FAST STRUCTURE FROM MOTION RECOVERY APPLIED TO 3D IMAGE STABILIZATION

S. Srinivasan and R. Chellappa

Center for Automation Research, University of Maryland
College Park, MD 20742-3275
{shridhar,rama}@cfar.umd.edu

ABSTRACT

In this paper, we address 3D image stabilization using a framework for the estimation of scene structure from a monocular motion field. We show that our algorithm rapidly and accurately determines the focus of expansion (FOE) in an optical flow field. This involves computing the least squares error of a large system of equations without actually solving the equations, to generate an error surface that describes the goodness of fit as a function of the hypothesized FOE. Consequently, we recover the rotational motion which we use to perform 3D image stabilization.

1. INTRODUCTION

Electronic image stabilization is a differential process that steadies an image sequence acquired by a moving camera by compensating for camera motion. Commonly, 2D image stabilization techniques apply interframe translation, similarity, affine or perspective transformation to stabilize the sequence. 2D techniques perform poorly when the scene is richly structured in 3D and the camera motion is not restricted to pan. When a 3D scene is being imaged by an unsteady camera, the resulting image motion is a result of the camera parallax motion (translation) as well as camera rotation. Since the parallax shift cannot be compensated for and is often deliberate, it is the rotation that is desired to be annulled. Unfortunately, the 3D structure of the scene enters the equations and does not permit rotation to be resolved independent of scene depth.

The extraction of 3D structure of a moving scene from a sequences of images is termed as the *structure from motion* (SFM) problem. The solution to this problem is a key step in the monocular rangefinding, 3D image stabilization, obstacle avoidance and time to collision. Mathematical analysis of SFM shows the nonlinear interdependence of structure and motion given observations on the image plane. While SFM has received considerable attention by researchers, the proposed solutions tend to have several shortcomings. Algorithms that eliminate the depth field by cross multiplication [1, 2, 3, 4, 5, 6], are not very stable. Assuming smoothness of the depth field or optical flow field is not always valid, more so when there is noise or discontinuity in the flow estimates. Thus, differentiating flow fields [7, 8] is unacceptable. Nonlinear optimization based solutions [9, 10] are relatively stable in the presence of noise. However, minimizing a nonlinear cost function exposes the solution to the pitfalls of local minima and slow convergence.

In this paper, we present a fast partial search technique for locating the focus of expansion (FOE) of a motion field. The FOE is hypothesized to lie within a bounded square on the image plane. For each candidate location on a discrete sampling of the plane, we generate a linear system of equations for estimating the remaining unknowns which are the rotational velocity and inverse depth map. We compute the least squares error of the system *without actually solving the equations*, to generate an error surface that describes the goodness of fit as a function of the hypothesized focus of expansion. The minimum of the error surface occurs at a discrete location very close to the true FOE. We use this FOE estimate to compute the rotation for performing 3D stabilization. Since the linear system used to solve for depth and rotation at each candidate location of the FOE is stable, bounded perturbances in the optical flow estimates lead to a deterministic, bounded offset of the error surface minimum from zero. Thus, noise resilience is inherent to this linear formulation.

This paper is organized as follows: the SFM problem is formulated in section 2 and an outline of our approach is presented in section 3. We discuss the application of our SFM solution to 3D stabilization with an experiment in section 4.

This work was partially supported by ONR-MURI grant N00014-95-1-0521.

2. PROBLEM FORMULATION

Assuming a camera centered coordinate system with linear dimensions normalized by the focal length *i.e.* $f = 1$, the optical flow $u(x, y)$, $v(x, y)$ observed on the image plane is related to the 3-D translation $t = (t_x, t_y, t_z)$, rotation $\omega = (\omega_x, \omega_y, \omega_z)$ and scaled inverse depth map $h(x, y) = \frac{t_z}{Z(x,y)}$ according to

$$\begin{aligned} u(x, y) &= -(x - x_f)h(x, y) + xy\omega_x - (1 + x^2)\omega_y + y\omega_z \\ v(x, y) &= -(y - y_f)h(x, y) + (1 + y^2)\omega_x - xy\omega_y - x\omega_z \end{aligned} \tag{1}$$

where $(x_f, y_f) \stackrel{\text{def}}{=} (\frac{t_x}{t_z}, \frac{t_y}{t_z})$ is known as the *focus of expansion* (FOE). The nonlinear coupling of the unknown depthmap $h(x, y)$ with translation t precludes a simple solution to (1). Many techniques build on eliminating $h(x, y)$ from (1) by cross multiplication, but this step gives rise to product terms of unknowns, and renders the solution very sensitive to noise in flow estimates. However, if the focus of expansion is estimated reasonably well, the remaining unknowns are recovered by solving an overdetermined set of linear equations, which is a well-conditioned process. The technique we propose in this paper locates the FOE within a bounded search space in an accurate and computationally efficient manner, given an input optical flow field.

