Geometrically Constrained Structure from Motion: Points on Planes

Richard Szeliski and P. H. S. Torr

Microsoft Research, One Microsoft Way, Redmond, WA 98052, USA,
szeliski-philtorr@microsoft.com,
http://www.research.microsoft.com/research/vision/

Abstract. Structure from motion algorithms typically do not use external geometric constraints, e.g., the coplanarity of certain points or known orientations associated with such planes, until a final post-processing stage. In this paper, we show how such geometric constraints can be incorporated early on in the reconstruction process, thereby improving the quality of the estimates. The approaches we study include hallucinating extra point matches in planar regions, computing fundamental matrices directly from homographies, and applying coplanarity and other geometric constraints as part of the final bundle adjustment stage. Our experimental results indicate that the quality of the reconstruction can be significantly improved by the judicious use of geometric constraints.

1 Introduction

Structure from (image) motion algorithms attempt to simultaneously recover the 3D structure of a scene or object and the positions and orientations of the cameras used to photograph the scene. Algorithms for recovering structure and motion have many applications, such as the construction of 3D environments and pose localization for robot navigation and grasping, the automatic construction of 3D CAD models from photographs, and the creation of large photorealistic virtual environments.

Structure from motion is closely related to photogrammetry, where the 3D location of certain key *control points* is usually known, thereby allowing the recovery of camera pose prior to the estimation of shape through triangulation techniques. In structure from motion, however, very few constraints are usually placed (or assumed) on the geometric structure of the scene being analyzed. This has encouraged the development of mathematically elegant and general formulations and algorithms that can be applied in the absence of any prior knowledge.

In practice, however, structure from motion is often applied to scenes which contain strong geometric regularities. The man made world is full of planar structures such as floors, walls, and tabletops, many of which have known orientations e.g. horizontal, vertical or known relationships e.g. parallelism and perpendicularity. Even the natural world tends to have certain regularities, such as the generally vertical direction of tree growth, or the existence of relatively

Reinhard Koch, Luc Van Gool (Eds.): SMILE '98, LNCS 1506, pp. 171–186, 1998.

flat ground planes. A quick survey of many recent structure from motion papers indicates that the test data sets include some very strong regularities (mostly horizontal and vertical planes and lines) which are never exploited [22,2], except perhaps for a final global shape correction.

In this paper, we argue using external geometric knowledge can never decrease the quality of a reconstruction so long as this knowledge is applied in a statistically valid way. Rather than developing a single algorithm or methodology, we examine a number of different plausible ways to bring geometric constraints to bear, and then evaluate these empirically. In this way, we hope to elucidate where geometric constraints can be used effectively. Our experiments demonstrate that hallucinating additional correspondences in areas of known planar motion, and applying higher order constraints such as perpendicularities between planes, can lead to significantly better reconstruction.

After a brief review of related literature in Section 2, we present the basic imaging equations, develop the relationships between point positions in two views, and show how this reduces to a homography for the case of coplanar points (Section 3). In Section 4 we preview the three main approaches we will use to solve the structure from motion problem when subsets of points are known to lie on planes: augmenting planes with additional sample points before computing the fundamental matrix (Section 5); using homographies to directly compute the fundamental matrix (Section 6); using plane plus parallax techniques (Section 7); and performing global optimization (bundle adjustment) (Section 8). In Section 9 we discuss how additional knowledge about the planes (e.g., perpendicularity constraints) can be used to improve the solution. Section 10 presents our experimental setup . We close with a discussion of the results, and a list of potential extensions to our framework, including the important case of line data.

2 Previous Work

There has been a large amount of work on recovery of structure and motion from image sequences (A good introductory text book on the subject is [4]). However, relatively little work has been done on incorporating prior geometric knowledge (e.g., the coplanarity of points, or known feature orientations) directly into the reconstruction process.

There has been some work in exploiting the motion of one or more planes for recovering structure and motion. Luong and Faugeras [14] show how to directly compute a fundamental matrix from two or more of the homographies induced by the motions of planes within the image. This technique however is very noise sensitive. Plane plus parallax technique directly exploit a known dominant planar motion to compute the epipole(s) and perform a projective reconstruction [11,17,10]. However, none of these approaches incorporate the geometric constraints of coplanarity in a statistically optimal fashion.

3 General Problem Formulation

Structure from motion can be formulated as the recovery of a set of 3-D structure parameters $\{\mathbf{x}_i = (X_i, Y_i, Z_i)\}$ and time-varying motion parameters $\{(\mathbf{R}_k, \mathbf{t}_k)\}$ from a set of observed image features $\{\mathbf{u}_{ik} = (u_{ik}, v_{ik}, 1)\}$. In this section, we present the forward equations, i.e., the rigid body and perspective transformations which map 3-D points into 2-D image points. We also derive the *homography* (planar perspective transform) which relates two views of a planar point set.

To project the ith 3-D point $\mathbf{x}_i$ into the kth frame at location $\mathbf{u}_{ik}$, we write

$$\mathbf{u}_{ik} \sim \mathbf{V}_k \mathbf{R}_k (\mathbf{x}_i - \mathbf{t}_k), \tag{1}$$

where $\sim$ indicates equality up to a scale, $\mathbf{R}_k$ is the rotation matrix for camera k, $\mathbf{t}_k$ is the location of its optical center, and $\mathbf{V}_k$ is its projection matrix (usually assumed to be upper triangular or some simpler form, e.g., diagonal). In most cases, we will assume that $\mathbf{R}_0 = \mathbf{I}$ and $\mathbf{t}_0 = 0$, i.e., the first camera is at the world origin. The location of a 3D point corresponding to an observed image feature is

$$\mathbf{x}_i = w_{ik} \mathbf{R}_k^{-1} \mathbf{V}_k^{-1} \mathbf{u}_{ik} + \mathbf{t}_k, \tag{2}$$

where w_{ik} is an unknown scale factor.

