Direct Recovery of Planar-Parallax from Multiple Frames

Michal Irani, *Member*, *IEEE*,
P. Anandan, *Member*, *IEEE*, and Meir Cohen

Abstract—In this paper, we present an algorithm that estimates dense planar-parallax motion from *multiple uncalibrated* views of a 3D scene. This generalizes the "plane+parallax" recovery methods to more than two frames. The parallax motion of pixels across multiple frames (relative to a planar surface) is related to the 3D scene structure and the camera epipoles. The parallax field, the epipoles, and the 3D scene structure are estimated directly from image brightness variations across multiple frames, without precomputing correspondences.

Index Terms—Plane+parallax, direct (gradient-based) methods, multiframe analysis, correspondence estimation, structure from motion.

1 Introduction

THE recovery of the 3D structure of a scene and the camera epipolar-geometries (or camera motion) from multiple views has been a topic of considerable research. The large majority of the work on structure-from-motion (SFM) has assumed that correspondences between image features (typically, a sparse set of image points) are given and focused on the problem of recovering SFM based on this input. Another class of methods has focused on recovering dense 3D structure from a set of dense correspondences or an optical flow field. While these have the advantage of recovering *dense* 3D structure, they require that the correspondences are known. However, correspondence (or flow) estimation is a notoriously difficult problem.

A small set of techniques have attempted to combine the correspondence estimation step together with SFM recovery. These methods obtain dense correspondences while *simultaneously* estimating the 3D structure and the camera geometries (or motion) [4], [15], [19], [22], [21], [2]. By interweaving the two processes, the local correspondence estimation process is constrained by the current estimate of (global) epipolar geometry (or camera motion) and vice versa. These techniques minimize the violation of the brightness gradient constraint with respect to the unknown structure and motion parameters. Typically, this leads to a significant improvement in the estimated correspondences (and the attendant 3D structure) and some improvement in the recovered camera geometries (or motion). These methods are sometimes referred to as "direct methods" [4] since they directly use image brightness information to recover 3D structure and motion without explicitly computing correspondences as an intermediate step.

While [4], [22], [21], [2] recover 3D information relative to the camera, the "plane+parallax" approach [20], [15], [19], [10], [13], [12], recovers 3D information relative to a planar surface in the scene (the "reference plane"). The underlying concept is that, after the alignment of the reference plane, the residual image motion is due only to the *translational* motion of the camera and to the *deviations* of the scene structure from the planar surface. All effects of camera rotation or changes in camera calibration are eliminated by the plane stabilization. Hence, the residual image motion (the planar-parallax displacements) forms a *radial flow field* centered at the epipole. The "plane+parallax" representation has several benefits over the traditional camera-centered representation which make it an attractive framework for correspondence estimation and for 3D shape recovery:

1. *Reduced search space.* By parametrically aligning a visible image structure (which usually corresponds to a planar surface in the scene), all effects of unknown rotation and calibration parameters are folded into the homographies used for patch alignment. The only remaining unknown global camera parameters are the epipoles (i.e., three global unknowns per frame; gauge ambiguity is reduced to a single global scale factor for all epipoles across all frames). Since, after plane alignment, the residual parallax displacements are constrained to lie along radial lines emerging from the epipoles, correspondence estimation at each pixel reduces from a 2D search problem into a simpler 1D search problem. This has the additional benefit that it can uniquely resolve correspondences, even for pixels which lie on line structures (i.e., pixels which suffer from the aperture problem).
2. *Provides shape relative to a plane in the scene.* In many applications, fluctuations with respect to a plane in the scene are more useful than distances from the camera. For example, in robot navigation, heights of scene points from the ground plane can be immediately translated into evidence for obstacles or holes.
3. *A compact representation.* By removing the common global component (the plane homography), the residual parallax displacements are usually very small and, hence, require significantly fewer bits to encode the shape fluctuations than as required to encode distances from the camera.
4. *A stratified 2D-3D representation.* Work on motion analysis can be roughly classified into two classes of techniques: 2D algorithms, which handle cases with no 3D parallax (e.g., estimating homographies, 2D affine transformations, etc.), and 3D algorithms which handle cases with dense 3D parallax (e.g., estimating fundamental matrices, trifocal tensors, 3D shape, etc). Prior *model selection* [23] is usually required to decide which set of algorithms to apply, depending on the underlying scenario. The plane+parallax representation provides a *unified* approach to 2D and 3D scene analysis, with a strategy to gracefully bridge the gap between those two extremes [14]. Within the plane+parallax framework, the analysis always starts with 2D estimation (i.e., the homography estimation). When that is all the information available in the image sequence, that is where the analysis stops. The 3D analysis then gradually builds *on top* of the 2D analysis (in the form of planar-parallax displacements and shape-fluctuations w.r.t. the planar surface).

Kumar et al. [15] and Sawhney [19] used the plane+parallax framework to recover dense structure relative to the reference plane from *two* uncalibrated views. While their algorithm linearly solves for the structure directly from brightness measurements in two frames, it does not naturally extend to multiple frames. In this paper, we show how dense planar-parallax displacements and relative structure can be recovered directly from brightness measurements in *multiple* frames. As with camera-centered SFM methods, many of the ambiguities existing in the two-frame plane+parallax case of [15], [19] are resolved by extending the analysis to multiple frames. Our algorithm assumes as input a sequence of images in which a planar surface has been previously aligned with respect to a reference image (e.g., via one of the 2D parametric estimation techniques, such as [1], [9]). We do not assume that the camera calibration information is known. The output of the algorithm is:

1. The epipoles for all the images with respect to the reference image.

- *M. Irani is with the Department of Computer Science and Applied Math, The Weizmann Institute of Science, 76100 Rehovot, Israel. E-mail: irani@wisdom.weizmann.ac.il.*
- *P. Anandan is with Microsoft Corporation, One Microsoft Way, Redmond, WA 98052. E-mail: anandan@microscoft.com.*
- *At the time this work was done, M. Cohen was with the Department of Computer Science and Applied Math, The Weizmann Institute of Science, 76100 Rehovot, Israel.*

Manuscript received 23 July 1999; revised 2 May 2001; accepted 14 Aug. 2001. Recommended for acceptance by M. Black.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number 110290.