2.1. Partial Search

Suppose the nonlinear set of equations for which a solution is desired is given by

$$\mathbf{f}(\mathbf{x}) = \mathbf{0}, \qquad \mathbf{x} \in \Re^M, \mathbf{0} \in \Re^K, K \geq M \tag{2}$$

Exhaustive search of a solution $\mathbf{x}_e$ involves (i) enumerating a finite set of candidate solutions $\mathcal{X} = \{\mathbf{x}_0, \mathbf{x}_1, \ldots\}$ that adequately cover the solution space, (ii) computing an error metric (*e.g.* $\|\mathbf{f}(\mathbf{x}_i)\|^2$) which associates each candidate solution $\mathbf{x}_i$ with a compliance measure and (iii) locating the minimum error and corresponding candidate solution, which for the squared error metric is

$$\mathbf{x}_e = \arg\min_{\mathbf{x}\in\mathcal{X}} \{\|\mathbf{f}(\mathbf{x})\|^2\}$$

In general, the order of complexity of exhaustive search is proportional to $|\mathcal{X}|$, which can get unmanageably large as the dimensionality M of $\mathbf{x}$ increases.

Another approach to searching for all the components of the solution is to enumerate only a few components and solve for the remaining components based on the hypothesis. For example in (2), assume that the argument $\mathbf{x}$ can be partitioned as $\mathbf{x}' = (\mathbf{a}' \ \mathbf{b}')$, $\mathbf{a} \in \Re^{M_1}$, $\mathbf{b} \in \Re^{M_2}$, $M_1 + M_2 = M$ and given $\mathbf{a}_i$, (2) can be solved for the remaining components $\mathbf{b}_i$ with a small number of computations. We define $\mathbf{a}$ as the *search component* and $\mathbf{b}$ as the *dependent component* of $\mathbf{x}$. *Partial search* of a solution $\mathbf{x}_p$ is performed by (i) enumerating a finite set of candidate partial solutions $\mathcal{A} \in \{\mathbf{a}_0, \mathbf{a}_1, \ldots\}$ that adequately cover the search component space, (ii) computing the dependent component $\mathbf{b}_i$ corresponding to each $\mathbf{a}_i \in \mathcal{A}$ that closely satisfies (2), (iii) computing an error metric, for example $\|\mathbf{f}([\mathbf{a}_i' \ \mathbf{b}_i']')\|^2$ for the squared error case and (iv) picking the candidate solution corresponding to the minimum error. Step (ii) can be defined as a minimization over the continuous space of permissible $\mathbf{b}$ of the same squared error metric to formulate the partial search solution $\mathbf{x}_p{}' = (\mathbf{a}^{*\prime} \ \mathbf{b}^{*\prime})$ as

$$\mathbf{a}^* = \arg\min_{\mathbf{a}\in\mathcal{A}} \min_{\mathbf{b}} \left\|\mathbf{f}\begin{pmatrix}\mathbf{a}\\ \mathbf{b}\end{pmatrix}\right\|^2 \quad \text{and} \quad \mathbf{b}^* = \arg\min_{\mathbf{b}} \left\|\mathbf{f}\begin{pmatrix}\mathbf{a}^*\\ \mathbf{b}\end{pmatrix}\right\|^2$$

Partial search separates the problem into a search and a minimization. In general, the complexity of the original problem is proportional to the cardinality $|\mathcal{A}|$ of $\mathcal{A}$, and to the number of operations required to compute $\mathbf{b}_i$ given $\mathbf{a}_i$. In specific situations, as in the approach used here, the search complexity can be reduced even further by not explicitly evaluating $\mathbf{b}^*$ for every $\mathbf{a}^*$.

