustment [120] before being upgraded to a Euclidean one via self calibration [30, 78, 93, 121] using prior knowledge of camera parameters such as image center or focal length. Fitzgibbon and Zisserman [32] and Pollefeys *et al.* [94] describe complete systems capable of automatically acquiring Euclidean models of complex natural scenes.

The techniques discussed until now construct an explicit 3D object model from images. Recent work has demonstrated the possibility of synthesizing new views of 3D scenes *without* 3D reconstruction, a process dubbed *image-based rendering* [79]. Gortler *et al.* [36] and Levoy and Hanrahan [66] have used the fact that the set of all visual rays (*light field*) is four-dimensional to assemble new images from radiance information collected from a two-dimensional sample of images of a scene (see also [14, 104] for methods using mosaics to generate novel images). In contrast, the techniques proposed by Laveau and Faugeras [62], Seitz and Dyer [102], Kutulakos and Vallino [59] and Avidan and Shashua [1] rely on feature correspondences established across a discrete and usually small set of views. They are related to the problem of *transfer* in photogrammetry: Given the image positions of *tie points* in a set of reference images and in a new image, and given the image positions of a *ground point* in the reference images, predict the position of that point in the new image [4]. In the projective case, Laveau and Faugeras [62] have proposed to first estimate the fundamental matrix associated with each pair of reference views, then reproject the scene points into a new image by specifying the position of the new optical center in two reference images and the position of four reference points in the new image.[1] Once the feature points have been reprojected, realistic rendering is achieved using ray tracing and texture mapping. Related methods have been proposed by various authors in the affine and projective cases [1, 59, 102]. Their main drawback is that the synthesized images are in general separated from the "correct" ones by arbitrary planar affine or projective transformations. The methods proposed by Avidan and Shashua [1] and Genc and Ponce [34] overcome this difficulty by taking into account the constraints associated with calibrated cameras from the start.

We have focused in this section on the geometric aspects of 3D photography. It should

[1]Notably, this work involves the first explicit use of *oriented projective geometry* [107] in computer vision, allowing the distinction between points that are in front of a camera, and those that are behind (see also [43]). We will come back to oriented projective geometry in Section 2, where it will play a fundamental role in the construction of projective visual hulls.

be noted that a number of approaches to the construction of accurate photometric scene models have also been proposed. These include methods for viewpoint-dependent texture mapping [24, 76, 77, 97], and techniques for constructing surface light [81, 129] and reflectance [23, 73, 87] fields. We will revisit those in Section 5.

2 Object Modeling from Shape Cues

Since contour rotoscoping is a common operation in modern film and television production pipelines, and commercial packages are available to smaller-scale content creators, the visual hulls briefly discussed in Section 1.1 are an attractive means for capturing the overall structure of surface models in many applications, especially when used in combination with texture mapping. We introduced in [109] a method for automatically acquiring 3D models from objects' silhouettes found in a few registered photographs, where a polyhedral visual hull is used to construct a smooth triangular spline surface, which is then deformed until it is tangent to all viewing cones. Although this method gives satisfactory results, it requires computationally expensive and possibly fragile algorithms for computing the intersection of polyhedral solids, as well as precise calibration data for all input cameras.

We propose here a new extension of the visual hull, the *projective visual hull*, which consists of two graphical structures, dubbed the *rim mesh* and *visual hull mesh* [63, 64], and can be constructed *directly* from weakly-calibrated image data, without any explicit 3D reconstruction or precise knowledge of the cameras' positions or intrinsic parameters (see [76, 77] for related work). Let us define the *occluding contour*, or *rim* as the curve where the viewing cone associated with an object's silhouette grazes its surface (Figure 1). The rim mesh is a boundary representation of the surface: Its vertices are the *frontier points* [15, 98] where rims associated with two images intersect; its arcs are the occluding contour branches joining successive pairs of frontier points along the same rim, and its faces are the surface patches delimited by these curves. Likewise, the visual hull mesh is a boundary representation of the visual hull: Its faces are the parts of the viewing cone surfaces—called *strips* because they bound the regions of space where the corresponding rim branches may lie—that actually belong to the visual hull; its vertices are frontier points (where two strips cross) and *triple points* (where three viewing cones intersect), and its edges are intersection curve segments between consecutive vertices.

The key insight is that both the rim and visual hull meshes are actually *projective* structures—that is, their topology is invariant under projective transformations. Intuitively, this is simply due to the fact that these transformations preserve the incidence relationship and order of contact between rays and surfaces. In particular, graphical

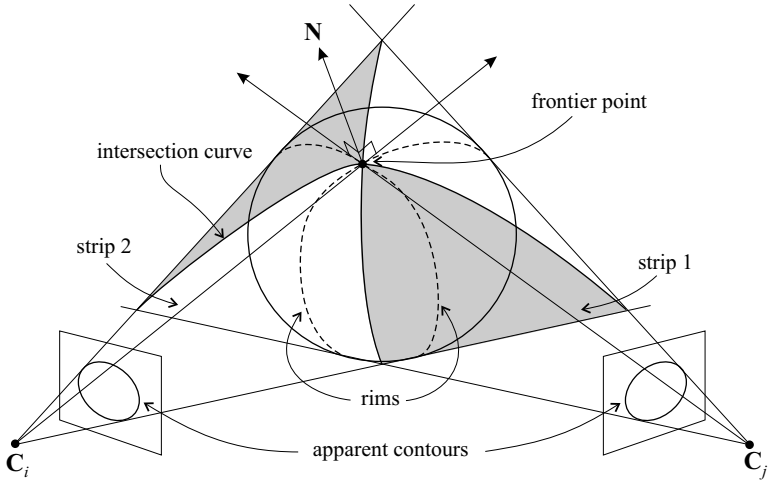

Figure 1: Left: A solid observed by two cameras. The two rims intersect at the frontier point with the intersection curves of the two viewing cones. The faces of the visual hull form strips enclosing the rim arcs.

descriptions of the two meshes can in principle be constructed directly from image information when the epipolar geometry is known: As shown in [64], the 1-skeleton of the rim mesh—that is, the graph formed by its vertices (frontier points) and edges (rim arcs)—can be found by computing the (image) frontier points as the places where tangent epipolar rays graze the contour [15, 98], then inserting these points in the correct order along the contour. We focus here in the construction of the visual hull mesh [63, 65]. Its 1-skeleton can be constructed using transfer to trace the projection of the intersection curves of the viewing cones into the input images, and inserting frontier and triple points as vertices along the contour and the intersection curves. Although point insertion and the subsequent construction of the faces of the visual hull mesh are relatively simple when Euclidean calibration information is available, these processes require orienting matching epipolar lines in a consistent way and identifying the convex and concave parts of the contour [63, 64]. Unfortunately, neither of these notions makes sense in the usual context of projective geometry. This has prompted us to investigate *oriented projective geometry* [107]. Recall that an ordinary projective space is the quotient of a vector space under the equivalence relation defined by $\boldsymbol{u} \approx \boldsymbol{v}$ when there exists some nonzero scalar λ such that $\boldsymbol{v} = \lambda\boldsymbol{u}$ [7]. In contrast, an oriented projective space is the quotient of a vector space under the equivalence relation $\boldsymbol{u} \approx \boldsymbol{v}$ when there exists some *positive* scalar λ such that $\boldsymbol{v} = \lambda\boldsymbol{u}$ [107]. Line orientation and convexity can be given proper definitions in this setting, that can also be used in projective motion analysis to distinguish valid point reconstructions lying *in front* of the cameras from incorrect ones lying *behind* at least one of the cameras [43, 62].

An additional difficulty in our case is the necessity of defining a local criterion for convexity [63]. In the Euclidean case, such a criterion is that the curvature at a point be positive. Curvature is not defined in (oriented) projective geometry, which has prompted us to investigate *differential* oriented projective geometry [60]. Briefly, it turns out that a differential oriented projective invariant, akin to curvature, can be properly defined in terms of first and second derivatives of a curve parameterization. Its value itself is meaningless, but its sign determines whether the curve is locally convex or concave at a point. Likewise, another invariant (akin but of course not identical to Gaussian curvature) can be defined for surfaces, and used to determine whether a surface patch is locally convex, concavem or saddle-shaped. Armed with these invariants, it is possible to prove the following result [63, 65]:

Proposition 1 *A convex (resp. convave, inflection) point on the apparent contour of a smooth solid is the projection of a convex (resp. hyperbolic, parabolic) point on the rim of its surface (Figure 2).*

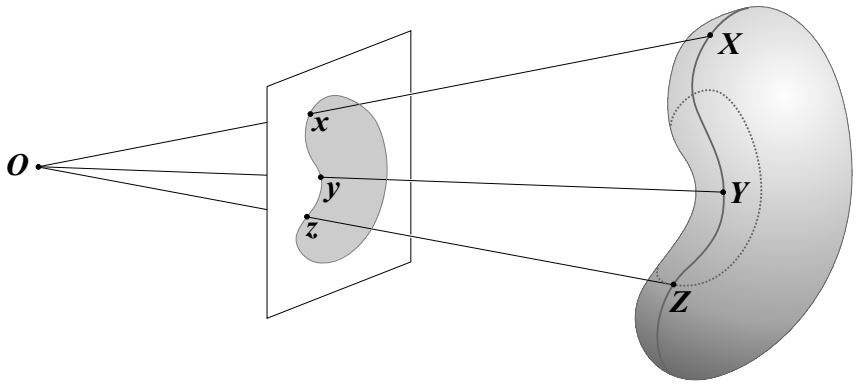

Figure 2: A smooth solid and its perspective projection. The rim is the solid curve drawn on the surface. The apparent contour is the boundary of the projection. The dashed curve is the locus of the parabolic points, or *parabolic curve*. The rim points X, Y, and Z are respectively convex, hyperbolic, and parabolic, and their images x, y, and z are respectively convex, concave, and inflection points of the contour.