It is useful to distinguish three cases, depending on the form of $\mathbf{V}_k$. If $\mathbf{V}_k$ is known, we have the *calibrated* image case. If $\mathbf{V}_k$ is unknown and general (upper triangular), we have the *uncalibrated* image case, from which we can only recover a *projective* reconstruction of world [4]. If some information about $\mathbf{V}_k$ is known (e.g., that it is temporally invariant, or that it has a reduced form), we can apply *self-calibration* techniques [7,12].

The motion of a point between two images k and l can thus be written as

$$\mathbf{u}_{ik} \sim \mathbf{V}_k \mathbf{R}_k (w_{il} \mathbf{R}_l^{-1} \mathbf{V}_l^{-1} \mathbf{u}_{il} + \mathbf{t}_l - \mathbf{t}_k) \sim \mathbf{H}_{kl}^{\infty} \mathbf{u}_{il} + w_{il}^{-1} \mathbf{e}_{kl}, \tag{3}$$

with $\mathbf{R}_{kl} = \mathbf{R}_k \mathbf{R}_l^{-1}$. The matrix $\mathbf{H}_{kl}^{\infty} = \mathbf{V}_k \mathbf{R}_{kl} \mathbf{V}_l^{-1}$ is the *homography* (planar perspective transform) which maps points at infinity ($w_{il}^{-1} = 0$) from one image to the next, while $\mathbf{e}_{kl} = \mathbf{V}_k \mathbf{R}_k (\mathbf{t}_l - \mathbf{t}_k)$ is the *epipole* which is the vanishing point of the *residual parallax vectors* once this planar perspective motion has been subtracted (the epipole is also the image of camera k's center in camera l's image, as can be seen by setting $w_{il} \to 0$).

When the cameras are uncalibrated, i.e., the $\mathbf{V}_k$ can be arbitrary, the homography $\mathbf{H}_{kl}^{\infty}$ cannot be uniquely determined, i.e., we can add an arbitrary matrix of the form $\mathbf{e}_{kl} \mathbf{v}^T$ to $\mathbf{H}_{kl}^{\infty}$ and subtract a plane equation $\mathbf{v}^T \mathbf{u}_k$ from w_{il}^{-1} and still obtain the same result. More globally, the reconstructed 3D shape can only be determined up to an overall 3D global perspective transformation (*collineation*) [4,19].

The inter-image transfer equations have a simpler form when $\mathbf{x}$ is known to lie on a plane $\hat{\mathbf{n}}^T \mathbf{x} - d = 0$. In this case, we can compute w_{il} using

$$\hat{\mathbf{n}}^T \mathbf{x}_i - d = w_{il} \hat{\mathbf{n}}^T \mathbf{R}_l^{-1} \mathbf{V}_l^{-1} \mathbf{u}_{il} + \hat{\mathbf{n}}^T \mathbf{t}_l - d = 0,$$

or

$$w_{il}^{-1} = \hat{\mathbf{n}}^T \mathbf{R}_l^{-1} \mathbf{V}_l^{-1} \mathbf{u}_{il} / (d - \hat{\mathbf{n}}^T \mathbf{t}_l) = d_l^{-1} \hat{\mathbf{n}}^T \mathbf{R}_l^{-1} \mathbf{V}_l^{-1} \mathbf{u}_{il},$$

where $d_l = d - \hat{\mathbf{n}}^T \mathbf{t}_l$ is the distance of camera center l ($\mathbf{t}_l$) to the plane $(\hat{\mathbf{n}}, d)$. Substituting w_{il}^{-1} into (3) and multiplying through by d_l, we obtain [24]

$$\mathbf{u}_{ik} \sim (\mathbf{H}_{kl}^{\infty} + d_l^{-1} \mathbf{e}_{kl} \hat{\mathbf{n}}^T \mathbf{R}_l^{-1} \mathbf{V}_l^{-1}) \mathbf{u}_{il}. \tag{4}$$

Letting $\tilde{\mathbf{n}}_l = \mathbf{V}_l^{-T} \mathbf{R}_l \hat{\mathbf{n}}$ be the plane normal in the lth camera's (scaled) coordinate system, we see that the homography induced by the plane can be written as

$$\mathbf{H}_{kl} \sim \mathbf{H}_{kl}^{\infty} + d_l^{-1} \mathbf{e}_{kl} \tilde{\mathbf{n}}_l^T \tag{5}$$

i.e., it is very similar in form to the projective ambiguity which arises when using uncalibrated cameras (this also forms the basis of the plane plus parallax techniques discussed below).

4 Structure from Motion with Planes

In the remainder of this paper, we develop a number of techniques for recovering the structure and motion of a collection of points seen with 2 or more cameras. In addition to being given the estimated position of each point in two or more images, we also assume that some of the points are coplanar. We may also be given one or more image regions where the inter-frame homographies are known, but no explicit correspondences have been given.

Given this information, there are several ways we could proceed.

1. We can, of course, solve the problem ignoring our knowledge of coplanarity. This will serve as our reference algorithm against which we will compare all others.
2. We can hallucinate (additional) point matches based on the homographies which are either given directly or which can be computed between collections of coplanar points.
3. We can re-compute the 2D point locations so that the estimated or computed homographies are exactly satisfied.
4. We can use the homographies induced by the planes in the image to estimate the fundamental matrix, and thence structure.
5. We can use plane + parallax techniques to recover the camera geometry, and after that the projective 3D structure.
6. We can perform a global optimization (bundle adjustment), using the knowledge about coplanarity as additional constraints to be added to the solution.