3. APPROACH

Let the true FOE be (x_f, y_f). Assuming that the flow field is of size $N \times N$ and all N^2 flow estimates are available, the optical flow at pixel location $i, j \in \{0, 1, \ldots, N-1\}^2$ is given by

$$u_{i,j} = -(x_{i,j} - x_f)h_{i,j} + x_{i,j}y_{i,j}\omega_x - (1 + x_{i,j}^2)\omega_y + y_{i,j}\omega_z$$
$$v_{i,j} = -(y_{i,j} - y_f)h_{i,j} + (1 + y_{i,j}^2)\omega_x - x_{i,j}y_{i,j}\omega_y - x_{i,j}\omega_z$$

where $x_{i,j} = \frac{i-(N-1)/2}{f}$, $y_{i,j} = \frac{j-(N-1)/2}{f}$, and $\{h, u, v\}_{i,j} = \{h, u, v\}(x_{i,j}, y_{i,j})$. Define

$$\begin{aligned}
r_{i,j} &= (x_{i,j}y_{i,j} \ \ -(1 + x_{i,j}^2) \ y_{i,j})' \\
s_{i,j} &= (1 + y_{i,j}^2 \ \ -x_{i,j}y_{i,j} \ \ -x_{i,j})' \\
\mathbf{Q} &= [r_{0,0} \ s_{0,0} \ r_{0,1} \ \ldots \ s_{N-1,N-1}]' \\
h &= -(h_{0,0} \ h_{0,1} \ \ldots \ h_{N-1,N-1})' \\
\mathbf{u} &= (u_{0,0} \ v_{0,0} \ \ldots \ v_{N-1,N-1})' \\
\mathbf{P}(x_f, y_f) &= \begin{bmatrix} x_{0,0} - x_f & & \\ y_{0,0} - y_f & & \\ & \ddots & \\ & & x_{N-1,N-1} - x_f \\ & & y_{N-1,N-1} - y_f \end{bmatrix} \\
\mathbf{A}(x_f, y_f) &= \begin{bmatrix} \mathbf{P}(x_f, y_f) & \mathbf{Q} \end{bmatrix} \\
\mathbf{x}_0 &= \begin{bmatrix} h \\ \omega \end{bmatrix}.
\end{aligned}$$

This allows us to consolidate the motion equations for all individual flow vectors in the brief form

$$\begin{bmatrix} \mathbf{P}(x_f, y_f)\mathbf{Q} \end{bmatrix} \begin{bmatrix} h \\ \omega \end{bmatrix} = \mathbf{A}(x_f, y_f) \ \mathbf{x}_0 = \mathbf{u}.$$

Replacing the unknowns x_f, y_f and $\mathbf{x}_0$ by the hypothesized variables x_h, y_h and $\mathbf{x}$ we get the general condition

$$\mathbf{A}(x_h, y_h) \ \mathbf{x} \to \mathbf{u} \tag{3}$$

where the true solution exactly satisfies (3). We now define a squared error cost function $C(x_h, y_h, \mathbf{x}) = \|\mathbf{A}(x_h, y_h) \ \mathbf{x}\|_2^2$. Since

$$C(x_h, y_h, \mathbf{x}) \geq 0 \quad \text{and} \quad C(x_f, y_f, \mathbf{x}_0) = 0 \tag{4}$$

(i) the true solution to the system (3) minimizes the cost function $C(\ )$ and (ii) all minimizers of $C(\ )$ satisfy (4) exactly. Thus, we have reduced the original problem to

$$\min_{x_h, y_h, \mathbf{x}} C(x_h, y_h, \mathbf{x}) = \min_{x_h, y_h} \min_{\mathbf{x}} C(x_h, y_h, \mathbf{x}) \tag{5}$$

The inner minimization occurs at the least squares (LS) solution $\mathbf{x}_{LS}$ of $\mathbf{A}(x_h, y_h) \ \mathbf{x} \to \mathbf{u}$. It is not difficult to see that even with coarse discretization, the number of free variables is too large to permit exhaustive search. Referring to §2.1, we see that partial search is an ideal technique for solving (5).