This proposition was originally proven by Koenderink [56] using Euclidean concepts such as the curvature of plane curves and the Gaussian curvature of surfaces. In contrast, the elementary proof presented in [63, 65] is purely projective. Other classical results can also be generalized to the projective setting: For example (with proper orientations chosen for all objects involved), it is easy to show [63, 65] that a counterclockwise change in viewpoint in the tangent plane at a surface point X will cause the rim tangent to rotate counterclockwise when X is elliptic, and clockwise when it is hyperbolic (Figure 3). The following result is a corollary of this fact and Proposition 1 [63, 65].

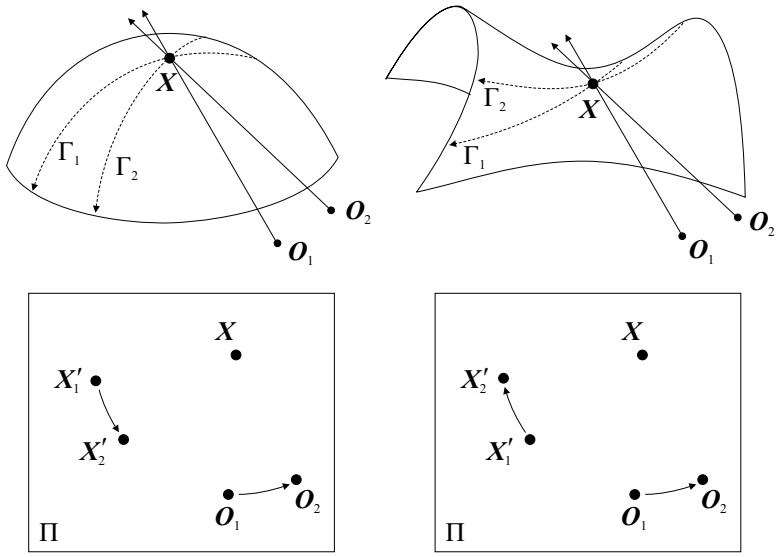

Figure 3: Relative orientation of rims and camera centers for an elliptic point (left), and a hyperbolic point (right).

Proposition 2 *It is possible to determine the relative orientation of the two rim branches intersecting at a frontier point from image information alone—namely, from the first and second derivatives of some parameterization of the visual contour at the image of the frontier point, and from the orientation of the corresponding epipolar tangents.*

With this proposition, it is possible to design a line-sweep algorithm for tracing the intersection curves and finding the frontier and triple points, correctly inserting these points along the contours and intersection curves, and determining the faces of the rim and visual hull meshes [63]. Figure 4 shows the results of a preliminary experiment, where silhouettes extracted by hand from 11 moderate-resolution (4 Mpixels) images, kindly provided by S. Sullivan and Industrial Light & Magic, have been used to construct the visual hull of a person.

3 Object Modeling from Texture Cues

The approach described in the previous section relies on shape information alone to create 3D models of rigid objects. In this section, we combine geometry with texture information in the modeling process. We use an implementation of the affine-invariant region detector proposed by Mikolajczyk and Schmid [80] to capture local appearance information (see Lindeberg and Gårding [67, 68] for related work). In this approach, the dependency of an image patch's appearance on affine transformations is eliminated by an iterative rectification process using (a) the second-moment matrix computed in the

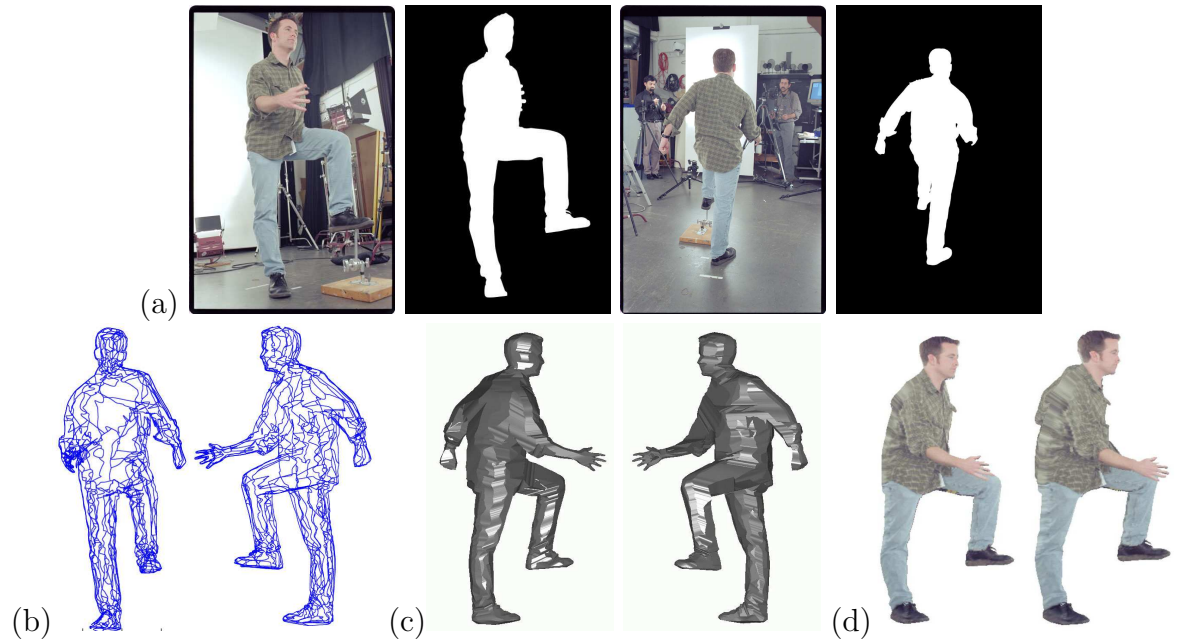

Figure 4: Projective visual hulls: (a) Sample pictures and silhouettes in the sequence (note that there are holes in the second silhouette); (b) the 1-skeleton of the visual hull mesh; (c) a triangulation of the visual hull mesh; (d) two texture-mapped views of this triangulation.

neighborhood of a point to normalize the shape of the corresponding image patch in an affine-invariant manner; (b) the local extrema of the normalized Laplacian over scale to determine the characteristic scale of the local brightness pattern; and (c) an affine-adapted Harris detector to determine the patch location. The output of the affine-invariant region detection/rectification process is a set of image patches in the shape of ellipses, together with the (affine) transformation mapping these ellipses onto a unit circle centered at the origin. This transformation is only defined up to a rotational ambiguity (this is intuitively obvious since a planar affine transformation is defined by six independent parameters but an ellipse is only defined by five parameters). We use image gradient information to eliminate this ambiguity. This allows us to turn the shape of an affine-invariant patch from an ellipse to a parallelogram, and to determine the six degrees of freedom of an affine *rectifying transformations* $\mathcal{R}$ that maps this corresponding parallelogram onto a square with unit edge half-length centered at the origin (Figure 5).

The rectified patch is a *normalized* representation of the local surface appearance that is invariant under planar affine transformations. We will assume from now on an affine—that is, orthographic, weak-perspective, or paraperspective—projection model. Under this model, our normalized appearance representation is invariant under arbitrary changes in viewpoint. For Lambertian patches and distant light sources, it can also be made invariant to changes in illumination (ignoring shadows) by subtracting the mean patch intensity

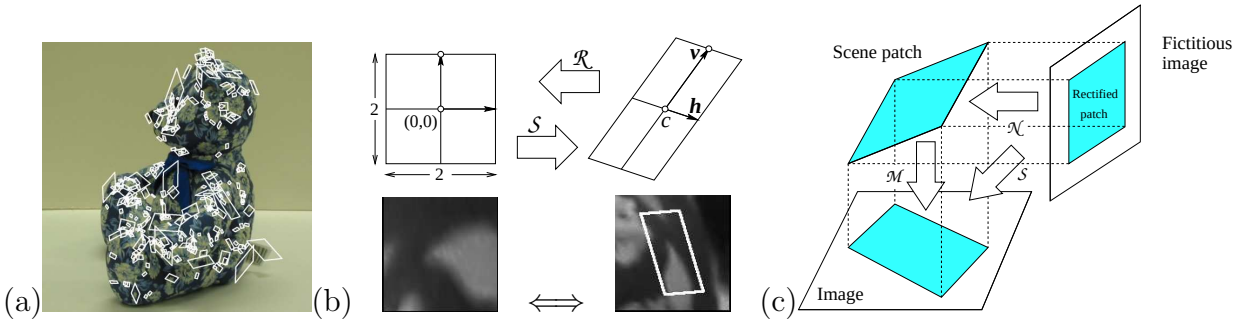

Figure 5: Affine-invariant patches: (a) Sample image and its patches; (b) the patch normalization process and its (two-dimensional) geometric interpretation; (c) a three-dimensional interpretation of this process.

from each pixel value and normalizing the sum of squared intensity values to one (or equivalently using *normalized* correlation to compare patches).