2. Dense 3D structure of the scene relative to a planar surface.
3. The correspondences of all the pixels across all the frames, which must be consistent with 1 and 2.

The estimation process uses the *exact* equations (as opposed to *instantaneous* equations, such as in [5], [21]) relating the residual parallax motion of pixels across multiple frames to the relative 3D structure and the camera epipoles. The 3D scene structure and the camera epipoles are computed directly from image measurements by minimizing the variation of image brightness across the views without precomputing a correspondence map.

As in the two-frame case of [15], [19], our technique relies on good prior alignment of the video frames with respect to a planar surface. This requires that a large enough real physical plane exist in the scene and be visible in all the video frames. Most indoor scenes have a planar surface (e.g., walls, floor, pictures, windows, etc.) and, in outdoor scenes, the ground or any large enough *distant* object can serve as a planar surface. If the planar surface captures a large enough image region, it can automatically be detected and aligned using robust methods for locking onto a dominant planar motion (e.g., [9]). However, if no such planar surface exists in the scene, then our method will not be applicable.

The remainder of the paper describes the algorithm and shows its performance on real and synthetic data. A shorter version of this paper appeared in [11].

2 The Plane+Parallax Decomposition

The induced 2D image motion of a 3D scene point between two images can be decomposed into two components [13], [10], [14], [15], [19], [19], [12], [3]:

1. The image motion of a reference planar surface Π (i.e., a homography).
2. The residual image motion, known as "planar parallax."

We begin with the plane+parallax motion equations of [14]. Let $\vec{p} = (x, y, 1)$ denote the image location (in homogeneous coordinates) of a point in one view (the "reference view") and let $\vec{p'} = (x', y', 1)$ be its coordinates in another view. Let $\mathbf{B}$ denote the homography of the plane Π between the two views. Let $\mathbf{B}^{-1}$ denote its inverse homography and $\mathbf{B}^{-1}{}_3$ be the third row of $\mathbf{B}^{-1}$. When the second image is warped toward the first image using the inverse homography $\mathbf{B}^{-1}$, then the point $\vec{p'}$ will move to $\vec{p_w}$ in the *warped image*:

$$\vec{p_w} = (x_w, y_w, 1) = \frac{\mathbf{B}^{-1}\vec{p'}}{\mathbf{B}^{-1}{}_3\vec{p'}}.$$

For 3D points on the plane Π, $\vec{p_w} = \vec{p}$. For 3D points off Π, $\vec{p_w} \neq \vec{p}$. It was shown [14] that:[1]

$$\vec{p'} - \vec{p} = (\vec{p'} - \vec{p_w}) + (\vec{p_w} - \vec{p}),$$

where $\vec{p'} - \vec{p_w}$ is the *planar* part of the image motion (the homography due to Π) and $\vec{p_w} - \vec{p}$ is the residual *planar parallax* displacement:

$$\vec{\mu} = \vec{p_w} - \vec{p} = -\gamma(t_3\vec{p_w} - \vec{t}). \quad (1)$$

$\gamma = H/Z$ represents the 3D structure of the point $\vec{p}$, H is the perpendicular distance (or "height") of the point from the reference plane Π, and Z is its depth with respect to the reference camera. All *unknown* calibration parameters are folded into the canceled homography $\mathbf{B}$ and into $\vec{t} = (t_1, t_2, t_3)$, which is the epipole in projective coordinates.

1. The notation we use here is slightly different than the one used in [14]. The change to projective notation is used to unify the two separate expressions provided in [14], one for the case of a finite epipole and the other for the case of an infinite epipole.

For any given pixel $\vec{p}$ in the reference image, the unknown corresponding pixel $\vec{p_w}$ in the other image appears on both sides of (1). We eliminate it from the right-hand side to obtain an expression of $\vec{p_w}$ (and of the parallax displacement) as a function of the pixel $\vec{p}$:

$$\vec{p_w} - \vec{p} = -\frac{\gamma}{1 + \gamma t_3}(t_3\vec{p} - \vec{t}). \quad (2)$$

This last expression will be used in our direct estimation algorithm.

3 Multiframe Parallax Estimation

Let $\{\Phi_j\}_{j=0}^{l}$ be $l+1$ images of a rigid scene, taken using cameras with unknown calibration parameters. Let Φ_0 denote the reference frame (usually the middle frame of the sequence). Let Π be a plane in the scene that is visible in all $l+1$ images (the "reference plane"). Using a technique similar to [1], [9], we estimate the homography of Π between the reference frame Φ_0 and each of the other frames $\{\Phi_j\}_{j=1}^{l}$. Warping the images by those homographies, $\{\mathbf{B}_j\}_{j=1}^{l}$, yields a new sequence of $l+1$ images, $\{I_j\}_{j=0}^{l}$, where the image of Π is aligned across all frames and $I_0 = \Phi_0$ is the reference image in the plane-stabilized sequence (for notational simplicity, we will often drop the subscript of the reference image I_0, i.e., $I = I_0$). The only residual image motion between reference frame I and the warped images, $\{I_j\}_{j=1}^{l}$, is the residual planar-parallax displacement $\{\vec{p_w^j} - \vec{p}\}_{j=1}^{l}$ due to 3D scene points that are *not* located on the reference plane Π. This residual planar parallax motion is what remains to be estimated.