To illustrate these algorithms, we initially use two simple data sets (Figure 1):

1. a collection of n points lying in a fronto-parallel plane with m points lying on a closer fronto-parallel plane;

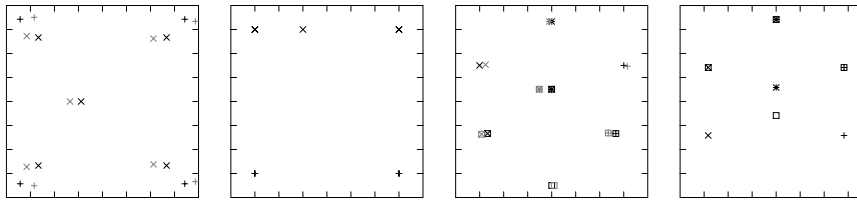

Fig. 1. Experimental datasets (front and top views): *Front view* shows the location of the points projected into image 1 (black symbol) and image 2 (grey symbol). *Top view* shows the relative 3d disposition of the points in orthographic projection from above. (a) $n = 5$ points lying on a plane with $m = 4$ points lying in front (b) $n = 4$ points on each face of a trihedral vertex. For our experiments, we use $\mathbf{t}_0 = 6$ and rotate the data around the vertical axis through 10°.

2. a trihedral vertex with n points on each of the three faces, with two of the points on each face being located along the common edge.

Although we could, we will not use data sets where homographies are directly given. Instead, we compute whatever homographies we need from the (noisy) 2D point measurements, and use these as inputs. A more detailed explanation of our data and methodology is given in Section 10.

5 Fundamental Matrices from Point Correspondences

Referring back to the basic two-frame transfer equation (3), we can pre-multiply both sides by $[\mathbf{e}_{kl}]_\times$, where $[\mathbf{v}]_\times$ is the matrix form of the cross-product operator with vector $\mathbf{v}$, to obtain

$$[\mathbf{e}_{kl}]_\times \mathbf{u}_{ik} \sim [\mathbf{e}_{kl}]_\times \mathbf{V}_k \mathbf{R}_{kl} \mathbf{V}_l^{-1} \mathbf{u}_{il}$$

(since $[\mathbf{e}_{kl}]_\times$ annihilates the $\mathbf{e}_{kl}$ vector on the right hand side). Pre-multiplying this by $\mathbf{u}_{ik}^T$, we observe that the left-hand side is 0 since the cross product matrix is skew symmetric, and hence

$$\mathbf{u}_{ik}^T \mathbf{F}_{kl} \mathbf{u}_{il} = 0, \tag{6}$$

where

$$\mathbf{F}_{kl} \sim [\mathbf{e}_{kl}]_\times \mathbf{V}_k \mathbf{R}_{kl} \mathbf{V}_l^{-1} = [\mathbf{e}_{kl}]_\times \mathbf{H}_{kl}^\infty \tag{7}$$

is called the *fundamental matrix* [4]. The fundamental matrix is of rank 2, since that is the rank of $[\mathbf{e}_{kl}]_\times$, and has seven degrees of freedom (the scale of $\mathbf{F}$ is arbitrary).

When the camera calibration is known, we can premultiply screen coordinates by $\mathbf{V}^{-1}$ (i.e., convert screen coordinates into Euclidean directions), and obtain

the simpler *essential matrix*, $\mathbf{E} \sim [\mathbf{t}_{kl}]_\times \mathbf{R}_{kl}$, which has two identical non-zero singular values, and hence 5 degrees of freedom (the fundamental matrix has 7).

The *fundamental matrix* or *essential matrix* approach to two-frame structure from motion is one of the most widely used techniques for structure from motion, and some recent modifications have made this technique quite reliable in practice [23,25]. The essential matrix method was first developed for the calibrated image case [13]. This method was then generalized to the fundamental matrix approach [3,9], which can be used with uncalibrated cameras.

Once the fundamental (or essential) matrix has been computed, we can estimate $\mathbf{e}_{kl}$ [23,25], and then compute the desired homography $\mathbf{H}_{kl}^{\infty} = [\mathbf{e}_{kl}]_\times \mathbf{F}_{kl}$. The 3D location of each point can then be obtained by triangulation, and in our experiments, this can be compared to the known ground truth.

As mentioned earlier, when we know that certain points are coplanar, we can use this information in one of two ways: (1) hallucinate (additional) point matches based on the homographies; or (2) re-compute the 2D point locations so that the estimated or computed homographies are exactly satisfied.

5.1 Hallucinating Additional Correspondences

The first approach proves to be useful in data-poor situations, e.g., when we only have four points on a plane, and two points off the plane. By hallucinating additional correspondences, we can generate enough data (say, two additional points on the plane) to use a regular 8-point algorithm. If it helps for data poor situations, why not for other situations as well (say, eight points grouped onto two planes)? Eventually, of course, the new data must be redundant, but at what point? Methods which exploit homographies directly [14] (Section 6) indicate that there are six independent constraints available from a single homography. Is this so when the data is noisy?

Let's get a feel for how much additional points help by running some experiments. Table 1 shows the results of adding p hallucinated points per plane to both of our test data sets (bi-plane and trihedral),[1] and then running an 8-point algorithm to reconstruct the data [8,25]. For the initially underconstrained data sets ($n = 4, m = 2$ bi-plane and $n = 4$ trihedral), and even for the minimally constrained data sets ($n = 4, m = 4$ bi-plane), adding enough hallucinated points to get more than the minimum required 8 provides a dramatic improvement in the quality of the results. On the other hand, adding hallucinated points to data set which already have more than 8 points only gives a minor improvement. This suggests that having more than the minimal number of sample points is more important than fully exploiting all of the constraints available from our homographies.