The rectifying transformation associated with a planar patch and its inverse can be represented by two 2×3 matrices $\mathcal{R}$ and $\mathcal{S}$ that map homogeneous (affine) plane coordinates onto non-homogeneous ones (Figure 5[b]). These transformations play a fundamental role in the rest of this section. Let us first note that the columns vectors of the matrix $\mathcal{S}$ admit a simple geometric interpretation: Since they are respectively the images of the vectors $(1,0,0)^T$, $(0,1,0)^T$, and $(0,0,1)^T$ under that mapping, the third column $\boldsymbol{c}$ of $\mathcal{S}$ is the (non-homogeneous) coordinate vector of the patch center c, and its first two columns $\boldsymbol{h}$ and $\boldsymbol{v}$ are respectively the (non-homogeneous) coordinate vectors of the "horizontal" and "vertical" vectors joining c to the sides of the patch. These two vectors can also be interpreted as the positions of the points, dubbed *normalized side points* in the sequel, where the "horizontal" and "vertical" axes of a copy of the image patch placed at the origin pierce its right and top side. The second key (and new) insight is that a rectified patch can also be thought of as a *fictitious* view of the original surface patch (Figure 5[c]), and the inverse mapping $\mathcal{S}$ can thus be decomposed into an *inverse projection* $\mathcal{N}$ [27] that maps the rectified patch onto the corresponding surface patch, followed by a projection $\mathcal{M}$ that maps that patch onto its (true) image projection, i.e., $\mathcal{S} = \mathcal{M}\mathcal{N}$. Note that in the affine projection setting chosen here, we can write

$$\mathcal{M} = [\,\mathcal{A} \quad \boldsymbol{b}\,] \quad \text{and} \quad \mathcal{N} = \begin{bmatrix} \mathcal{B} \\ (0,0,1) \end{bmatrix},$$

where $\mathcal{A}$ and $\mathcal{B}$ are respectively 2×3 and 3×3 matrices, and $\boldsymbol{b}$ is a vector in $\mathbb{R}^2$.[2] The columns of the matrix $\mathcal{B}$ admit a geometric interpretation related to that of the matrix $\mathcal{S}$:

[2]This is an affine instance of the characterization of homographies induced by planes given in Faugeras, Luong and Papadopoulo [27, Prop. 5.1].

Namely, the first two are the (non-homogeneous) coordinate vectors of the "horizontal" and "vertical" axes of the surface patch, and the third one is the (non-homogeneous) coordinate vector of its center C.

In particular (and not surprisingly), a match between $m \geq 2$ images of the same affine-invariant patches contains *exactly* the same information as a match between m triples of points. It is thus clear that all the machinery of structure from motion from point matches [27, 44, 117] can be exploited in modeling tasks, the multi-view constraints associated with the matrix $\mathcal{S}$ providing a unified and convenient representation for all stages of the process. In particular, let us assume that we are given n patches observed in m images, together with the corresponding 2×3 matrices $\mathcal{R}_{ij}$ and $\mathcal{S}_{ij}$ for $i = 1, \ldots, m$ and $j = 1, \ldots, n$ (i and j serving respectively as image and patch indices). Following Tomasi and Kanade [117], we can take the center of mass of the observed patches' centers as the origin of the world coordinate system, and the center of mass of these points' projections as the origin of every image coordinate system. In this case, the vectors $\boldsymbol{b}_i$ are equal to zero, and we have $\mathcal{S}_{ij} = \mathcal{A}_i \mathcal{B}_j$, or equivalently,

$$\hat{\mathcal{S}} = \hat{\mathcal{A}}\hat{\mathcal{B}}, \quad \text{where} \quad \hat{\mathcal{S}} \stackrel{\text{def}}{=} \begin{bmatrix} \mathcal{S}_{11} & \ldots & \mathcal{S}_{1n} \\ \ldots & \ldots & \ldots \\ \mathcal{S}_{m1} & \ldots & \mathcal{S}_{mn} \end{bmatrix}, \; \hat{\mathcal{A}} \stackrel{\text{def}}{=} \begin{bmatrix} \mathcal{A}_1 \\ \vdots \\ \mathcal{A}_m \end{bmatrix}, \text{ and } \hat{\mathcal{B}} \stackrel{\text{def}}{=} [\, \mathcal{B}_1 \quad \ldots \quad \mathcal{B}_n \,].$$

In particular, $\hat{\mathcal{S}}$ has at most rank 3, a fact that can be used as a matching constraint when at least two matches are visible in at least two views. Alternatively, singular value decomposition can be used as in Tomasi and Kanade [117] to factor $\hat{\mathcal{S}}$ and compute estimates of the matrices $\hat{\mathcal{A}}$ and $\hat{\mathcal{B}}$ that minimize the squared Frobenius norm of the matrix $\hat{\mathcal{S}} - \hat{\mathcal{A}}\hat{\mathcal{B}}$. The normalized (residual) norm $|\hat{\mathcal{S}} - \hat{\mathcal{A}}\hat{\mathcal{B}}|/\sqrt{3mn}$ of this matrix can be interpreted geometrically as the root mean squared distance between the center and normalized side points of the patches observed in the image, and the center and normalized side points predicted from the recovered matrices $\hat{\mathcal{A}}$ and $\hat{\mathcal{B}}$. Again, two views of two matches are sufficient to bring this constraint to bear on the matching process.

Image matching requires two key ingredients: (a) A measure of appearance similarity between two images of the same patch, and (b) a measure of geometric consistency between n matches M_1, ..., M_n established across m images (a match is an m-tuple of image patches). For the former we use normalized correlation between rectified patches. For the latter, we use the method described in the previous section to estimate (when $m, n \geq 2$) the matrices $\hat{\mathcal{A}}$ and $\hat{\mathcal{B}}$, and define $d(M_1, \ldots, M_n) = |\hat{\mathcal{S}} - \hat{\mathcal{A}}\hat{\mathcal{B}}|/\sqrt{3mn}$ as a measure of consistency among matches. In our current implementation, we only match patches across pairs of images ($m = 2$), and follow a strategy similar to that used in the range data domain by Johnson and Hebert [51] with *spin images*. Given a patch in one image, we first

select its most promising matches in the second image based on normalized correlation of the rectified patches. We then discard the matches M such that the number of consistent matches M' (i.e., matches such that $d(M, M')$ is less than some preset threshold) is less than some fixed percentage of the total number of candidate matches. At this point, we find groups of consistent matches as follows: For each one of the surviving $p < n$ matches, we initialize the group G to that match M, we then find the match M' minimizing $d(G, M')$ (naturally defined as $d(M_1, \ldots, M_k, M')$ when $G = (M_1, \ldots, M_k)$). If $d(G, M')$ is small enough, we add M' to G and continue. This results in the construction of p groups. Finally, we discard the smallest groups, and the remaining matches are judged to be correct.

The proposed matching strategy can be used in modeling tasks to match successive pairs of views of the same object. When some of the patches are only observed in some of the frames (the usual case), the data can be split into overlapping blocks of two or more frames, using all the patches visible in all images of the same block to run the factorization technique, then using the points common to overlapping blocks to register the successive reconstructions in a common frame. In principle, it is sufficient to have blocks that overlap by four points. Once all blocks are registered, the initial estimates of the variables $\mathcal{M}_i$ and $\mathcal{N}_j$ can be refined through a few non-linear least-squares iterations. When three or more views are available, it is then a simple matter to compute the corresponding Euclidean weak-perspective projection matrices (assuming the aspect-ratios are known) and recover the Euclidean structure of the scene [91, 95, 117, 126]. Figure 6 shows a photograph of a teddy bear and the Euclidean model reconstructed from 14 images including that one.

(a)  (b) 

Figure 6: An object modeling experiment [99]: (a) one of the 15 input photographs, and (b) three views of the reconstructed model. Note that the photographs are taken from viewpoints that are too far apart for conventional correlation-based stereo to work.

Patch-based models like the one shown in Figure 6 are too rough for direct use in computer graphics applications. They are, on the other hand, sufficient for object recognition purposes, as demonstrated by Figure 7 [99]. We will briefly address in Section 5 the prob-

lem of combining this type of surface representation with visual hulls and SFM techniques to construct more realistic models.

Figure 7: Object recognition experiments [99]: (top) a photograph with the patches matched to the bear model overlaid, and this model in its estimated pose; and (bottom) recognition of a salt can and a plastic rubble stand model.