Let $\vec{w^j} = (u^j, v^j)$ denote the first two coordinates of $\vec{p_w^j} - \vec{p}$ (the third coordinate is 0). From (2), we know that the residual parallax is:

$$\vec{w^j} = \begin{bmatrix} u^j \\ v^j \end{bmatrix} = -\frac{\gamma}{1 + \gamma t_3^j}\begin{bmatrix} t_3^j x - t_1^j \\ t_3^j y - t_2^j \end{bmatrix}, \quad (3)$$

where the superscripts j denote the parameters associated with the jth frame.

In the *two*-frame case, one can define $\alpha = \frac{\gamma}{1+\gamma t_3}$ and then the problem posed in (3) becomes a bilinear problem in α and in $\vec{t} = (t_1, t_2, t_3)$. This can be solved using a standard iterative method. Once α and $\vec{t}$ are known, γ can be recovered. A similar approach was used in [15] for shape recovery from two-frames. However, this approach does not extend to multiple (> 2) frames because α is *not* a shape invariant (as it depends on t_3) and, hence, varies from frame to frame. In contrast, γ *is* a shape invariant which is shared by all image frames. Our multiframe process directly recovers γ from *multiframe* brightness quantities.

The basic idea behind our direct estimation algorithm is that, rather than estimating l separate $\vec{w^j}$ vectors (corresponding to each frame) for each pixel, we can simply estimate a single γ (the shape parameter) which, for a particular pixel, is common to all the frames and a single $\vec{t^j} = (t_1, t_2, t_3)$ which, for each frame I_j, is common to all image pixels. There are two advantages in doing this:

- For n pixels over l frames, we reduce the number of unknowns from $2nl$ to $n + 3l$ and, more importantly,
- the recovered flow vector is constrained to satisfy the epipolar structure implicitly captured in (2).

This can be expected to significantly improve the quality of the recovered parallax flow vectors.

Our direct estimation algorithm follows the same computational framework outlined in [1] for the *quasi-parametric* class of models. The basic components of this framework are:

- pyramid construction,
- iterative estimation of global (motion) and local (structure) parameters,
- coarse-to-fine refinement.

Our algorithm is therefore as follows:

1. Construct pyramids from each of the images I_j and the reference frame I.
2. Initialize the structure parameter γ for each pixel and motion parameter $\vec{t^j}$ for each frame (usually, we start with $\gamma = 0$ for all pixels and $\vec{t^j} = (0, 0, 1)^T$ for all frames).
3. Starting with the coarsest pyramid level, at each level refine the structure and motion using the method outlined in Section 3.1.
4. Repeat this step several times (usually about four or five times per level).
5. Propagate the final values of the structure and motion parameters to the next finer pyramid level. Use these as initial estimates for processing the next level.
6. The final output is the structure and the motion at the finest pyramid level (at the resolution of the input images) and the residual parallax flow field synthesized from these.

It is important to note that accurate shape and motion recovery relies on having obtained good prior alignment of the sequence with respect to a viewed planar surface. Inaccurate plane alignment will naturally introduce errors in the shape and motion recovery. Moreover, as with any other iterative nonlinear minimization scheme, this process has the risk of locking onto local minima. However, this risk is significantly reduced by the coarse-to-fine minimization strategy (see [1]). Furthermore, the limited search space (only 3 degrees of freedom (d.o.f.) per frame in the global motion estimation step and only 1 d.o.f. per point in the local correspondence and shape estimation step—regardless of the number of frames) increases the overall robustness of the algorithm. Introducing additional assumptions, such as smoothness of the camera motion (i.e., temporal smoothness), would probably further condition the algorithm, but would also restrict it to a continuous set of images obtained by a video camera. We do not make such assumptions and can therefore also handle collections of still images obtained from multiple view-points.

Of the various steps outline above, the pyramid construction and the coarse-to-fine propagation of parameters are common to many techniques for motion estimation (e.g., see [1]), hence we omit the description of these steps. On the other hand, the refinement step is specific to our current problem. This is described next.

3.1 The Estimation Process

The inner loop of the estimation process involves refining the current values of the structure parameters γ (one per pixel in the reference image) and the motion parameters $\vec{t^j}$ (three parameters per frame). Let us denote the "true" (but unknown) values of these parameters by $\gamma(x,y)$ (at location (x,y) in the reference frame) and $\vec{t^j}$. Let $\vec{w^j}(x,y) = (u^j, v^j)$ denote the corresponding unknown true parallax flow vector. Let $\gamma_c, \vec{t_c^j}, \vec{u_c^j}$ denote the *current estimates* of these quantities. Let $\delta\gamma = \gamma - \gamma_c$, $\delta\vec{t^j} = (\delta t_1^j, \delta t_2^j, \delta t_3^j) = \vec{t^j} - \vec{t_c^j}$, and $\delta\vec{w^j} = (\delta u^j, \delta v^j) = \vec{w^j} - \vec{u_c^j}$. These refinements δ are estimated during each iteration.

Assuming brightness constancy (namely, that corresponding image points across all frames have a similar brightness value[2]) and a small residual displacement error $\delta\vec{w^j}$, we use the linearized brightness constancy equation [7]:

$$I_{t_j}(x,y) + I_x\delta u^j + I_y\delta v^j \approx 0, \tag{4}$$

where I_x, I_y denote the spatial derivatives of the reference image (at pixel location (x,y)) and I_{t_j} denotes the temporal derivative after compensating for the parallax vector $\vec{u_c^j}$ estimated in the previous iteration: $I_{t_j}(x,y) = I_j(x + u_c^j, y + v_c^j) - I(x,y)$. Substituting the expression for I_{t_j} and the expression for $\delta\vec{w^j} = \vec{w^j} - \vec{u_c^j}$ into (4) and regrouping the terms yields:

$$I_{\tau_j}(x,y) + I_x u^j + I_y v^j \approx 0, \tag{5}$$

where $I_{\tau_j}(x,y) \stackrel{\text{def}}{=} I_j(x + u_c^j, y + v_c^j) - I(x,y) - I_x u_c^j - I_y v_c^j$. If we now substitute the expression for the local parallax flow vector $\vec{w^j}$ given in (3), we obtain the following equation that relates the structure and motion parameters directly to image brightness information:

$$I_{\tau_j}(x,y) \ + \frac{\gamma(x,y)}{1 + \gamma(x,y)t_3^j}\Big(I_x(t_3^j x - t_1^j) + I_y(t_3^j y - t_2^j)\Big) \approx 0. \tag{6}$$

We refer to the above equation as the "epipolar brightness constraint."