[1] The $n = 6, m = 2$ and $n = 4, m = 2$ data sets actually only have a single plane for which a homography can be computed.

data set	n	m	p	N	method	Euclidean	affine	co-planarity
plane + 2pts	4	2	2	8	"8 pt" $\mathbf{F}$	0.0651	0.0130	0.0023
"	4	2	0	6	plane + $\|$ax	0.0651	0.0130	0.0024
plane + 2pts †	6	2	0	8	"8 pt" $\mathbf{F}$	0.1879	0.0430	0.0149
"	6	2	1	9	"	0.1482	0.0285	0.0158
"	6	2	0	8	plane + $\|$ax	0.1185	0.0184	0.0105
2 $\|$ planes †	4	4	0	8	"8 pt" $\mathbf{F}$	0.1382	0.0335	0.0141
"	4	4	1	10	"	0.0858	0.0235	0.0128
"	4	4	2	12	"	0.0702	0.0200	0.0100
"	4	4	0	8	plane + $\|$ax	0.1709	0.0395	0.0077
2 $\|$ planes	5	5	0	10	"8 pt" $\mathbf{F}$	0.0538	0.0226	0.0144
"	5	5	0	10	reproject	0.0484	0.0189	0.0114
"	5	5	0	10	$\mathbf{H} \rightarrow \mathbf{F}$	0.5516	0.3698	0.0163
"	5	5	0	10	plane + $\|$ax	0.0673	0.0189	0.0079
"	5	5	0	10	bundle adj.	0.0467	0.0170	0.0092
"	5	5	0	10	plane enf.	0.0392	0.0117	0.0000
"	5	5	0	10	plane constr.	0.0384	0.0081	0.0000
6 $\|$ planes	4	4	0	24	"8 pt" $\mathbf{F}$	0.0761	0.0234	0.0074
"	4	4	0	24	$\mathbf{H} \rightarrow \mathbf{F}$	0.9459	0.7652	0.0088
"	4	4	0	24	plane + $\|$ax	0.8145	0.5312	0.0078
tilted cube	4	4	1	10	"8 pt" $\mathbf{F}$	0.1549	0.0307	0.0091
"	4	4	2	13	"	0.1301	0.0265	0.0076
"	4	4	0	7	$\mathbf{H} \rightarrow \mathbf{F}$	0.1383	0.0237	0.0079
"	4	4	0	7	plane + $\|$ax	0.2070	0.0411	0.0087
tilted cube	5	5	0	10	"8 pt" $\mathbf{F}$	0.1460	0.0295	0.0110
"	5	5	1	13	"	0.1263	0.0256	0.0111
"	5	5	0	10	$\mathbf{H} \rightarrow \mathbf{F}$	0.1014	0.0213	0.0093
"	5	5	0	10	plane + $\|$ax	0.1657	0.0348	0.0107

† Randomized data point placement

Table 1. Reconstruction error for various methods of structure estimation. n and m are defined in Figure 1, p is the number of extra hallucinated points, and N is the total number of points. The Euclidean and affine reconstruction errors are for calibrated cameras. The coplanarity error measures the Euclidean distance of points to their best-fit plane (calibrated reconstruction).

5.2 Reprojecting Points Based on Homographies

A second approach to exploiting known coplanarity in the data set is to perturb the input 2D measurements such that they lie exactly on a homography. This seems like a plausible thing to do, e.g., projecting 3D points onto estimated planes is one way to "clean up" a 3D reconstruction. However, it is possible that this early application of domain knowledge may not be statistically optimal or even admissible. Let's explore this idea empirically.

The simplest way to perform this reprojection is to first compute homographies between a plane in the kth frame and the 0th frame, and to then project the points from the first frame into the kth frame using this homography. This is equivalent to assuming that the points in the first frame are noise-free. Another approach is to find $\mathbf{u}_{ik}^*$ such that they exactly satisfy the homographies and minimize the projected errors. Since the latter involves a complicated minimization, we have chosen to study the former, simpler idea. Methods to incorporate coplanarity as a hard constraint on the solution will be presented in Section 9.

Table 1 shows some results of reprojecting points in the second frame based on the computed homographies (row *reproject*). A slight decrease in error is visible, but this technique does not yield as dramatic improvements as hallucinating additional correspondences.

6 Fundamental Matrices from Homographies

Assuming that we are given (or can estimate) the inter-frame homographies associated with two or more planes in the scene, there is a more direct method for computing the fundamental matrix [14]. Recall from (5) that the homography associated with a plane $\hat{\mathbf{n}}^T\mathbf{x} - d = 0$ is $\mathbf{H}_{kl} \sim \mathbf{H}_{kl}^{\infty} + d_l^{-1}\mathbf{e}_{kl}\tilde{\mathbf{n}}^T$ and that the fundamental matrix (7) associated with the same configuration has the form $\mathbf{F}_{kl} \sim [\mathbf{e}_{kl}]_{\times}\mathbf{H}_{kl}^{\infty}$. The product

$$\mathbf{H}_{kl}^T\mathbf{F}_{kl} \sim \mathbf{H}_{kl}^{\infty T}[\mathbf{e}_{kl}]_{\times}\mathbf{H}_{kl}^{\infty} + d_l^{-1}\tilde{\mathbf{n}}\mathbf{e}_{kl}^T[\mathbf{e}_{kl}]_{\times}\mathbf{H}_{kl}^{\infty} = \mathbf{H}_{kl}^{\infty T}[\mathbf{e}_{kl}]_{\times}\mathbf{H}_{kl}^{\infty}$$

is skew symmetric, and hence

$$\mathbf{S} = \mathbf{H}_{kl}^T\mathbf{F}_{kl} + \mathbf{F}_{kl}^T\mathbf{H}_{kl} = 0. \tag{8}$$

Writing out these equations in terms of the entries h_{ij} and f_{ij} of $\mathbf{H}$ and $\mathbf{F}$ (we'll drop the kl frame subscripts) gives us

$$s_{ij} = \sum_k h_{ki}f_{kj} + f_{ki}h_{kj} = 0, \quad \forall(i,j). \tag{9}$$

Each known plane homography $\mathbf{H}$ contributes six independent constraints on $\mathbf{F}$, since the matrix $\mathbf{H}^T\mathbf{F} + \mathbf{F}^T\mathbf{H}$ is symmetric, and hence only has six degrees of freedom. Using two or more plane homographies, we can form enough equations to obtain a linear least squares problem in the entries in $\mathbf{F}$.