4 Object Modeling from Motion Cues

The approaches proposed in the previous sections are limited to controlled settings where a fixed set of cameras capture a few snapshots of a scene. This section addresses the more general (and challenging) case of dynamic image sequences. In this context, we will exploit the significant advances in motion analysis that have taken place in the past ten years [44, 27]. Keys to this progress have been the emergence of reliable interest-point detectors [41, 80] and feature trackers [69, 117, 118]; a shift from methods relying on a minimum number of images [124] to techniques using a large number of images [93, 117, 118]; and a vastly improved understanding of the geometric, statistical and numerical issues involved [28, 29, 45, 46, 44, 49, 57, 72, 93, 103, 105, 117, 118, 127].

Concretely, let us consider m perspective cameras with projection matrices $\mathcal{M}_i$ $(i = 1, .., m)$ observing n fixed points with homogeneous coordinate vectors $\boldsymbol{P}_j$ $(j = 1, .., n)$. We assume that point correspondences have been established and address in this section the classical SFM problem of recovering both the matrices $\mathcal{M}_i$ and the vectors $\boldsymbol{P}_j$ from the image positions $\boldsymbol{p}_{ij}$ of the point projections. This is a least-squares problem that can be

expressed as the minimization of $E_1 = \sum_{i,j} |\boldsymbol{p}_{ij} - \frac{1}{z_{ij}} \mathcal{M}_i \boldsymbol{P} j|^2$ or $E_2 = \sum_{i,j} |z_{ij} \boldsymbol{p}_{ij} - \mathcal{M}_i \boldsymbol{P} j|^2$ with respect to the parameters z_{ij}, $\mathcal{M}_i$ and $\boldsymbol{P}_j$, where z_{ij} is the depth of point number j relative to camera number i. The error E_1 measures the mean squared distance between observed and predicted image points, but its minimization involves the use of iterative non-linear least-squares techniques (*bundle adjustment*), with a cost of $O((m+n)^3)$ per iteration (more efficient techniques are available in *sparse* cases, i.e., when $m \ll n$ or $n \ll m$ [44]). Here we propose instead to minimize E_2, which is not geometrically as satisfying but will prove computationally convenient. This requires imposing some constraint on the unknowns since E_2 admits a trivial zero for $\mathcal{M}_i = 0$, $\boldsymbol{P}_j = 0$ and $z_{ij} = 0$ otherwise. Another expression for E_2 is obtained by introducing the data matrix [108, 117]:

$$\mathcal{D} \stackrel{\text{def}}{=} \begin{pmatrix} z_{11}\boldsymbol{p}_{11} & \dots & z_{1n}\boldsymbol{p}_{1n} \\ \dots & \dots & \dots \\ z_{m1}\boldsymbol{p}_{m1} & \dots & z_{mn}\boldsymbol{p}_{mn} \end{pmatrix} = \mathcal{M}\mathcal{P} \text{ where } \mathcal{M} \stackrel{\text{def}}{=} \begin{pmatrix} \mathcal{M}_1 \\ \dots \\ \mathcal{M}_m \end{pmatrix} \text{ and } \mathcal{P} \stackrel{\text{def}}{=} (\mathcal{P}_1, \dots, \mathcal{P}_n).$$

It follows immediately that $E_2 = |\mathcal{D} - \mathcal{M}\mathcal{P}|^2$, where "$|\ |$" denotes the Frobenius norm. Minimizing E_2 is thus equivalent to finding the parameters z_{ij}, $\mathcal{M}$ and $\mathcal{P}$ that minimize the Frobenius norm of the difference between $\mathcal{D}$ and $\mathcal{M}\mathcal{P}$. Sturm and Triggs [108, 119] have proposed constraining the columns of $\mathcal{D}$ to have unit norm and minimizing E_2 by alternating steps where $\mathcal{M}$ and $\mathcal{P}$ are estimated using singular value decomposition with steps where the columns of $\mathcal{D}$ are renormalized and used to compute the projective depths z_{ij}. Although this method gives good results in practice, there is no guarantee that it will converge because of the column renormalization step. This has motivated Mahamud and Hebert [74] to propose a variant where the minimization is done under the constraint that the vectors $(z_{ij}, .., z_{mj})$ (where $j = 1, .., n$) have unit norm, which avoids the renormalization step and reduces minimization to a series of factorization steps mixed with the resolution of eigenvalue problems. It is then easy to show that the error decreases at each step of the iterative process, and it is in fact possible to show that the method actually converges to a local minimum [75, 89] although the proof is much more difficult and involves the global convergence theorem (GCT) from [70].

Here we propose to minimize E_2 under the constraint $\sum_{ij} |\mathcal{M}_i \boldsymbol{P}_j|^2 = 1$ [75]. Writing that the derivative of E_2 with respect to z_{ij} is zero at one of its minima can be used to eliminate this variable and show that, at a minimum, $E_2 = \sum_{ij} |\boldsymbol{p}_{ij} \times (\mathcal{M}_i \boldsymbol{P}_j)|^2$, where the vectors $\boldsymbol{p}_{ij}$ have been normalized as a preprocessing step. In particular, E_2 is proportional to the mean squared norm of vectors that depend on $\mathcal{M}_i$ and $\boldsymbol{P}_j$ in a bilinear fashion. Thus it can be minimized by alternating steps where $\mathcal{M}$ is fixed and $\mathcal{P}$ is estimated using homogeneous linear least squares with steps where $\mathcal{P}$ is fixed and $\mathcal{M}$ is estimated using homogeneous least squares (see [13, 40, 83] for related work). It is easy to show that the

error decreases at each step of the iterative process, and it is in fact possible to show that the method actually converges to a local minimum [75] using once again the GCT. Figure 8(a) uses real data to compare the proposed method with the Sturm-Triggs iterative factorization algorithm [108, 119], its provably-convergent variant proposed by Mahamud and Hebert [74], and the Morris implementation of non-linear bundle adjustment [84]. The figure plots the average and maximum reprojection errors (in pixels) obtained on a real image sequence that consists of 20 images of 30 points. In this case, 10 of the images have been used for training, and 10 have been used for testing.

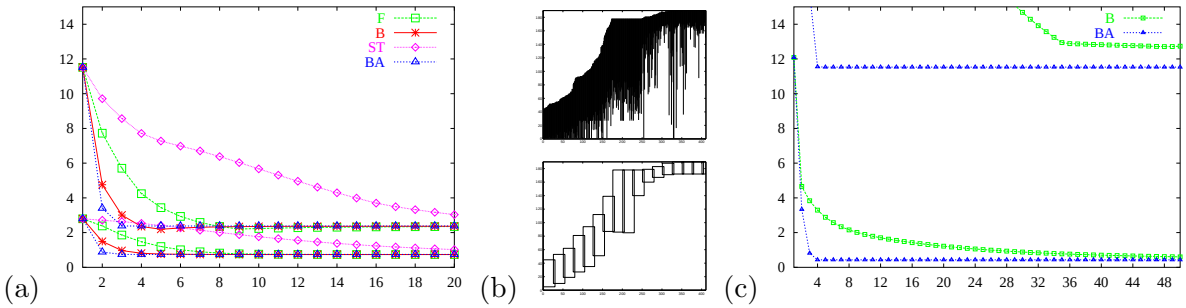

Figure 8: Empirical comparison of the proposed **B**ilinear iterative method (B), the **S**turm-**T**riggs algorithm (ST), the provably-convergent iterative **F**actorization method of Mahamud and Hebert (F), and the Morris implementation of **B**undle **A**djustment (BA) [75]. See text for details.

Figure 8(b)-(c) shows an other experiment with a sequence of 180 images that contain a total of 411 points. Not all the points are visible in all frames (Figure 8[b,top]). We have split the data in consecutive blocks of 30 frames with a 5-frame overlap, and used the Tomasi-Kanade method in the full rectangle of each block (Figure 8[b,bottom]) to compute an affine reconstruction of the scene. The successive reconstructions have been registered and used as input to the bilinear and bundle-adjustment methods. As shown by Figure 8(c), the initial errors are much larger in this case, and it takes the bilinear algorithm about 20 iterations to reach sub-pixel mean error, as opposed to 4 iterations for bundle adjustment. Although comparing the speed of the two implementations is a bit like comparing apples and oranges, it is worth noting that the bilinear algorithm takes 4 minutes to converge on this data, while bundle adjustment takes three hours. Thus the low cost of bilinear iterations greatly outweighs the fast convergence of bundle adjustment.

Projective scene reconstructions are not directly suitable for image synthesis: They first need to be "upgraded" to metric reconstructions using self-calibration techniques [30, 78, 93, 121], or equivalently, by computing an appropriate projective transformation. We have recently introduced a quasi-linear approach [95] to metric reconstruction from uncalibrated images under minimal assumptions that are true for most real cameras (i.e.,

rectangular pixels and/or known aspect ratio [48, 92]). This method is based on a novel characterization of metric upgrades from zero skew matrices. It is well known [29] that a 3×4 matrix $\hat{\mathcal{M}}$ represents a zero-skew camera when $(\hat{\boldsymbol{m}}_1 \times \hat{\boldsymbol{m}}_3) \cdot (\hat{\boldsymbol{m}}_2 \times \hat{\boldsymbol{m}}_3) = 0$, where $(\hat{\boldsymbol{m}}_i^T, \hat{m}_{i4})$ denotes the i^{th} row of $\hat{\mathcal{M}}$ $(i = 1, 2, 3)$. If $\boldsymbol{m}_i^T$ $(i = 1, 2, 3)$ and $\boldsymbol{q}_j$ $(j = 1, 2, 3, 4)$ are 4-vectors denoting respectively the rows of the projection matrix $\mathcal{M}$ and the columns of the metric upgrade matrix $\mathcal{Q}$, we have used line geometry to prove in [95] the following result.