Each pixel and each frame contributes one such equation where the unknowns are: the relative scene structure $\gamma(x,y)$ for each pixel (x,y) and the epipoles $\{\vec{t^j}\}_{j=1}^{l}$, one for each frame. Those unknowns are computed in two phases: In the "Local Phase," the relative scene structure $\gamma(x,y)$ is estimated separately for each pixel via least squares minimization over all frames simultaneously. This is followed by the "Global Phase," where each epipole $\vec{t^j}$ is estimated between the reference frame and each of the other frames, using least squares minimization over all pixels. These two phases are described in more detail below.

It should be noted that, although the linearized brightness constancy equation has a limited range of convergence (these usually iteratively converge to the correct solution when their initial guess is ≈ 2 pixels away from the correct solution), the multiresolution minimization approach extends its range of applicability to significantly larger displacements, up to $\approx 7\%$ of the image size. This limit occurs because we do not allow for images smaller than 30×30 at the highest resolution level (and $\frac{2}{30} \approx 7\%$). Thus, for example, if an image is 512×512, then recoverable parallax displacements are typically smaller than 35 pixels.

3.1.1 Local Phase

In the local phase, we assume all the epipoles are given (e.g., from the previous iteration) and we estimate the unknown scene structure γ from *all* the images. γ is a local quantity, but is common to all the images at a point. Each frame I_j provides one constraint of (6) on γ. When there are only two frames, there are n constraints (one from each pixel), but $n + 3$ unknowns (shape+epipole). Therefore, there is insufficient information for recovering structure and motion from the pointwise constraints of (6). This degeneracy is true for all two-frame direct methods [8], [4], [18]. In such cases, an additional assumption is usually made that the shape (γ in our case) is locally constant within a small (typically, 5×5) window around each pixel in the reference frame (e.g., see [17], [4], [15], [19], [2]). However, in the multiframe case, in general, there are enough constraints ($3l + n$ unknowns and ln constraints, where l is the number of frames), in which case, this window assumption is not necessary. However, using such a window-constraint provides additional numerical stability, especially if the different epipoles are very close to each other. Here, too, the window assumption will tend to smooth the results, but not as much as in the underconstrained two-frame case.

For each pixel (x,y) in the reference frame, we therefore seek a parameter $\gamma = \gamma(x,y)$ that will minimize the following multiframe-based error function:

$$E(\gamma) \stackrel{\text{def}}{=} \sum_j \sum_{\text{Win(x,y)}} \Big(\tilde{I_{\tau_j}}(1 + \gamma t_3^j) + \gamma\Big(\tilde{I_x}(t_3^j\tilde{x} - t_1^j) + \tilde{I_y}(t_3^j\tilde{y} - t_2^j)\Big)\Big)^2, \tag{7}$$

2. Note that, over multiple frames, the brightness of a scene point will tend to change somewhat, at least due to global illumination variation. We can handle this by using the Laplacian pyramid (as opposed to the Gaussian pyramid) or otherwise prefiltering the images (e.g., normalize to remove global mean and contrast changes) and applying the brightness constraint to the filtered images.

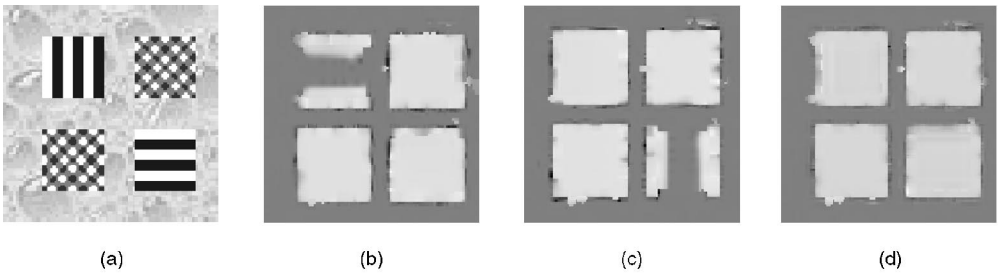

Fig. 1. Resolving aperture problem: (a) A sample image. (b) Shape recovery for pure vertical motion. Ambiguity along vertical bars. (c) Shape recovery for pure horizontal motion. Ambiguity along horizontal bars. (d) Shape recovery for a sequence with mixed motions. No ambiguity.

where the summation is over all pixels $(\tilde{x}, \tilde{y})$ in a 5×5 window $\mathrm{Win}(x, y)$ around (x, y), $\tilde{I}_{\tau_j} = I_{\tau_j}(\tilde{x}, \tilde{y})$, $\tilde{I}_x = I_x(\tilde{x}, \tilde{y})$, and $\tilde{I}_y = I_y(\tilde{x}, \tilde{y})$. Differentiating $E(\gamma)$ with respect to γ and equating it to zero yields a single linear equation that can be solved to estimate $\gamma(x, y)$. The error term $E(\gamma)$ was obtained by multiplying (6) by the denominator $(1 + \gamma t_3^j)$ to yield a *linear* expression in γ. Note that, without multiplying by the denominator, the local estimation process (after differentiation) would require solving a polynomial equation in γ whose order increases with l (the number of frames). Minimizing $E(\gamma)$, is in practice, equivalent to applying *weighted* least squares minimization on the collection of original (6), with weights equal to the denominators. We could apply *normalization* weights $\frac{1}{1+\gamma_c t_3^j}$ (where γ_c is the estimate of the shape at pixel (x, y) from the previous iteration) to the linearized expression in order to assure minimization of meaningful quantities (as is done in [24]), but, in practice, for the examples we used, we found it was not necessary to do so during the local phase. However, such a normalization weight was important during the global phase (see below).