While this idea is quite simple, Luong and Faugeras [14] report that the technique is not very stable (it only yields improvements over a point-based technique for two planes). Can we deduce why this method performs poorly?

Instead of using (8) to solve for $\mathbf{F}$, what if we "hallucinate" point correspondences based on the known homographies. Say we pick an image point $\mathbf{u} = (u_0, u_1, u_2)$ and project it to $\mathbf{u}' = (v_0, v_1, v_2) = \mathbf{H}\mathbf{u}$, i.e., $v_k = \sum_i h_{ki}u_i$. The resulting constraint on $\mathbf{F}$ (6) has the form

$$\sum_{lj} v_k f_{kj} u_j = \sum_{ijk} f_{kj} h_{ki} u_i u_j = 0. \tag{10}$$

By choosing appropriate values for (u_0, u_1, u_2), we can obtain elements (or combinations of the elements) of the symmetric $\mathbf{S}$ matrix in (8). For example, when $\mathbf{u} = \delta_i$, we get $\sum_k f_{ki} h_{ki} = \frac{1}{2} s_{ii}$. Thus, three of the constraints used by [14] correspond to sampling the homography at points $(1, 0, 0)$,(0,1,0), and $(0, 0, 1)$, two of which lie at infinity! Similarly, for $\mathbf{u} = \delta_i + \delta_j, i \neq j$, we get $\sum_k f_{ki} h_{ki} + f_{ki} h_{kj} + f_{kj} h_{ki} + f_{kj} h_{kj} = \frac{1}{2} s_{ii} + \frac{1}{2} s_{jj} + s_{ij}$. Thus, the remaining three constraints used by [14] are linear combinations of constraints corresponding to three sample points, e.g., $(0, 1, 1)$, $(0, 1, 0)$, and $(0, 0, 1)$. Again, each constraint uses at least one sample point at infinity! This explains why the technique does not work so well. First, the homographies are sampled at locations where their predictive power is very weak (homographies are most accurate at predicting the correspondence *within* the area from which they were extracted). Second, the resulting sample and projected points are far from having the kind of nice unit distribution required for total least squares to work reasonably well.

To demonstrate the overall weakness of this approach, we show the reconstruction error using the method of [14] in Table 1. From these results, we can see that the approach is often significantly inferior to simply sampling the same homography with sample points in the interior of the region from which it was extracted. The six-plane data set (Figure 2) is representative of the kind of data used in [14], where they partitioned the image into regions and then scattered coplanar points within each region. For the trihedral data set, however, the homography-based method works quite well. To obtain comparable results using the point hallucination method, quite a few additional sample points need to be used. At the moment, we do not yet understand the discrepancy between the fronto-parallel and trihedral data set results. A plausible conjecture is that fronto-parallel data, whose "vanishing points" lie at far away from the optical center, are more poorly represented by an $\mathbf{H}$ matrix.

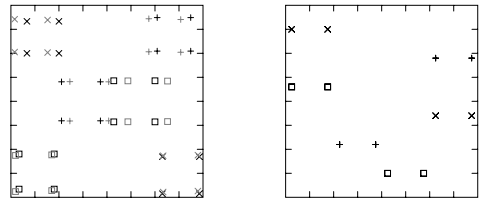

Fig. 2. Points clustered onto 6 fronto-parallel planes

7 Plane plus Parallax

Another traditional approach to exploiting one or more homographies between different views is to choose one homography as the *dominant motion*, and to

compute the *residual parallax*, which should point at the epipole. Such *plane plus parallax* techniques [11,17] are usually used to recover a *projective* description of the world, although some work has related the projective depth (magnitude of the parallax) to Euclidean depth.

To compute the fundamental matrix, we choose one of the homographies, say the first one, and use it to warp all points from one from to the other. We then compute the epipole by minimizing the sum of squared triple products, $(\mathbf{x}_i, \mathbf{x}'_i, \mathbf{e})^2$, where $\mathbf{x}_i$ and $\mathbf{x}_i$ are corresponding (after transfer by homography).[2] Once $\mathbf{e}$ has been determined, we can compute $\mathbf{F}_{kl}$.

Table 1 shows the results of using the plane plus parallax to recover the 3D structure for some of our data sets. The method works well when the points are mostly on the plane used for the homography ($n = 4$ or 6, $m = 2$), but not as well when the points are evenly distributed over several planes. This is not surprising. Plane plus parallax privileges one plane over all others, forcing the fundamental matrix to exactly match that homography. When the data is more evenly distributed, a point-based algorithm (with hallucination, if necessary) gives better results.

8 Global Optimization (Bundle Adjustment)

The final technique we examine in this paper is the one traditionally used by photogrammetrists, i.e., the simultaneous optimization of 3D point and camera placements by minimizing the squared error between estimated and measured image feature locations.