Proposition 3 *Given a projection matrix $\mathcal{M}$ and a projective transformation $\mathcal{Q}$, a necessary and sufficient condition for the matrix $\hat{\mathcal{M}} = \mathcal{M}\mathcal{Q}$ to satisfy the zero-skew constraint is that $\boldsymbol{\lambda}^T \mathcal{R}^T \mathcal{R} \boldsymbol{\mu} = 0$, where $\mathcal{R}^T$ is the 6×3 matrix $(\boldsymbol{q}_2 \wedge \boldsymbol{q}_3, \boldsymbol{q}_3 \wedge \boldsymbol{q}_1, \boldsymbol{q}_1 \wedge \boldsymbol{q}_2)$, $\boldsymbol{\lambda} = \boldsymbol{m}_1 \wedge \boldsymbol{m}_3$ $\boldsymbol{\mu} = \boldsymbol{m}_2 \wedge \boldsymbol{m}_3$, and "$\wedge$" denotes the exterior product that associates with two points the vector of Plücker coordinates of the line passing through them.*

This result can be used to decompose the estimation of $\mathcal{Q}$ into (a) the computation of the matrix $\mathcal{R}^T \mathcal{R}$ using homogeneous linear least squares, (b) the estimation of $\mathcal{R}$ using a new algorithm for computing the best estimate of the square root of a non-necessarily positive symmetric matrix, and (c) the computation of $\mathcal{Q}$ using once again homogeneous linear least squares [63, 65]. Figure 9 shows preliminary results obtained with a 39-frame sequence of teddy bear images. A total of 4389 points visible in all frames are tracked automatically in this sequence using S. Birchfield's KLT implementation of the Kanade-Lucas-Tomasi tracker [69, 116].

5 Discussion

We plan to integrate the three approaches to 3D photography presented in this paper into a system capable of acquiring realistic visual models of complex objects from photographs and video sequences. Projective visual hulls will be used primarily in controlled situations where blue-screen technology can be used to delineate objects' silhouettes. For scenes with complex backgrounds, we will rely on affine-invariant patch matching, feature tracking, and motion segmentation to separate objects of interest from their background. Let us conclude by sketching a few research directions that we intend to follow as we pursue our integration efforts.

Projective visual hulls. As demonstrated by Figure 4, texture-mapped pictures of visual hulls constructed from a few photographs are reasonably realistic, but their visual quality degrades as the virtual camera moves away from the real ones (this is particularly

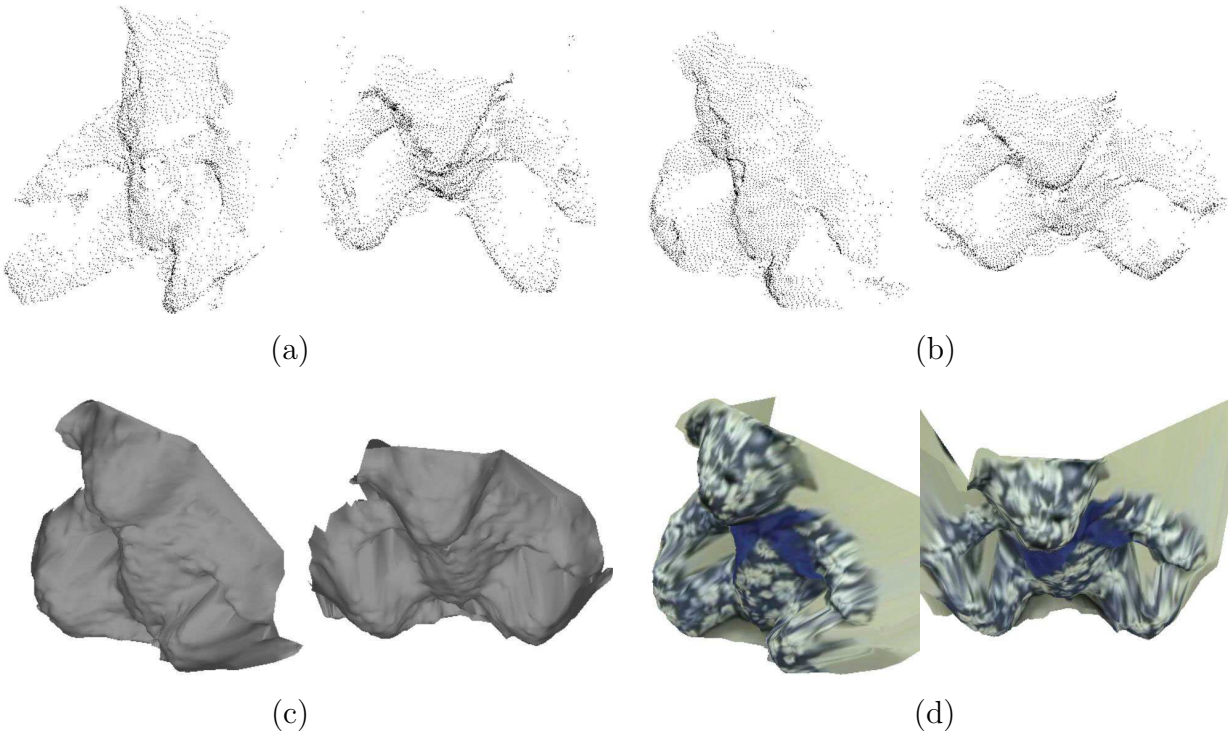

Figure 9: Object modeling from motion cues: (a) two views of the projective reconstruction of the teddy bear; (b) the corresponding Euclidean reconstruction; (c) its triangulation induced by the Delaunay triangulation of one of the input images' features; (d) texture-mapped views of this triangulation.

noticeable for the back of the shirt in the last view). Various methods can be used to remedy this problem. The simplest one is to interpolate the colors associated with all cameras observing each vertex of the mesh [24, 76, 77, 97]. A more accurate alternative is to add geometric detail to the raw visual hull. Classical stereo algorithms [25, 82, 86] cannot be used in this context because the input cameras are too far apart for normalized cross-correlation to return meaningful information. However, it is possible to take advantage of the rough surface model provided by the visual hull to reproject all input pictures in one reference image: This is an instance of the *stereo-from-shape* approach proposed by Debevec *et al.* [24], where the ray associated with a reference image pixel is first intersected with the model surface before being reprojected into an offset image. In practice, this zeroes out the wide-baseline disparities, allowing once again the use of correlation techniques to reveal fine surface detail. This idea is illustrated by Figure 10, where three of the pictures used to model a squash have been reprojected into the first image. Note the similarity between these pictures in their region of overlap, except (as could have been expected) in the neighborhood of specularities. In this setting, rim points play a particular role: They are the only surface points that belong to the boundary of the

visual hull. Accordingly, the disparity associated with each image reprojection must be zero along the occluding contours, and these curves can be found by a global optimization process maximizing correlation under the constraints of zero disparity and containment within the associated strips. We plan to implement a two-step process where the rims are found first, and used to anchor the subsequent stereo-from-shape process.

Figure 10: Three images of a squash, and their model-based reprojection into the first image; note how the specularities appear in different places in the three images.

Affine-invariant patches. When cameras are strongly calibrated, the projection matrices are known, and a single match between two affine-invariant patches provides 12 constraints (corresponding to the entries of the two matrices S_1 and S_2) in 11 unknowns (the 9 entries of $\mathcal{B}$ and the depths of the patch relative to the two cameras). Combining this constraint with the known epipolar geometry allows the independent verification of *individual* matches in wide-baseline situations (Figure 11). This complements the stereo-from-shape approach to adding geometric detail to visual hulls. Conversely, visual hulls can be used to discard incorrect matches between patches whose projections fall outside the input silhouettes.

Figure 11: Using affine-invariant patches for wide-baseline calibrated stereo. The matches found are indicated by lines joining corresponding features, and they are all correct in this example. We have only retained the affine-invariant patches with highest Harris response in this example, resulting in a sparse set of matches. A much larger set of patches would of course be used in actual applications.

When the cameras are *not* calibrated, the techniques described in Section 3 can be used to match successive images, (weakly) calibrate the cameras, and use the matches

found to add once again geometric detail to the visual hull. Perhaps more interestingly, affine-invariant patches should prove useful in modeling articulated objects from image sequences [6, 11, 26]: Indeed, the geometric consistency constraints derived in Section 3 will also hold for the *rigid parts* of an articulated object. Several recent algorithms for motion segmentation also rely on affine SFM constraints to find groups of rigidly-moving points in image sequences [9, 20, 33]. However, these techniques explicitly search for a permutation of the scene points producing geometrically consistent groups, and their high combinatorial cost has limited their practical applicability. They also assume that feature correspondences have already been established through tracking. In contrast, the matching strategy proposed in Section 3 is computationally efficient and it exploits geometric consistency constraints during the matching process itself. Figure 12 shows a preliminary experiment where the patches found in two pictures of an articulated object have been matched.