3.1.2 Global Phase

In the global phase, we assume the structure γ is given at every pixel (e.g., from previous iteration), and we estimate, for each image I_j, the position of its epipole $\vec{t^j}$ with respect to the reference frame. We do so by minimizing the following error function for each epipole:

$$E(\vec{t^j}) \stackrel{\mathrm{def}}{=} \sum_{(x,y)} \left(W_j \left[I_{\tau_j}(1 + \gamma t_3^j) + \gamma \left(I_x(t_3^j x - t_1^j) + I_y(t_3^j y - t_2^j) \right) \right] \right)^2, \tag{8}$$

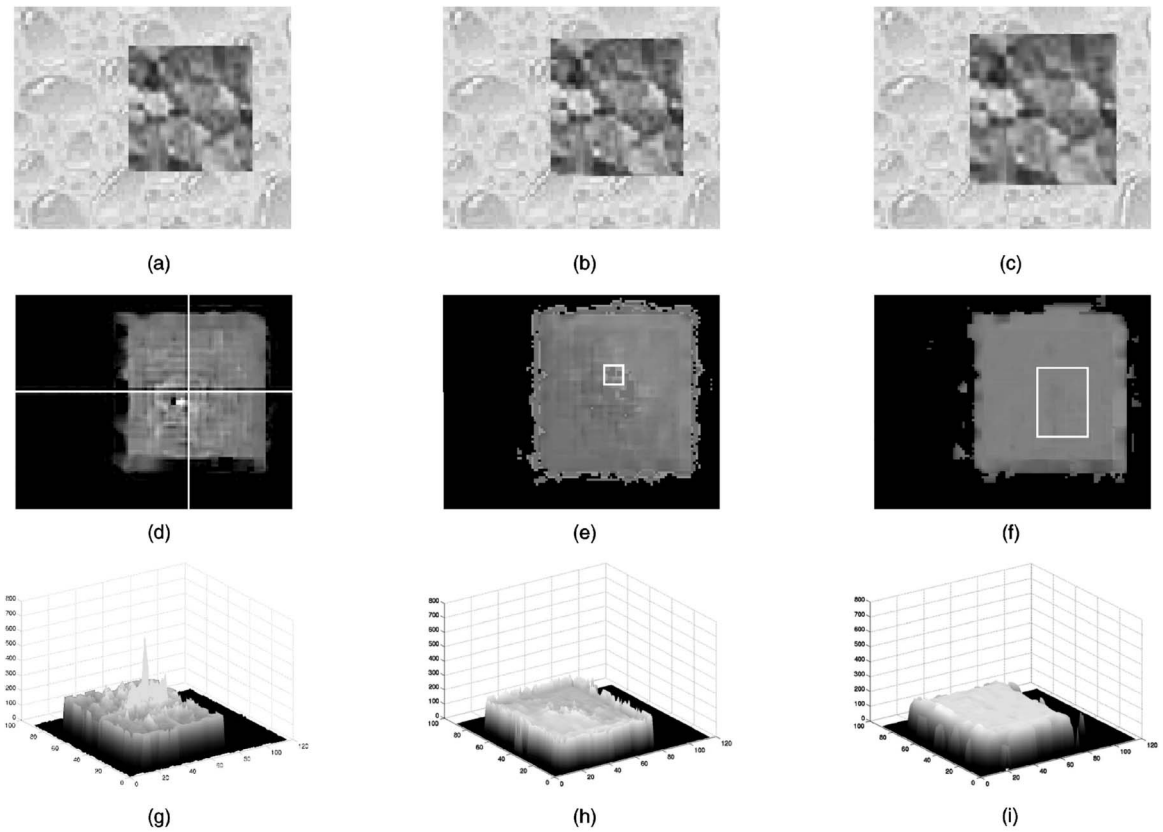

Fig. 2. Resolving epipole singularity in case of multiple epipoles. (a), (b), and (c) Three sample images from a 9-frame sequence with multiple epipoles (generated by simulating large forward motion and small sideways motion). (d) Shape recovery using two images. The cross-hair marks the location of the single epipole. There is singularity in the recovered shape in the vicinity of the epipole. A visual display of the recovered shape in the form of a 3D surface is shown in (g). (e) and (h) Shape recovery using three images with two different epipoles. The epipoles are within the small rectangular region marked in white. Note that epipole singularity disappears in the presence of multiple epipoles. (f) and (i) Shape recovery using five images with multiple epipoles. The accuracy of the recovered shape improves with the increase in the number of images (improved signal-to-noise ratio). Note, however, that the recovered depth-discontinuities are no longer sharp. This is due to the fact that, with more frames, there are larger (mistreated) occluded regions.