In order to perform partial search, we set $\{x_h, y_h\}$ to be the search component and $\mathbf{x}$ to be the dependent component. We discretize the search component space at midway locations between four pixels over the entire image area, in line with our (relaxable) assumption that the FOE lies within the image. The LS solution $\mathbf{x}_{LS}$ of $\mathbf{A}(x_h, y_h) \ \mathbf{x} \to \mathbf{u}$ satisfies $\mathbf{A}'\mathbf{A}\mathbf{x}_{LS} = \mathbf{A}'\mathbf{u}$ where the arguments of $\mathbf{A}$ has been dropped. Define diagonal matrix

$$\mathbf{D} = \mathbf{P}'\mathbf{P} = \text{Diag}\left\{\bar{x}_i^2 + \bar{y}_i^2\right\}$$

where $\bar{x}_i$ and $\bar{y}_i$ are functions of (x_h, y_h)

$$\bar{x}_i = x_{\lfloor i/N \rfloor, i \bmod N} - x_h \quad \text{and} \quad \bar{y}_i = y_{\lfloor i/N \rfloor, i \bmod N} - y_h$$

When $\mathbf{D}$ is nonsingular, $\mathbf{x}_{LS}$ is unique. Introducing matrices $\mathrm{M} \in \Re^{2N \times 2N}$ and $\hat{\mathrm{M}} \in \Re^{3\times 3}$ defined as

$$\mathrm{M} = (\mathbf{I} - \mathbf{P}\mathbf{D}^{-1}\mathbf{P}') \quad \text{and} \quad \hat{\mathrm{M}} = \mathbf{Q}'\mathrm{M}\mathbf{Q} \tag{6}$$

we get

$$\mathbf{x} = \begin{bmatrix} \mathbf{D}^{-1}\mathbf{P}'(\mathbf{I} - \mathbf{Q}\hat{\mathrm{M}}^{-1}\mathbf{Q}'\mathrm{M}) \\ \hat{\mathrm{M}}^{-1}\mathbf{Q}'\mathrm{M} \end{bmatrix} \mathbf{u} \tag{7}$$

The squared error can be shown to simplify to

$$\|\mathbf{A}\mathbf{x} - \mathbf{u}\|^2 = \mathbf{u}'\mathrm{M}\mathbf{u} - \mathbf{u}'\mathrm{M}\mathbf{Q}\hat{\mathrm{M}}^{-1}\mathbf{Q}'\mathrm{M}\mathbf{u} \tag{8}$$

The obvious strategy of computing the least squared error, or equivalently, of performing the inner minimization in (5), is to explicitly solve the linear system for the unknown $\mathbf{x}$ and use this estimate to evaluate the squared error. Even after taking into account the sparseness of $\mathbf{A}$, the overall complexity including the outer minimization search is an unacceptably high $\mathcal{O}(N^4)$. The crux of our algorithm lies in the fact that we can further exploit the structure of $\mathbf{A}$ so that the errors can be computed directly, without computing the solution $\mathbf{x}$ explicitly. Moreover, the least squared errors for all the candidate hypotheses can be computed in a single step using Fourier techniques, which leads to an overall complexity of $\mathcal{O}(N^2 \log N)$. While this seems incredible at the first sight since factoring out N^2 from the complexity introduced by the outer search leaves $\mathcal{O}(\log N)$ - a quantity insufficient even for vector addition, it is the simultaneous estimation of all errors in the search space that allows such a low overall complexity. Proof and details of our algorithm are presented together with an in-depth evaluation in [11].

4. 3D STABILIZATION

Stabilization is a differential process that compensates for the "unwanted" motion in an image sequence, which in typical situations is the rotational motion of the camera with respect to an inertial frame of reference. Stabilization of an image sequence is important for improving the throughput of a human or an automatic algorithm examining the sequence for "targets". Image stabilization is also an important step for motion super-resolution, which is the process of enhancing the resolution of an image from multiple shifted views of the scene. Videocompression is yet another application of stabilization. 2D image stabilization algorithms that employ interframe shift, similarity, affine or perspective transformations perform poorly when there is significant depth variation and camera translation.

The first step in image stabilization is the computation of the motion field. Since our emphasis here is on the recovery of rotation (as opposed to recovery of depth) we choose the modeled optical flow algorithm [12] for computing the motion field. This technique has the advantage of accuracy and speed at the expense of yielding a flow field described by a low order model, which is not a disadvantage for recovering rotation. Next, the FOE estimation algorithm proposed here is applied to the computed flow field and the location corresponding to the minimum error in (8) is picked as the FOE $(\hat{x}, \hat{y})$. A correction of $\frac{1}{2}$ pixel is applied to each direction, to undo the effect of staggering. Once the FOE is estimated, the corresponding angular velocity is available with no extra computations.