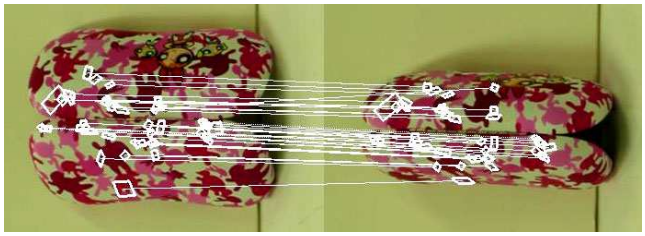

Figure 12: Two pictures of a glasses' case, completely open and just half open, with matches indicated by lines between corresponding features.

Structure from motion. Unlike affine or projective factorization methods [74, 108, 117], the bilinear algorithm proposed in Section 4 does not require all features to be present in all images (*sparse data*). In addition, its cost per iteration is only $O(mn)$ as opposed to $O(mn \min(3m, n))$ for factorization techniques and $O((m+n)^3)$ for bundle adjustment. Our bilinear algorithm is a constrained optimization technique that alternates steps where motion parameters are held constant with steps where structure parameters are held constant to solve a *homogeneous* least-squares problem. In the *non-homogeneous* case, it has been shown that a similar *alternation* [120] scheme called *NIPALS* [128] has the quadratic convergence rate of Gauss-Newton and other non-linear least-squares techniques used in bundle adjustment for dense data, but only linear convergence in the case of sparse data [100]. This is exactly the behavior observed in Figure 8. In the non-homogeneous case, algorithms that achieve a tradeoff between the NIPALS and Gauss-Newton schemes—with relatively low cost per iteration, yet near-quadratic convergence on sparse data—have been proposed [101]. We plan to adapt the same idea to our homogeneous setting by developing a variant of sequential quadratic programming [47] that decouples the motion

and structure parameters to achieve both computational efficiency and fast convergence. Finally, handling sparse data requires the discovery of a "good" set of overlapping blocks of frames and points in the corresponding image sequence. This can be formalized as the computation of a covering of an *interval graph* by (maximum) cliques. Clique problems are NP-complete in general [19], but simple and efficient algorithms are available for interval graphs [39]. We plan to apply these algorithms to the initial affine registration phase of the projective SFM technique proposed in Section 4.

Acknowledgments. This research was supported in part by the National Science Foundation under grants IRI-990709 and IIS-9907142, by the UIUC Campus Research Board, by the UIUC/CNRS collaboration agreement, and by the Beckman Institute. We wish to thank Edmond Boyer, Yasuhiro Omori, and Jeff Erickson for their participation to various stages of this research, and Andrew Fitzgibbon, Andrew Zisserman, Steve Sullivan, Oxford University, and Industrial Light & Magic for providing the data used in some of our experiments.

References

[1] S. Avidan and A. Shashua. Novel view synthesis in tensor space. In *Proc. IEEE Conf. Comp. Vision Patt. Recog.*, pages 1034–1040, San Juan, Puerto Rico, 1997.

[2] N. Ayache and B. Faverjon. Efficient registration of stereo images by matching graph descriptions of edge segments. *Int. J. of Comp. Vision*, pages 101–137, 1987.

[3] R. Bajcsy and F. Solina. Three-dimensional object representation revisited. In *Proc. Int. Conf. Comp. Vision*, pages 231–240, London, U.K., June 1987.

[4] E.B. Barrett, M.H. Brill, N.N. Haag, and P.M. Payton. Invariant linear models in photogrammetry and model-matching. In J. Mundy and A. Zisserman, editors, *Geometric Invariance in Computer Vision*, pages 277–292. MIT Press, Cambridge, Mass., 1992.

[5] B.G. Baumgart. Geometric modeling for computer vision. Technical Report AIM-249, Stanford University, 1974. Ph.D. Thesis. Department of Computer Science.

[6] S. Belongie, J. Malik, and J. Puzicha. Matching shapes. In *Proc. Int. Conf. Comp. Vision*, pages 454–461, 2001.

[7] M. Berger. *Geometry.* Springer-Verlag, 1987.

[8] P.J. Besl and N.D. McKay. A method for registration of 3D shapes. *IEEE Trans. Patt. Anal. Mach. Intell.*, 14(2):239–256, 1992.

[9] T.E. Boult and L.G. Brown. Factorization-based segmentation of motions. In *IEEE Workshop on Visual Motion*, pages 179–186, 1991.

[10] E. Boyer. Object Models from Contour Sequences. In *Proceedings of Fourth European Conference on Computer Vision, Cambridge, (England)*, pages 109–118, April 1996. Lecture Notes in Computer Science, volume 1065.

[11] C. Bregler and J. Malik. Tracking people with twists and exponential maps. In *Proc. IEEE Conf. Comp. Vision Patt. Recog.*, pages 8–15, 1998.

[12] O. Carmichael and M. Hebert. Unconstrained registration of large 3D point sets for complex model building. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 360–367, Victoria, Canada, October 1998.

[13] Q. Chen and G. Medioni. Efficient iterative solution to m-view projective reconstruction problem. In *Proc. IEEE Conf. Comp. Vision Patt. Recog.*, volume II, pages 55–61, Fort Collins, Colorado, June 1999.

[14] S.E. Chen. Quicktime VR: An image-based approach to virtual environment navigation. In *SIGGRAPH*, pages 29–38, Los Angeles, CA, August 1995.

[15] R. Cipolla, K.E. Astrom, and P.J. Giblin. Motion from the frontier of curved surfaces. In *Proc. Int. Conf. Comp. Vision*, pages 269–275, Boston, MA, 1995.

[16] R. Cipolla and A. Blake. Surface shape from the deformation of the apparent contour. *Int. J. of Comp. Vision*, 9(2):83–112, November 1992.

[17] R.T. Collins, A.R. Hanson, and E.M. Riseman. Site model acquisition under the UMass RADIUS project. In *Proc. DARPA Image Understanding Workshop*, pages 351–358, 1994.

[18] C.I. Connolly and J.R. Stenstrom. 3D scene reconstruction from multiple intensity images. In *Proc. IEEE Workshop on Interpretation of 3D Scenes*, pages 124–130, Austin, TX, November 1989.

[19] T.H. Cormen, C.E. Leiserson, and R.L.Rivest. *Introduction of algorithms.* MIT Press, 1990.

[20] J. Costeira and T. Kanade. A multi-body factorization method for motion analysis. In *Proc. Int. Conf. Comp. Vision*, pages 1071–1076, Boston, MA, 1995.

[21] G. Cross, A.W. Fitzgibbon, and A. Zisserman. Parallax geometry of smooth surfaces in multiple views. In *Proc. Int. Conf. Comp. Vision*, pages 323–329, Corfu, Greece, 1999.

[22] B. Curless and M. Levoy. A volumetric method for building complex models from range images. In *SIGGRAPH*, New Orleans, LA, August 1996.

[23] P. Debevec, T. Hawkins, C. Tchou, H.-P. Duiker, W. Sarokin, and M. Sagar. Acquiring the reflectance field of a human face. In *SIGGRAPH*, pages 145–156, 2000.

[24] P. Debevec, C.J. Taylor, and J. Malik. Modeling and rendering architecture from photographs: a hybrid geometry- and image-based approach. In *SIGGRAPH*, pages 11–20, New Orleans, LA, August 1996.

[25] F. Devernay and O.D. Faugeras. Computing differential properties of 3D shapes from stereopsis without 3D models. In *Proc. IEEE Conf. Comp. Vision Patt. Recog.*, pages 208–213, Seattle, WA, June 1994.

[26] D.E. DiFranco, T.-J. Cham, and J.M. Rehg. Recovery of 3D articulated motion from 2D correspondences. Technical Report CRL 99/7, Compaq Cambridge Research Laboratory, 1999.

[27] O. Faugeras, Q.-T. Luong, and T. Papadopoulo. *The Geometry of Multiple Images*. MIT Press, 2001.

[28] O.D. Faugeras. What can be seen in three dimensions with an uncalibrated stereo rig? In G. Sandini, editor, *Proc. European Conf. Comp. Vision*, volume 588 of *Lecture Notes in Computer Science*, pages 563–578, Santa Margherita, Italy, 1992. Springer-Verlag.

[29] O.D. Faugeras. *Three-Dimensional Computer Vision*. MIT Press, 1993.

[30] O.D. Faugeras. Stratification of 3D vision: projective, affine and metric representations. *J. Opt. Soc. Am. A*, 12(3):465–484, March 1995.

[31] O.D. Faugeras and E. Pauchon. Measuring the shape of 3D objetcs. In *Proc. IEEE Conf. Comp. Vision Patt. Recog.*, pages 2–7, Washington, D.C., 1983.