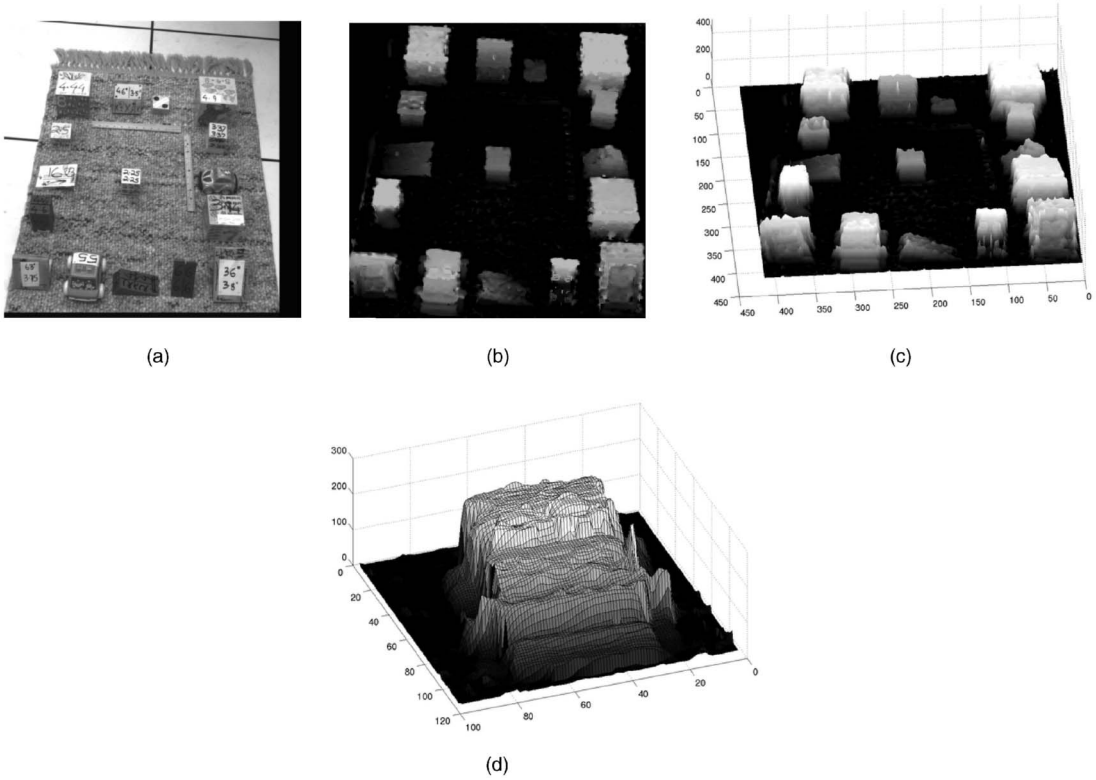

Fig. 3. Blocks sequence. (a) One image in the sequence. (b) The recovered shape (relative to the carpet plane). Brighter values correspond to taller points (larger γ). (c) The recovered shape (γ) shown from a different viewpoint. (d) A close-up mesh display of the toy car located at the bottom-left of the carpet.

where

$$W_j = W_j(x,y), I_x = I_x(x,y), I_y = I_y(x,y), I_{\tau_j} = I_{\tau_j}(x,y), \gamma = \gamma(x,y).$$

Note that, when $\gamma(x,y)$ are fixed, this minimization problem decouples into a set of independent individual minimization problems, each a function of one epipole $\vec{t^j}$ for the jth frame. The inside portion of this error term is similar to the one we used above for the local phase, with the addition of a scalar weight $W_j(x,y)$. The scalar weight is used to serve two purposes. First, if (8) did not contain the weights $W_j(x,y)$, it would be equivalent to a *weighted* least squares minimization of (6), with weights equal to the denominators $(1+\gamma(x,y)t_3^j)$. While this provides a convenient linear expression in the unknown $\vec{t^j}$, these weights are not physically meaningful and tend to skew the estimate of the recovered epipole. Therefore, in a fashion similar to [24], we choose the weights $W_j(x,y)$ to be $(1+\gamma(x,y)t_{3,c}^j)^{-1}$, where the γ is the updated estimate from the local phase, whereas the $t_{3,c}^j$ is based on the current estimate of $\vec{t^j}$ (from the previous iteration).

The scalar weight also provides us an easy way to introduce additional robustness to the estimation process in order to reduce the contribution of pixels that are potentially outliers. For example, we can use weights based on residual misalignment of the kind used in [9].

4 Multiframe versus Two-Frame Estimation

The algorithm described in Section 3 extends the plane+parallax estimation to multiple frames. The benefits of multiframe processing over two-frame processing are:

1. Overcoming the *aperture problem* from which the two-frame estimation often suffers.
2. Resolving the singularity of shape recovery in the vicinity of the epipole (we refer to this as the *epipole singularity*).
3. Improved signal-to-noise performance that is obtained due to having a larger set of independent samples.

These benefits are common to all SFM methods. We demonstrate these advantages in the context of the plane+parallax framework.

4.1 Eliminating the Aperture Problem

The residual parallax lies along epipolar lines (centered at the epipole, see (3)). When the image gradient at an image point is perpendicular to the epipolar line passing through the point, then the Brightness Constancy Constraint line (5) is parallel to the epipolar line, and the parallax displacement at that point cannot be uniquely determined (and, hence, also its structure). However, when multiple images with *multiple epipoles* are used, this ambiguity is resolved because the image gradient at a point can be perpendicular to at most one epipolar line passing through it. This observation was also made by [5], [21].

To demonstrate this, we used a sequence composed of nine images (105×105 pixels) of four squares (30×30 pixels) moving over a stationary textured background (which plays the role of the aligned reference plane). The four squares have the same motion: First, they were all shifted to the right (one pixel per frame) to generate the first five images and then they were all shifted down (one pixel per frame) to generate the next four images. The width of the stripes on the squares is five pixels. A sample frame is shown in Fig. 1a (the fifth frame).

The horizontal motion and the vertical motion have an epipole at infinity ($(\infty, 52.5]$ and $[52.5, \infty)$, respectively). Fig. 1b shows the depth map that results from applying the algorithm to sequences with purely vertical motion. (Dark gray corresponds to the reference plane and light gray corresponds to elevated scene parts, i.e., the squares). The structure for the square with vertical bars is not estimated well, as expected, because the epipolar constraints are parallel to those bars. Fig. 1c shows the same problem for horizontal bars under horizontal motion. Fig. 1d shows the depth map that results when multiple directions of motion are present. Note that now the shape recovery does not suffer from the aperture problem.