There are two general approaches to performing this optimization. The first interleaves structure and motion estimation stages [8]. This has the advantage that each point (or frame) reconstruction problem is decoupled from the other problems, thereby solving much smaller systems. The second approach simultaneously optimizes for structure and motion [19]. This usually requires fewer iterations, because the couplings between the two sets of data are made explicit, but requires the solution of larger systems. In this paper, we adopt the former approach. To reconstruct a 3D point location, we minimize

$$\sum_k \left(u_{ik} - \frac{\mathbf{p}_{k0}^T \tilde{\mathbf{x}}_i}{\mathbf{p}_{k2}^T \tilde{\mathbf{x}}_i} \right)^2 + \left(v_{ik} - \frac{\mathbf{p}_{k1}^T \tilde{\mathbf{x}}_i}{\mathbf{p}_{k2}^T \tilde{\mathbf{x}}_i} \right)^2 \tag{11}$$

where $\mathbf{p}_{kr}$ are the three rows of the *camera or projection matrix*

$$\mathbf{P}_k = \mathbf{V}_k[\mathbf{R}_k| - \mathbf{t}_k]$$

and $\tilde{\mathbf{x}}_i = [\mathbf{x}_i|1]$, i.e., the homogeneous representation of $\mathbf{x}_i$. As pointed out by [25], this is equivalent to solving the following overconstrained set of linear equations,

[2] The triple product measures the distance of $\mathbf{e}$ from the line passing through $\mathbf{x}_i$ and $\mathbf{x}'_i$, weighted by the length of this line segment.

$$D_{ik}^{-1}(\mathbf{p}_{k0} - u_{ik}\mathbf{p}_{k2})^T\tilde{\mathbf{x}}_i = 0 \qquad (12)$$
$$D_{ik}^{-1}(\mathbf{p}_{k1} - v_{ik}\mathbf{p}_{k2})^T\tilde{\mathbf{x}}_i = 0,$$

where the weights are given by $D_{ik} = \mathbf{p}_{k2}^T\tilde{\mathbf{x}}_i$ (these are set to $D_{ik} = 1$ in the first iteration).[3] Notice that since these equations are homogeneous in $\tilde{\mathbf{x}}_i$, we solve this system by looking for the rightmost singular vector of the system of equations [6].

The same equations can be used to update our estimate of $\mathbf{P}_k$, by simply grouping equations with common k's into separate systems. When reconstructing a Euclidean ($\mathbf{V}_k[\mathbf{R}_k|-\mathbf{t}_k]$) description of motion, the estimation equations become more complicated. Here, applying a linearized least squares like the Levenberg-Marquardt algorithm is more fruitful. Let us assume the following updates

$$\mathbf{R}_k \leftarrow \mathbf{R}_k(\mathbf{I} + [\omega_k]_\times), \qquad \mathbf{t}_k \leftarrow \mathbf{t}_k + \delta\mathbf{t}_k. \qquad (13)$$

We can compute the terms in

$$\mathbf{P}_k + \delta\mathbf{P}_k = \mathbf{V}_k[\mathbf{R}_k(\mathbf{I} + [\omega_k]_\times)| - (\mathbf{t}_k + \delta\mathbf{t}_k)] \qquad (14)$$

as functions of $\mathbf{V}_k, \mathbf{R}_k, \mathbf{t}_k$, i.e., the Jacobian of the twelve entries in $\delta\mathbf{P}_k$ with respect to ω_k and $\delta\mathbf{t}_k$. We can then solve the system of equations

$$D_{ik}^{-1}\tilde{\mathbf{x}}_i^T(\delta\mathbf{p}_{k0} - \hat{u}_{ik}\delta\mathbf{p}_{k2}) = u_{ik} - \hat{u}_{ik} \qquad (15)$$
$$D_{ik}^{-1}\tilde{\mathbf{x}}_i^T(\delta\mathbf{p}_{k1} - \hat{v}_{ik}\delta\mathbf{p}_{k2}) = v_{ik} - \hat{v}_{ik},$$

substituting the $\delta\mathbf{p}_k$ with their expansions in the unknowns $(\omega_k, \delta\mathbf{t}_k)$. The rotation and translation estimates can then be updated using (13), using Rodriguez's formula for the rotation matrix [1],

$$\mathbf{R} \leftarrow \mathbf{R}\left(\mathbf{I} + \sin\theta[\hat{\mathbf{n}}]_\times + (1-\cos\theta)[\hat{\mathbf{n}}]_\times^2\right)$$

with $\theta = \|\omega\|$, $\hat{\mathbf{n}} = \omega/\theta$. A similar approach can be used to update the focal length, or other intrinsic calibration parameters, if desired.

The above discussion has assumed that each point can be solved for independently. What about points that are known to be coplanar? Here, we need to incorporate constraints of the form $\hat{\mathbf{n}}_p^T\mathbf{x}_i - d_p = 0, \{i \in \mathit{\Pi}_p\}$. Two approaches come to mind. The first is to alternate a plane estimation stage with the point reconstruction stage. The second is to simultaneously optimize the point positions and plane equations. We describe the former, since it is simpler to implement.

Fitting planes to a collection of 3D points is a classic total least squares problem [6]. After subtracting the centroid of the points, $\overline{\mathbf{x}}_p$, we compute the

[3] The Levenberg-Marquardt algorithm [15] leads to a slightly different set of equations

$$D_{ik}^{-1}(\mathbf{p}_{k0} - \hat{u}_{ik}\mathbf{p}_{k2})^T\delta\tilde{\mathbf{x}}_i = u_{ik} - \hat{u}_{ik},$$

where $\hat{u}_{ik}$ is the current estimate of u_{ik} and $\delta\tilde{\mathbf{x}}_i$ is the desired update to $\tilde{\mathbf{x}}_i$. In practice, the two methods perform about as well [25].

singular value decomposition of the resulting deviations, and choose the rightmost singular vector as the plane equation. We then set $d_p = \hat{\mathbf{n}}_p^T \overline{\mathbf{x}}_p$.