Figs. 1 (a) and (b) show the first and hundredth frames of the Martin Marietta sequence. The camera is mounted looking ahead on a vehicle as it traverses unpaved terrain. There is sufficient texture in most of the image, and the interframe displacements are small, permitting differential optical flow computation. The FOE and rotation angles are computed using our algorithm. The estimated pitch, yaw and roll plots are shown in Figs. 1 (c), (d) and (e) respectively. These are in excellent visual compliance with the results obtained by Yao [13].

Fig. 2 demonstrates the effect of 3D stabilization - (a) shows the twentieth frame of the sequence. We chose this frame as it displays higher than average angular deviation from the first frame. With no stabilization, the difference between the twentieth and first frames is shown in Fig. 2(b). The fully stabilized image (compensated for roll, pitch and yaw) and its difference from the first frame are shown in Figs. 2(c) and (d) respectively. In the difference image, areas near the camera show larger deviations than those at a distance. This is the effect of translation of the camera.

Since our algorithm actually computes the three rotation angles for each frame, we can go one step further to perform "selective stabilization". For instance, if we wish to compensate only for camera roll, we disregard the effects of pitch and yaw while derotating the frames. Fig. 2(e) shows the twentieth frame of the Martin Marietta sequence, stabilized for roll only. The difference from the first frame is shown in Fig. 2(f). The parallel horizon and mountain profile in this figure reveals the unstabilized pitch and yaw motion. Extending this concept, one can selectively stabilize for certain frequencies of motion to eliminate handheld jitter while preserving deliberate camera pan, etc.

Although 3D SFM has received much attention over the years a fast and accurate solution has evaded researchers. We believe that the approach presented here by us is a satisfactory solution to this challenging problem. In conclusion, our solution using fast partial search of the FOE proves to work well in the application area of 3D image stabilization.

5. REFERENCES

[1] R.Y. Tsai and T.S. Huang. Estimating 3-d motion parameters of a rigid planar patch i. *ASSP*, 29(12):1147–1152, December 1981.

[2] X. Zhuang, T.S. Huang, N. Ahuja, and R.M. Haralick. A simplified linear optical flow-motion algorithm. *CVGIP*, 42(3):334–344, June 1988.

[3] X. Zhuang, T.S. Huang, N. Ahuja, and R.M. Haralick. Rigid body motion and the optic flow image. In *CAIA84*, pages 366–375, 1984.

[4] A.M. Waxman, B. Kamgar-Parsi, and M. Subbarao. Closed-form solutions to image flow equations for 3d structure and motion. *IJCV*, 1(3):239–258, October 1987.

[5] A. Mitiche, X. Zhuang, and R.M. Haralick. Interpretation of optical flow by rotational decoupling. In *CVWS87*, pages 195–200, 1987.

[6] N.C. Gupta and L.N. Kanal. 3-d motion estimation from motion field. *AI*, 78(1-2):45–86, October 1995.

[7] H.C. Longuet-Higgins and K. Prazdny. The interpretation of a moving retinal image. *RoyalP*, B-208:385–397, 1980.

[8] A.M. Waxman and S. Ullman. Surface structure and three-dimensional motion from image flow kinematics. *IJRR*, 4(3):72–94, 1985.

[9] A.R. Bruss and B.K.P. Horn. Passive navigation. *CVGIP*, 21(1):3–20, January 1983.

[10] G. Adiv. Determining 3-d motion and structure from optical flow generated by several moving objects. *PAMI*, 7(4):384–401, July 1985.

[11] S. Srinivasan. Extracting structure from optical flow using the fast error search technique. Technical Report CAR-TR-893, Univ. of Maryland, 1998.

[12] S. Srinivasan and R. Chellappa. Optical flow using overlapped basis functions for solving global motion problems. In *ECCV98*, 1998.

[13] Y. S. Yao. *Electronic Stabilization and Feature Tracking in Long Image Sequences*. PhD thesis, Univ. of Maryland, 1996. available as Tech. Rep. CAR-TR-790.

