The Recursive Multi-Frame Planar Parallax Algorithm

Abstract

This paper presents a method for obtaining accurate dense elevation and appearance models of terrain using a single camera on-board an aerial platform, which has many applications including geographical information systems, robot path planning, immersion and visualization, and surveying for scientific purposes such as watershed analysis. When given geo-registered images, the method can compute terrain maps on-line in real time. This algorithm, called the Recursive Multi-frame Planar Parallax algorithm, is a recursive extension of Irani et al.'s multi-frame planar parallax framework and in theory, with perfectly registered imagery, it will produce range data with error expected to increase between linearly and with the square root of the range, depending on image properties and whether other constants such as framerate and vehicle velocity are held constant. This is an improvement over stereo systems whose expected errors are proportional to the square of the range. We show experimental evidence on synthetic imagery and on a real video sequence taken in an experiment for autonomous helicopter landing.

1 Introduction

In this paper we address the problem of recovering an accurate digital elevation map (DEM) with a passive sensor such as a camera, and doing so at close to framerate. Digital terrain models have applications ranging from visualization (e.g. Google Earth and NASA's World Wind [1]) and hydrological analysis [2], to robot path planning [3]. Though active technologies such as radar and LIDAR are available and tend to have very high accuracy, passive sensors are cheaper, usually have a smaller form factor, consume less power, and being emissionless are more difficult to detect. For a small platform such as a micro air vehicle (MAV) [4], passive sensors may be the only viable option.

We plan to use DEMs to evaluate landing sites for an unmanned aerial vehicle, such as a helicopter. Therefore the DEM needs to be accurate. A suitable area is a clearing at least 200ft in diameter, clear of obstacles larger than a soccer ball, and having a slope no greater than 4 degrees. In addition, we must usually perform this selection task from an altitute of at least 300ft above ground level (AGL). We present an algorithm for accurately estimating a digital elevation map using the parallax present in multiple images taken from a moving vehicle whose egomotion has been previously obtained.

A common method for passive range estimation is the use of a stereo camera pair [5, 6]. Stereo systems, however, cannot attain the desired accuracy given constraints on resolution and platform space. For a stereo system the variance of depth estimates ($\hat{z}$) grow *quartically* with depth—that is, the variance of expected differences between the true depth and the estimate $\mathbb{E}[(z - \hat{z})^2] = \mathsf{O}(z^4)$—which in our case is unacceptable.[1]

In a rigid scene, however, we can treat multiple images—obtained as the vehicle moves through space—as a multiple camera system. We describe here a recursive method based on multi-frame planar parallax framework [7, 8] that uses multiple image pairs, which have baselines larger than that physically attainable on the platform, to reduce variance. We show that in theory we can recover range with a variance that is asymptotically quadratic in the depth.

The novelty of this result is a method which is (*i*) *recursive* in the sense that the cost of incorporating measurements from a new image is proportional only to the number of pixels in the image and does *not* depend on the number of frames already seen; (*ii*) it is dense, in the sense that it provides estimates of depth for any sufficiently textured region; (*iii*) it is more accurate than instantaneous stereo; and (*iv*) it is direct (see the discussion, pro: [9], con: [10]), by which we mean that the algorithm does not depend on the matching of features, but rather expresses a cost function directly in terms of the image, and the gradients of the cost function are computed by linearization of the brightness constancy constraint.

Other methods feature some, but not all, of these elements. For example, bundle adjustment [11] is the optimal estimator for determining structure and motion from multiple views when correspondences are known and correct. However, it provides neither dense structure, nor the ability to recover structure recursively. Stereo and multi-baseline methods are the most favored methods for recovering dense structure, e.g. [12, 13]. Regarding stereo error, Matthies and Shafer [6], and later Xiong and Matthies [14], investigate sources of error in stereo. The primary drawback of stereo is its inaccuracy as discussed above. Planar parallax

[1] At a range of 100 meters, a baseline of 1 meter, and focal length $f = 500$, standard deviation of $\hat{z}$ is approximately 20 meters.

is a related framework based on registration using a plane in the scene [7]. Irani et al. [8] propose a method for estimating planar parallax, and from it the depth, using more than two views, though it is not recursive. Their work is the closest in spirit to ours. We improve on this method in that we present a Recursive Multi-Frame Planar Parallax (RMFPP) algorithm. Other closely related works are Zucchelli et al. [15], in which they sparsely estimate structure and motion and update a dense structure map; and Matthies et al. [16], in which they propose a Kalman filter for updating disparities.