[32] A. Fitzgibbon and A. Zisserman. Automatic 3D model acquisition and generation of new images from video sequences. In *European Signal Processing Conference*, pages 311–326, Rhodes, Greece, 1998.

[33] C.W. Gear. Multibody grouping in moving objects. *Int. J. of Comp. Vision*, 29(2):133–150, August/September 1998.

[34] Y. Genc and J. Ponce. Image-based rendering using parameterized image varieties. *Int. J. of Comp. Vision*, 41(3):143–170, 2001.

[35] D.B. Gennery. *Modelling the environment of an exploring vehicle by means of stereo vision.* PhD thesis, Stanford University, Stanford, CA, 1980.

[36] S.J. Gortler, R. Grzeszczuk, R. Szeliski, and M. Cohen. The lumigraph. In *SIGGRAPH*, pages 43–54, New Orleans, LA, August 1996.

[37] W.E.L. Grimson. A computer implementation of a theory of human stereo vision. *Philosophical Transactions of the Royal Society of London*, pages 217–253, 1981.

[38] A.D. Gross and T.E. Boult. Error of fit measures for recovering parametric solids. In *Proc. Int. Conf. Comp. Vision*, pages 690–694, Tampa, FL, December 1988.

[39] U.I. Gupta, D.T. Lee, and Y.Y.-T. Leung. Efficient algorithms for interval graphs and circular-arc graphs. *Networks*, 12:459–467, 1982.

[40] M. Han and T. Kanade. Creating 3d models with uncalibrated cameras. In *Proceedings WACV*, 2000.

[41] C. Harris and M. Stephens. A combined edge and corner detector. In 4^{th} *Alvey Vision Conference*, pages 189–192, Manchester, UK, 1988.

[42] R. Hartley. Lines and points in three views and the trifocal tensor. *Int. J. of Comp. Vision*, 22(2):125–140, March 1997.

[43] R. Hartley. Cheirality. *Int. J. of Comp. Vision*, 26(1):41–61, 1998.

[44] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision.* Cambridge University Press, 2000.

[45] R.I. Hartley. In defence of the 8-point algorithm. In *Proc. Int. Conf. Comp. Vision*, pages 1064–1070, Boston, MA, 1995.

[46] R.I. Hartley, R. Gupta, and T. Chang. Stereo from uncalibrated cameras. In *Proc. IEEE Conf. Comp. Vision Patt. Recog.*, pages 761–764, Champaign, IL, 1992.

[47] M.T. Heath. *Scientific Computing: An Introductory Survey.* McGraw-Hill, 2002. Second edition.

[48] A. Heyden and K. Åström. Minimal conditions on intrinsic parameters for Euclidean reconstruction. In *Asian Conference on Computer Vision*, Hong Kong, 1998.

[49] M. Irani and P. Anandan. A unified approach to moving object detection in 2D and 3D scenes. *IEEE Trans. Patt. Anal. Mach. Intell.*, 20(6):577–589, 1998.

[50] A. Johnson and M. Hebert. Object recognition by matching oriented points. In *Proc. IEEE Conf. Comp. Vision Patt. Recog.*, pages 684–689, San Juan, Puerto Rico, June 1997.

[51] A.E. Johnson and M. Hebert. Surface matching for object recognition in complex three-dimensional scenes. *Image and Vision Computing*, 16:635–651, 1998.

[52] T. Joshi, N. Ahuja, and J. Ponce. Structure and motion estimation from dynamic silhouettes under perspective projection. *Int. J. of Comp. Vision*, 31(1):31–50, February 1999.

[53] T. Kanade, P.W. Rander, and J.P Narayanan. Virtualized reality: Constructing virtual worlds from real scenes. *IEEE Multimedia*, 4(1):34–47, 1997.

[54] R.E. Kelly, P.R.H. McConnell, and S.J. Mildenberger. The Gestalt photomapping system. *Photogrammetric Engineering and Remote Sensing*, 43(11):1407–1417, 1977.

[55] D. Keren, D. Cooper, and J. Subrahmonia. Describing complicated objects by implicit polynomials. *IEEE Trans. Patt. Anal. Mach. Intell.*, 16(1):38–53, 1994.

[56] J.J. Koenderink. What does the occluding contour tell us about solid shape? *Perception*, 13:321–330, 1984.

[57] J.J. Koenderink and A.J. Van Doorn. Affine structure from motion. *J. Opt. Soc. Am. A*, 8:377–385, 1990.

[58] K.M. Kutulakos and S.M. Seitz. A theory of shape by space carving. In *Proc. Int. Conf. Comp. Vision*, pages 307–314, Corfu, Greece, 1999.

[59] K.N. Kutulakos and J. Vallino. Calibration-free augmented reality. *IEEE Transactions on Visualization and Computer Graphics*, 4(1):1–20, Jan–Mar 1998.

[60] E. Lane. *Projective Differential Geometry of Curves and Surfaces*. The University of Chicago Press, 1932.

[61] A. Laurentini. How far 3D shapes can be understood from 2D silhouettes. *IEEE Trans. Patt. Anal. Mach. Intell.*, 17(2):188–194, February 1995.

[62] S. Laveau and O.D. Faugeras. 3D scene representation as a collection of images and fundamental matrices. Technical Report 2205, INRIA Sophia-Antipolis, 1994.

[63] S. Lazebnik. Projective visual hulls. Technical Report MS Thesis, University of Illinois at Urbana-Champaign, 2002.

[64] S. Lazebnik, E. Boyer, and J. Ponce. On computing exact visual hulls of solids bounded by smooth surfaces. In *Proc. IEEE Conf. Comp. Vision Patt. Recog.*, pages 156–161, 2001.

[65] S. Lazebnik and J. Ponce. The local projective shape of smooth surfaces and their outlines. In *Proc. Int. Conf. Comp. Vision*, 2003. Submitted.

[66] M. Levoy and P. Hanrahan. Light field rendering. In *SIGGRAPH*, pages 31–42, New Orleans, LA, August 1996.

[67] T. Lindeberg. Feature detection with automatic scale selection. *Int. J. of Comp. Vision*, 30(2):79–116, 1998.

[68] T. Lindeberg and J. Gårding. Shape-adapted smoothing in estimation of 3-D shape cues from affine deformations of local 2-D brightness structure. *Image and Vision Computing*, 15(6):415–434, 1997.

[69] B.D. Lucas and T. Kanade. An iterative image resistration technique with an application to stereo vision. In *Proc. International Joint Conference on Artificial Intelligence*, pages 674–679, 1981.

[70] D.G. Luenberger. *Linear and nonlinear programming*. Addison-Wesley, 1984. Second edition.

[71] Q.-T. Luong and O.D. Faugeras. The fundamental matrix: theory, algorithms, and stability analysis. *Int. J. of Comp. Vision*, 17(1):43–76, January 1996.

[72] Y. Ma, S. Soatto, J. Kosecka, and S. Sastry. Euclidean reconstruction and reprojection up to subgroups. In *Proc. Int. Conf. Comp. Vision*, pages 773–780, Corfu, Greece, 1999.

[73] S. Magda, T. Zickler, D. Kriegman, and P. Belhumeur. Beyond Lambert: Reconstructing surfaces with arbitrary BRDFs. In *Proc. Int. Conf. Comp. Vision*, Vancouver, CA, 2001.

[74] S. Mahamud and M. Hebert. Iterative projective reconstruction from multiple views. In *Proc. IEEE Conf. Comp. Vision Patt. Recog.*, pages II–430–437, Hilton Head, SC, June 2000.

[75] S. Mahamud, M. Hebert, Y. Omori, and J. Ponce. Provably-convergent iterative methods for projective structure from motion. In *Proc. IEEE Conf. Comp. Vision Patt. Recog.*, pages 1018–1025, 2001.

[76] W. Matusik, C. Buehler, R. Raskar, S. Gortler, and L. McMillan. Image-based visual hulls. In *SIGGRAPH*, 2001.

[77] W. Matusik, H. Pfister, A. Ngan, P. Beardsley, R. Ziegler, and L. McMillan. Image-based 3D photography using opacity hulls. In *SIGGRAPH*, 2002.

[78] S.J. Maybank and O.D. Faugeras. A theory of self-calibration of a moving camera. *Int. J. of Comp. Vision*, 8(2):123–151, 1992.

[79] L. McMillan and G. Bishop. Plenoptic modeling: an image-based rendering approach. In *SIGGRAPH*, pages 39–46, Los Angeles, CA, August 1995.

[80] K. Mikolajczyk and C. Schmid. An affine invariant interest point detector. In *Proc. European Conf. Comp. Vision*, volume I, pages 128–142, Copenhagen, Denmark, 2002.

[81] G. Miller, S. Rubin, and D. Ponceleon. Lazy decompression of surface light fields for precomputed global illumination. In *Eurographics Rendering Workshop*, pages 281–292, 1998.

[82] H.P. Moravec. Toward automatic visual obstacle avoidance. In *Proc. International Joint Conference on Artificial Intelligence*, page 584, Cambridge, MA, 1977.