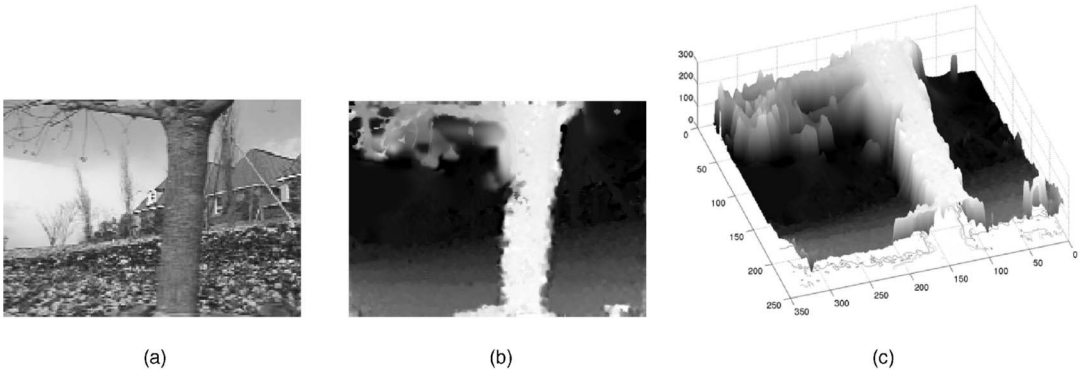

Fig. 4. Flower-garden sequence. (a) One frame from the sequence. (b) The recovered shape (relative to the facade of the house). Brighter values (larger γ) correspond to points farther from the house. A visual display of the recovered shape γ in the form of a 3D surface is shown in (c).

4.2 Epipole Singularity

From the planar parallax (3), it is clear that the structure γ cannot be determined at the epipole because, at the epipole: $t_3^j x - t_1^j = 0$ and $t_3^j y - t_2^j = 0$. The recovered structure in the *vicinity* of the epipole will also be unreliable. However, when there are multiple epipoles, this ambiguity disappears. The singularity at one epipole is resolved by information from another epipole.

To test this, we compared the results for the case of one epipole (i.e., two-frames) to cases with multiple epipoles at different locations. Results are shown in Fig. 2. The sequence that we used was composed of images of a square elevated from a reference plane and the simulated motion was a forward motion with a slight sideways translation to allow for different epipoles. Figs. 2a, 2b, and 2c show three sample images from the sequence. Figs. 2d and 2g show singularity around the epipole in the two-frame case. Figs. 2e, 2f, 2h, and 2i show that the singularity at the epipoles is *eliminated* when there is more than one epipole. Using more images also increases the signal-to-noise ratio and further improves the shape reconstruction. However, there are stray errors near depth-discontinuities due to the fact that with more frames, occluded and disoccluded regions become larger.

5 Real World Examples

This section provides experimental results of applying our algorithm to three real sequences. Even though, in some of these sequences, the original image motion (before plane alignment) was large (e.g., due to camera rotations), after plane alignment, the residual planar-parallax displacement were small (typically, no more than 10 pixels). The reference frame was usually chosen to be the middle frame to minimize the sizes of planar parallax displacements between the reference frame and any other frame in the sequence. Even though, in general, the algorithm can recover planar parallax displacements of up to 7 percent of the image size (see Section 3.1), by working with small planar-parallax displacements, we avoid the need to explicitly treat occluding boundaries.

Fig. 3a shows one of three images taken by a still camera (extracted from the "block" sequence of [15]). The second and the third images were captured after moving the camera sideways and forward, respectively. The images were aligned with respect to the carpet (the reference plane). Fig. 3b shows the recovered structure. The brightness reflects the magnitude of the structure parameter γ. Brighter gray levels correspond to taller points relative to the carpet. Fig. 3c shows the recovered shape from a different view point. Fig. 3d shows a close-up mesh plot of the toy car at the bottom-left of the carpet. Comparison of these results to the ones in [15] shows that the multiframe algorithm recovers more of the finer details than the two-frame algorithm.

Fig. 4 shows an example of shape recovery for a sequence of five frames (part of the flower garden sequence). The camera moves sideways in this sequence. The reference plane is the facade of the house. Fig. 4a shows the reference frame from the sequence. Figs. 4b and 4c show the recovered structure. Note the gradual change of depth in the field of flowers.

Fig. 5 shows an example of shape recovery for a sequence of five frames. The reference plane is the flat ground region in front of the building. The sequence was taken by a hand-held video camera while walking toward the building. The epipoles in this case fall inside the frames, but there are multiple (nearby) epipoles. Fig. 5a shows one frame from the sequence. Figs. 5b and 5c show the recovered structure. The shape of the building wall is not fully recovered because of lack of texture in that region.

6 Conclusion

We presented an algorithm for estimating dense planar-parallax displacements from multiple uncalibrated views. The image

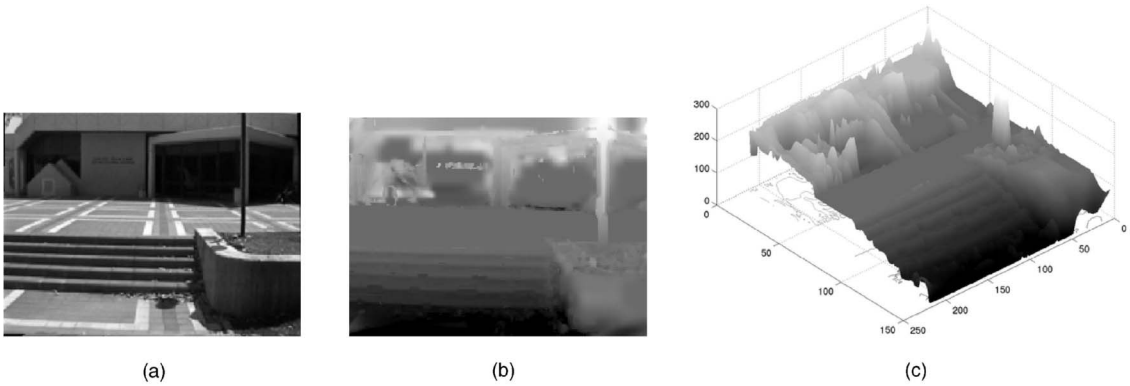

Fig. 5. Stairs sequence. (a) One frame from the sequence. (b) The recovered shape (relative to the ground surface in front of the building). A visual display of the recovered shape γ in the form of a 3D surface is shown in (c).