Here we employ a direct method that takes advantage of the observation that for a smoothly moving camera, the initial small-baseline disparity estimates may lead to inaccurate range estimates, they are nevertheless accurate disparity measurements. Furthermore, later improvement in the range estimates will not drastically change the refined small-baseline disparities. Therefore, in the cost function described in [8], the image need not be rewarped and relinearized. Instead the linearized terms are kept and encoded in sufficient statistics (mean and variance) and a 1D Kalman filter is run for each pixel.

The method described here is subject to several assumptions. We reiterate that this method updates estimates of structure only; we assume that the positions and orientations of the cameras have been previously determined. We have *not* found a method to recursively estimate structure *and* motion which is both dense and direct—its possibility seems unlikely but remains open. We also rely on the usual assumptions: validity of the brightness constancy constraint within some regions, rigidity of the scene, and the presence of sufficient texture. Finally, the motions between the camera positions should be sufficiently small—though this constraint can be lessened by the use of image pyramids.

2 Analysis of Depth Errors

In this section we model range errors in a stereo pair and in an idealized multiple-baseline system. For a fixed-baseline stereo pair, the predicted standard deviation is quadratic in the range, i.e. $E[(\hat{z}-z)^2]^{1/2} = O(z^2)$. Using the simple case of a camera moving in a straight line at constant velocity, we show that by appropriately weighting pair-wise estimates we can theoretically attain errors between $O(z)$ and $O(z^{1/2})$, depending on the correlation between disparity estimates.

Fixed-baseline stereo. Consider a rectified stereo pair separated by a baseline b, observing a point at depth z. The relationship between disparity and depth is given by $z = f\,b/\delta$, where δ is the disparity and f is the focal length. Ignoring quantization errors and mismatches, we can obtain an approximation of the variance of the depth estimate at any single pixel, namely:

$$\begin{aligned} \text{var}(\hat{z}) &= E\big[(z-\hat{z})^2\big] = E\left[\left(\frac{f\,b}{\delta} - \frac{f\,b}{\delta+\epsilon}\right)^2\right] \\ &\approx \frac{z^4}{f^2\,b^2}\,\text{var}(\epsilon), \end{aligned} \tag{1}$$

where ϵ is the error in the disparity estimate, and where we have taken the first-order Taylor series approximation in ϵ about 0 in the second equation. Generally $\text{var}(\epsilon)$ depends on the image derivatives along the scanline. Using a single stereo measurement would be ill-advised at distances greater than $f \cdot b$—corresponding to a disparity equal to one pixel—above which the predicted standard deviation would become larger than the range.

Can we achieve greater accuracy in range using only passive means? We can fight uncertainty by increasing b, or by utilizing the independence in the measurements, if there is any. Often it is not practical to increase the baseline between two vehicle-mounted cameras beyond some fixed limit, but on a moving aerial vehicle we can get wider baselines for free. If the camera's motion can be recovered (either by structure-from-motion methods or by some combination of inertial and GPS systems) then we can use multiple measurements to reduce error. If the measurements are to some degree independent, then we can drive down uncertainty.

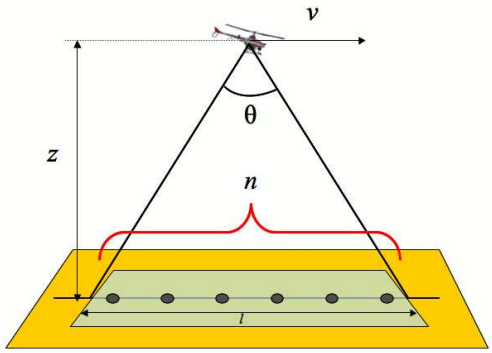

Figure 1: Idealized flight for purposes of analyzing range accuracy.