To enforce this hard constraint on the point reconstruction stage, we add the equation $\hat{\mathbf{n}}_p^T \mathbf{x}_i - d_p = 0$. to the system (13) as a linear constraint [6]. Since points may end up lying on several planes, we use the *method of weighting* approach to constrained least squares [6, p. 586], i.e., we add the constraints $\hat{\mathbf{n}}_p^T \mathbf{x}_i - d_p = 0$ to the set of equations for $\mathbf{x}_i$ with a large weight (currently $2^{60} \approx 10^{20}$).

Table 1 shows the results of applying bundle adjustment to the initial structure and motion estimates computed using an 8-point method. For fronto-parallel planes, bundle adjustment significantly reduces the reconstruction error. For the trihedral data set, it has little effect. Notice, however, the large discrepancy between the Euclidean and affine reconstruction errors for the trihedral data. This suggests that the major source of error is probably a *bas-relief ambiguity* [20], which is not removable even with a statistically optimal technique such as bundle adjustment. Enforcing coplanarity ("plane enf." in Table 1) does not significantly reduce the reconstruction error, although it is successful at reducing coplanarity error to 0 (which may be desirable to make the data appear less "wobbly").

9 Constraints on Planes

In addition to grouping points onto planes, we can apply additional constraints on the geometry of the planes themselves. For example, if we know that two or more planes are parallel, then we can compute a single normal vector for all the "coplanar" points after their individual centroids have been subtracted.

The line corresponding to method "plane constr." in Table 1 shows the result of applying a parallelism constraint to our fronto-parallel data set. The results are not all that different from not using the constraint.

If we know that certain planes are perpendicular, this too can be enforced during the normal computation stage. If two or three planes are known to be mutually orthogonal, we can concatenate the normals into a matrix, compute its SVD, replace the singular values with 1, and reconstitute the matrix.

Applying this idea to the trihedral data set as part of the bundle adjustment loop yields dramatically lower reconstruction errors (Table 1). Adding the perpendicularity constraint removes most of the bas-relief ambiguity (uncertainty) in the reconstruction, with the resulting reconstruction error being more closely tied to the triangulation error.

Lastly, if planes have explicitly known orientations (e.g., full constraints in the case of ground planes, or partial constraints in the case of vertical walls), these too can be incorporated. However, a global rotation and translation of coordinates may first have to be applied to the current estimate before these constraints can be enforced.

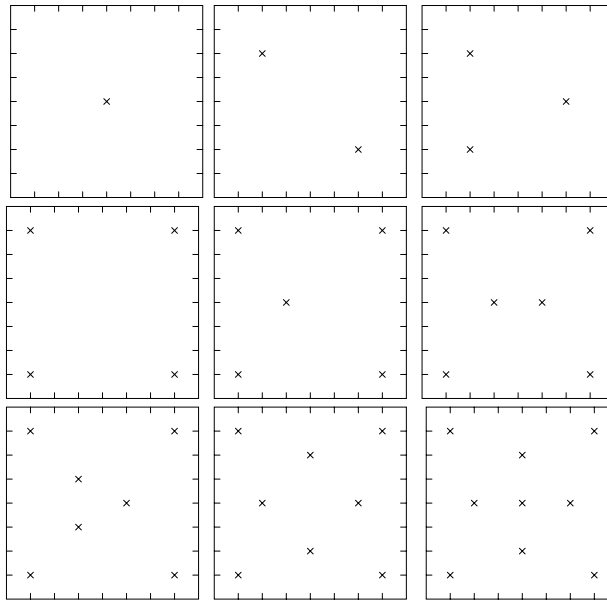

Fig. 3. Point layouts for $n = 1 \ldots 9$ points

10 Experiments

We have performed an extensive set of experiments to validate our algorithms and to test the relative merits of various approaches. Our experimental software first generates a 3D dataset in one of three possible configurations: a set of fronto-parallel planes filling the field of view (Figure 1a), a set of fronto-parallel planes in non-overlapping regions of the image (Figure 2), or a trihedral corner (Figure 1b). On each of the planes, we generate from 1 to 9 sample points, in the configurations shown in Figure 3.

The 3D configuration of points is projected onto the camera's image plane. For our current experiments, we rotate the data around the y axis in increments of 10°, and place the camera 6 units away from the data (the data itself fills a cube spanning $[-1, 1]^3$). We then generate 50 noise-corrupted versions of the projected points (for the experiments described in this paper, $\sigma = 0.2$ pixels, on a 200×200 image), and use these as inputs to the reconstruction algorithms. The *mean* RMS reconstruction error across all 50 trials is then reported.

The reconstruction errors are computed after first finding the best 3D mapping (Euclidean/similarity or affine) from the reconstructed data points onto the known ground truth points. The columns labeled "Euclidean" and "affine" in Table 1 measure the errors between data reconstructed using calibrated cameras and the ground truth after finding the best similarity and affine mappings. The

coplanarity error is computed by finding the best 3D plane fit to each coplanar set of reconstructed points (calibrated camera), and then measuring the distances to the plane.

Table 1 is a representative sample from our more extensive set of experiments.

11 Discussion and Conclusions

In this paper, we have presented a number of techniques for exploiting the geometric knowledge typically available in structure from motion problems. In particular, we have focused on how to take advantage of known coplanarities in the data. Our techniques also enable us to directly exploit homographies between different regions of the image, when these are known. Of the techniques tried, hallucinating additional correspondences is simple to implement, and often yields a significant improvement in the results, especially in situations which are initially data-poor. Reprojecting the data to exactly fit the homography does not appear to significantly improve the results. Using homographies to directly estimate the fundamental matrix sometimes works, but also often fails dramatically; using hallucinated correspondences seems like a more prudent approach.