displacements, the 3D structure, and the camera epipoles are estimated *directly* from image brightness variations across multiple frames. The algorithm relies on having good prior alignment of the sequence with respect to a viewed planar surface. This algorithm extends the two-frames plane+parallax estimation algorithm of [15], [19] to multiple frames. The benefits of multiframe processing over two-frame processing are: 1) overcoming the aperture problem, 2) resolving the singularity of shape recovery in the vicinity of the epipole, and 3) improved signal-to-noise performance. These were illustrated in the paper.

ACKNOWLEDGMENTS

The work of Michal Irani and Meir Cohen was supported by the Israeli Ministry of Science (grant no. 1229).

REFERENCES

[1] J.R. Bergen, P. Anandan, K.J. Hanna, and R. Hingorani, "Hierarchical Model-Based Motion Estimation," *Proc. European Conf. Computer Vision,* pp. 237-252, May 1992.
[2] T. Brodsky and C. Fermüller, and Y. Aloimonos, "Shape from Video," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. B, pp. 146-15, June 1999.
[3] C. Criminisi, I. Reid, and Z. Zisserman, "Duality, Rigidity, and Planar Parallax," *Proc. European Conf. Computer Vision,* vol. II, 1998.
[4] K.J. Hanna, "Direct Multi-Resolution Estimation of Ego-Motion and Structure from Motion," *Proc. Workshop Visual Motion,* pp. 156-162, Oct. 1991.
[5] K.J. Hanna and N.E. Okamoto, "Combining Stereo and Motion for Direct Estimation of Scene Structure," *Proc. Int'l Conf. Computer Vision,* pp. 357-365, 1993.
[6] R.I. Hartley, "In Defense of the Eight-Point Algorithm," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 19, no. 6, pp. 580-593, June 1997.
[7] B.K.P. Horn and B.G. Schunk, "Determining Optical Flow," *Artificial Intelligence,* vol. 17, nos. 1-3, pp. 185-203, Aug. 1981.
[8] B.K.P. Horn and E.J. Weldon, "Direct Methods for Recovering Motion," *Int'l J. Computer Vision,* vol. 2, no. 1, pp. 51-76, June 1988.
[9] M. Irani, B. Rousso, and S. Peleg, "Computing Occluding and Transparent Motions," *Int'l J. Computer Vision,* vol. 12, no. 1, pp. 5-16, Jan. 1994.
[10] M. Irani and P. Anandan, "Parallax Geometry of Pairs of Points for 3D Scene Analysis," *Proc. European Conf. Computer Vision, A,* pp. 17-30, Apr. 1996.
[11] M. Irani, P. Anandan, and M. Cohen, "Direct Recovery of Planar-Parallax from Mulitple Frames," *Proc. Workshop Vision Algorithms,* Sept. 1999.
[12] M. Irani, B. Rousso, and P. Peleg, "Recovery of Ego-Motion Using Region Alignment," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 19, no. 3, pp. 268-272, Mar. 1997.
[13] M. Irani, P. Anandan, and D. Weinshall, "From Reference Frames to Reference Planes: Multi-View Parallax Geometry and Applications," *Proc. European Conf. Computer Vision,* vol. II, pp. 829-845, 1998.
[14] M. Irani and P. Anandan, "A Unified Approach to Moving Object Detection in 2D and 3D Scenes," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 20, no. 6, pp. 577-589, June 1998.
[15] R. Kumar, P. Anandan, and K. Hanna, "Direct Recovery of Shape from Multiple Views: A Parallax Based Approach," *Proc. Int'l Conf. Pattern Recognition,* pp. 685-688, Oct. 1994.
[16] H.C. Longuet-Higgins and K. Prazdny, "The Interpretation of a Moving Retinal Image," *Proc. Royal Soc. London B,* vol. 208, pp. 385-397, 1980.
[17] B.D. Lucas and T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision," *Proc. Seventh Int'l Joint Conf. Artificial Intelligence,* pp. 121-130, 1981.
[18] S. Negahdaripour and B.K.P. Horn, "Direct Passive Navigation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 9, no. 1, pp. 168-176, Jan. 1987.
[19] H.S. Sawhney, "3D Geometry from Planar Parallax," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 929-934, June 1994.
[20] A. Shashua and N. Navab, "Relative Affine Structure: Theory and Application to 3D Reconstruction from Perspective Views," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 483-489, 1994.
[21] G.P. Stein and A. Shashua, "Model-Based Brightness Constraints: On Direct Estimation of Structure and Motion," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 400-406, 1997.
[22] R. Szeliski and S.B. Kang, "Direct Methods for Visual Scene Reconstruction," *Proc. Workshop Representations of Visual Scenes,* 1995.
[23] P.H.S. Torr, "Geometric Motion Segmentation and Model Selection," *Proc. Royal Soc. London A,* vol. 356, pp. 1321-1340, 1998.
[24] Z. Zhang, "Determining the Epipolar Geometry and Its Uncertainty: A Review," *Int'l J. Computer Vision,* vol. 27, no. 2 pp. 161-195, 1997.