[83] D.D. Morris and T. Kanade. A unified factorization algorithm for points, line segments and planes with uncertainty models. In *Proc. Int. Conf. Comp. Vision*, pages 696–702, Bombay, India, 1998.

[84] D.D. Morris, K. Kanatani, and T. Kanade. Uncertainty modeling for optimal structure from motion. In B. Triggs, A. Zisserman, and R. Szeliski, editors, *Vision Algorithms: Theory and Practice*, pages 200–215. Springer-Verlag, 2000. Lecture Notes in Computer Science 1883.

[85] W. Niem and R. Buschmann. Automatic modelling of 3D natural objects from multiple views. In *European Workshop on Combined Real and Synthetic Image Processing for Broadcast and Video Production*, Hamburg, Germany, 1994.

[86] H.K. Nishihara. PRISM, a practical real-time imaging stereo matcher. AI Memo 780, MIT, 1984.

[87] K. Nishino, Y. Sato, and K. Ikeuchi. Appearance compression and synthesis based on 3D model for mixed reality. In *Proc. Int. Conf. Comp. Vision*, pages 38–45, 1999.

[88] Y. Ohta and T. Kanade. Stereo by intra- and inter-scanline search. *IEEE Trans. Patt. Anal. Mach. Intell.*, 7(2):139–154, 1985.

[89] J. Oliensis. Fast and accurate self-calibration. In *Proc. Int. Conf. Comp. Vision*, pages 745–752, Kerkyra, Greece, September 1999.

[90] A.P. Pentland. Perceptual organization and the representation of natural form. *Artificial Intelligence Journal*, 28:293–331, 1986.

[91] 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.

[92] M. Pollefeys. *Self-calibration and metric 3D reconstruction from uncalibrated image sequences*. PhD thesis, Katholieke Universiteit Leuven, 1999.

[93] M. Pollefeys, R. Koch, and L. Van Gool. Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters. *Int. J. of Comp. Vision*, 32(1):7–26, August 1999.

[94] M. Pollefeys, R. Koch, M. Verguwen, and L. Van Gool. Flexible 3D acquisition with a monocular camera. In *IEEE Int. Conf. on Robotics and Automation*, pages 2771–2776, Leuven, Belgium, June 1998.

[95] J. Ponce. Metric upgrade of a projective reconstruction under the rectangular pixel assumption. In *Second Workshop on Structure from Multiple Images of Large Scale Environments*, pages 18–27, Dublin, Ireland, 2000.

[96] P. Pritchett and A. Zisserman. Wide baseline stereo matching. In *Proc. Int. Conf. Comp. Vision*, pages 754–760, Bombay, India, 1998.

[97] K. Pulli, M. Cohen, T. Duchamp, , H. Hoppe, L. Shapiro, and W. Stuetzle. View-based rendering: Visualizing real objects from scanned range and color data. In *Eurographics Rendering Workshop*, pages 23–34, 1997.

[98] J.H. Rieger. Three-dimensional motion from fixed points of a deforming profile curve. *Optics Letters*, 11:123–125, 1986.

[99] F. Rothganger, S. Lazebnik, C. Schmid, and J. Ponce. 3D object modeling and recognition using affine-invariant patches and multi-view spatial constraints. In *Proc. IEEE Conf. Comp. Vision Patt. Recog.*, 2003. Accepted for publication.

[100] A. Ruhe. Numerical computation of principal components when several observations are missing. Technical Report 1974-08-13, University of Umeå, 1974.

[101] A. Ruhe and P.ÅDewin. Algorithms for separable nonlinear least squares problems. *SIAM Review*, 22(3):318–337, 1980.

[102] S.M. Seitz and C.R. Dyer. Physically-valid view synthesis by image interpolation. In *Workshop on Representations of Visual Scenes*, Boston, MA, 1995.

[103] A. Shashua. Algebraic functions for recognition. *IEEE Trans. Patt. Anal. Mach. Intell.*, 17(8):779–789, August 1995.

[104] H.-Y. Shum, M. Han, and R. Szelisky. Interactive construction of 3D models from panoramic mosaics. In *Proc. IEEE Conf. Comp. Vision Patt. Recog.*, pages 427–433, Santa Barbara, CA, June 1998.

[105] M.E. Spetsakis and Y. Aloimonos. Structure from motion using line correspondences. *Int. J. of Comp. Vision*, 4(3):171–183, 1990.

[106] S. Srivastava and N. Ahuja. Octree generation from object silhouettes in perspective views. *Computer Vision, Graphics and Image Processing*, 49(1):68–84, 1990.

[107] J. Stolfi. *Oriented Projective Geometry: A Framework for Geometric Computations.* Academic Press, 1991.

[108] P. Sturm and B. Triggs. A factorization-based algorithm for multi-image projective structure and motion. In *Proc. European Conf. Comp. Vision*, pages 709–720, 1996.

[109] S. Sullivan and J. Ponce. Automatic model construction, pose estimation, and object recognition from photographs using triangular splines. *IEEE Trans. Patt. Anal. Mach. Intell.*, 20(10):1091–1096, Oct. 1998.

[110] S. Sullivan, L. Sandford, and J. Ponce. Using geometric distance fits for 3D object modelling and recognition. *IEEE Trans. Patt. Anal. Mach. Intell.*, 16(12):1183–1196, December 1994.

[111] G. Taubin. Estimation of planar curves, surfaces and nonplanar space curves defined by implicit equations, with applications to edge and range image segmentation. *IEEE Trans. Patt. Anal. Mach. Intell.*, 13(11):1115–1138, 1990.

[112] G. Taubin, F. Cukierman, S. Sullivan, J. Ponce, and D.J. Kriegman. Parameterized families of polynomials for bounded algebraic and surface curve fitting. *IEEE Trans. Patt. Anal. Mach. Intell.*, 16(3):287–303, March 1994.

[113] D. Tell and S. Carlsson. Wide baseline point matching using affine invariants computed from intensity profiles. In *Proc. European Conf. Comp. Vision*, pages 814–828, Dublin, Ireland, 2000.

[114] D. Terzopoulos and D. Metaxas. Dynamic 3D models with local and global deformation: deformable superquadrics. *IEEE Trans. Patt. Anal. Mach. Intell.*, 13(7):703–714, 1991.

[115] D. Terzopoulos, A. Witkin, and M. Kass. Constraints on deformable models: Recovering 3D shape and nonrigid motion. *Artificial Intelligence*, pages 91–123, 1988.

[116] C. Tomasi and T. Kanade. Detection and tracking of point features. Technical Report CMU-CS-91132, Carnegie Mellon University, 1991.

[117] 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.

[118] P.H.S. Torr, A.W. Fitzgibbon, and A. Zisserman. The problem of degeneracy in structure and motion estimation from uncalibrated motion sequences. *Int. J. of Comp. Vision*, 32(1):27–44, August 1999.

[119] B. Triggs. Factorization methods for projective structure from motion. In *Proc. IEEE Conf. Comp. Vision Patt. Recog.*, pages 845–851, 1996.

[120] B. Triggs, P.F. McLauchlan, R.I. Hartley, and A.W. Fitzgibbon. Bundle adjustment - a modern synthesis. In B. Triggs, A. Zisserman, and R. Szeliski, editors, *Vision Algorithms: Theory and Practice*, pages 298–372. Springer-Verlag, 2000. Lecture Notes in Computer Science 1883.

[121] W. Triggs. Auto-calibration and the absolute quadric. In *Proc. IEEE Conf. Comp. Vision Patt. Recog.*, pages 609–614, San Juan, Puerto Rico, June 1997.

[122] G. Turk and M. Levoy. Zippered polygon meshes from range images. In *SIGGRAPH*, pages 311–318, Orlando, Fla, July 1994.

[123] T. Tuytelaars and L. Van Gool. Matching widely separated views based on affinely invariant neighbourhoods. *Int. J. of Comp. Vision*, 2002. Submitted.

[124] S. Ullman. *The Interpretation of Visual Motion*. The MIT Press, Cambridge, MA, 1979.

[125] R. Vaillant and O.D. Faugeras. Using extremal boundaries for 3D object modeling. *IEEE Trans. Patt. Anal. Mach. Intell.*, 14(2):157–173, February 1992.

[126] D. Weinshall and C. Tomasi. Linear and incremental acquisition of invariant shape models from image sequences. *IEEE Trans. Patt. Anal. Mach. Intell.*, 17(5), May 1995.

[127] J. Weng, N. Ahuja, and T. Huang. Matching two perspective views. *IEEE Trans. Patt. Anal. Mach. Intell.*, 14(8):806–825, August 1992.

[128] H. Wold and E. Lyttkens. Nonlinear iterative partial least squares (nipals) estimation procedures. *Bull. ISI*, 43:29–51, 1969.

[129] D. Wood, D. Azuma, K. Aldinger, B. Curless, T. Duchamp, D. Salesin, and W. Stuerzle. Surface light fields for 3D photography. In *SIGGRAPH*, pages 287–296, 2000.

[130] Z. Zhang, R. Deriche, O.D. Faugeras, and Q.-T. Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence Journal*, 78:87–119, October 1995.