Bundle adjustment improves the results obtained with the 8-point algorithm, but often not by that much. Adding coplanarity as a hard constraint does not seem to make a significant difference in the accuracy of the reconstruction, although it does make the reconstruction look smoother. Adding parallelism as a geometric constraint does not seem to improve the results that much. On the other hand, adding perpendicularity constraints for the trihedral data set leads to a dramatic decrease in reconstruction error (most likely due to a reduction in the bas-relief ambiguity). As mentioned above, plane plus parallax works well when the points are mostly on the plane used for estimating the homography, but not as well when the points are evenly distributed over several planes.

These results suggest that adding hallucinated correspondences to planar grouping of points (or hallucinating correspondences in regions with known homographies) is a useful and powerful idea which improves structure from motion results with very little additional complexity. Similarly, geometric constraints (coplanarity, parallelism, and perpendicularity) can be added to the bundle adjustment stage with relatively little effort, and can provide significantly improved results.

11.1 Future Work

This paper has concentrated on the geometric constraints available from knowing that certain points are coplanar. Similar constraints are available for points which are known to be collinear. The situation, however, is often a little different: line matching algorithms often do not localize the endpoints of lines in each image, so there may be no initial points in correspondence, nor is it possible to hallucinate such correspondences prior to an actual reconstruction. However,

exploiting known orientations for lines (e.g., vertical and horizontal), and geometric constraint between their orientations (parallelism and perpendicularity) is indeed possible, and can lead to algorithms which reconstruct a 3D scene from a single view.

In terms of points on planes, our current results could be extended in a number of directions. First, we have not yet explored the use of multi-frame algebraic approaches such as trilinear tensors [18]. Second, we have not explored multi-frame bundle adjustment techniques, nor have we explored the use of robust estimation techniques [23]. Hallucinating correspondences should be equally applicable to all three of these approaches. We would also like to better understand the differences in results obtained from fronto-parallel and oblique planes, and in general to anticipate the expected accuracy of results for various geometric configurations and camera motions.

References

1. N. Ayache. *Vision Stéréoscopique et Perception Multisensorielle*. InterEditions., 1989.
2. P. Beardsley, P. Torr, and A. Zisserman. 3D model acquisition from extended image sequences. In *ECCV'96*, volume 2, pages 683–695, 1996.
3. O. D. Faugeras. What can be seen in three dimensions with an uncalibrated stereo rig? In *ECCV'92*, pages 563–578, 1992.
4. O. Faugeras. *Three-dimensional computer vision: A geometric viewpoint*. MIT Press, 1993.
5. O. D. Faugeras and F. Lustman. Motion and structure from motion in a piecewise planar environment. *Int J Pattern Recognition and Artificial Intelligence*, 2(3):485–508, 1988.
6. G. Golub and C. F. Van Loan. *Matrix Computation, third edition*. The John Hopkins Univ Press, 1996.
7. R. I. Hartley. An algorithm for self calibration from several views. In *CVPR'94*, pages 908–912, 1994.
8. R. I. Hartley. In defense of the 8-point algorithm. *PAMI*, 19(6):580–593, 1997.
9. R. Hartley, R. Gupta, and T. Chang. Stereo from uncalibrated cameras. In *CVPR'93*, pages 489–494, 1993.
10. M. Irani, B. Rousso, and S. Peleg. Recovery of ego-motion using image stabilization. *PAMI*, 19(3):268–272, March 1997.
11. R. Kumar, P. Anandan, and K. Hanna. Direct recovery of shape from multiple views. In *ICPR'94*, volume A, pages 685–688, 1994.
12. Q.-T. Luong and O. D. Faugeras. Self-calibration of a moving camera from point correspondences and fundamental matrices. *IJCV*, 22(3):261–289, 1997.
13. H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, 1981.
14. Q.-T. Luong and O. Faugeras. Determining the fundamental matrix with planes. In *CVPR'93*, volume 1, pages 489–494, 1993.
15. W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in C*. Cambridge University Press, 1992.
16. M. E. Spetsakis and J. Y. Aloimonos. Optimal motion estimation. In *IEEE Workshop on Visual Motion*, pages 229–237, Irvine, California, March 1989.

17. H. S. Sawhney. 3D geometry from planar parallax. In *CVPR'94*, pages 929–934, 1994.
18. A. Shashua. Trilinearity in visual recognition by alignment. In *ECCV'94*, volume 1, pages 479–484, 1994. Springer-Verlag.
19. R. Szeliski and S. B. Kang. Recovering 3D shape and motion from image streams using nonlinear least squares. *Journal of Visual Communication and Image Representation*, 5(1):10–28, 1994.
20. R. Szeliski and S. B. Kang. Shape ambiguities in structure from motion. *PAMI*, 19(5):506–512, 1997.
21. R. Y. Tsai and T. S. Huang. Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surfaces. *PAMI*, PAMI-6(1):13–27, 1984.
22. C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization method. *IJCV*, 9(2):137–154, 1992.
23. P. H. S. Torr and D. W. Murray. A review of robust methods to estimate the fundamental matrix. *IJCV*, 24(3):271–300, 1997.
24. T. Viéville, C. Zeller, and L. Robert. Using collineations to compute motion and structure in an uncalibrated image sequence. *IJCV*, 20(3):213–242, 1996.
25. Z. Zhang. Determining the epipolar geometry and its uncertainty: A review. *International Journal of Computer Vision*, accepted 1997.