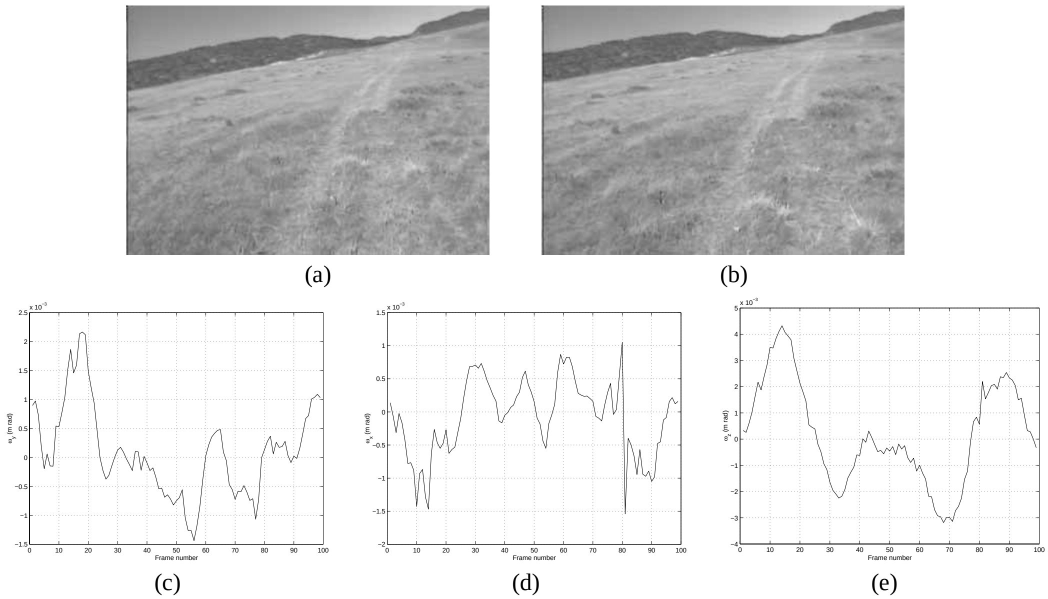

Figure 1: 3D Stabilization: (a) first and (b) hundredth frame of Martin Marietta sequence, (c) pitch, (d) yaw and (e) roll as a function of frame number

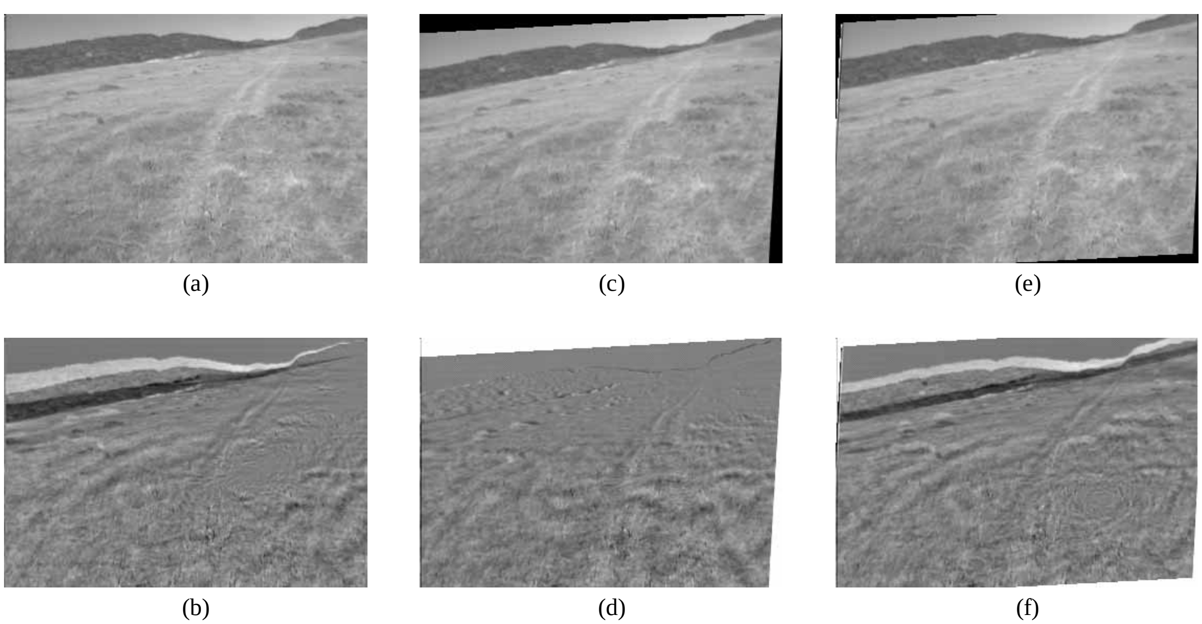

Figure 2: 3D Stabilization: (a) twentieth frame of Martin Marietta sequence, (b) difference between first and twentieth frame with no stabilization, (c) fully stabilized twentieth frame, (d) stabilized difference, (e) stabilized only for roll, (f) difference between roll-stabilized frame and the first frame of the sequence