Monocular camera at constant velocity and framerate. We consider the following situation, depicted in Figure 1: an aerial vehicle flying at constant altitude above a terrain with average relative height z meters, at constant horizontal velocity v. Assume that the position of the vehicle is known without error at all times, and that a downward pointing onboard camera with field of view θ captures an image every t seconds. The baseline between the first and k-th frame is $b_k = kvt$ and the number of times a point a distance z from the camera is seen is at most $n = 2z\tan\frac{\theta}{2}/vt \approx \theta z/v$.

Let us use a single pair of frames separated by a wider baseline and determine the resulting range error. Choose the first frame and the $c \cdot n$-th frame, where $c < 1$—

that is, a frame a constant fraction in between the first and the last frame in which there is overlap in the two views of the ground. Then the predicted accuracy (variance) is $z^2\,\mathrm{var}(\epsilon)/c^2 f^2 \theta^2$.[2] This is a significant improvement over the fixed baseline case. Instead of being quadratic in the range, here the predicted standard deviation is proportional to the range.

Can we do better than error linear in range by using multiple measurements? We can get the most out of multiple measurements when they are known to be independent. However, in simulated experiments with $1/f$ noise, we find that errors in disparity estimates are correlated with correlation coefficient up to 0.6; i.e. if $\epsilon_{i,j}$ is the error in the estimate of disparity between frames i and j, then we find there to be correlation between errors $\epsilon_{1,2}$ and $\epsilon_{1,3}$. Let us construct a linear estimator which is blind to the generally unknown correlation coefficient, and then evaluate the estimator's squared error while assuming a non-zero correlation.

Let $\hat{z}_k$ be the measurement of depth using the k-th estimated disparity, $\hat{\delta}_k = \delta_k + \epsilon_k$, between frames 1 and k. If $\sigma_k^2 = z^4\,\mathrm{var}(\epsilon)/f^2 k^2 T^2 v^2$ is $\hat{z}_k$'s variance, which we have calculated using formula (1), then the minimum variance linear estimate of z using $m = c \cdot n$ estimates $\hat{z}_k$ is $\hat{z} = \sum_{k=1}^{m} w_k \hat{z}_k$, where $w_k = \sigma_k^{-2} / \sum_{j=1}^{m} \sigma_j^{-2}$. Since the first image does not change it is reasonable to assume that, for any fixed pixel in the first image, $\mathrm{var}(\epsilon_k)$ does not change with k. However, as we have discussed, we cannot guarantee statistical independence of the ϵ_k.

Predicted error. It is difficult, if not impossible, to empirically determine the correlation among ϵ_k for real images. However, if we have evidence that the correlation is bounded, then we can gauge the effect of correlation on the accuracy of the linear estimator. Assume that, for some $0 \le \rho \le 1$, $E[\epsilon_j \epsilon_k] < \rho\,\mathrm{var}(\epsilon)$ for all j and k. The expected squared error, as a function of ρ, for the linear estimator defined above is:

$$E[(\hat{z}-z)^2] = \frac{z^4\,\mathrm{var}(\epsilon)}{f^2}\left(\sum_{1\le k\le m}\frac{w_k^2}{b_k^2} + \rho\sum_{1\le j<k\le m}\frac{w_j w_k}{b_j b_k}\right) \approx \left(c_1\, z + c_2\,\rho\, z^2\right)\cdot \mathrm{var}(\epsilon) \quad (2)$$

for $z >> vt$, and where c_1 and c_2 depend on t, v, θ, and c, and are given in formula **(??)** of the appendix. If there is no correlation, i.e. $\rho = 0$, then the resulting estimator has range error proportional to the square root of the range. When there is non-zero correlation, then the variance is asymptotically linear. In Figure 2 we plot formula (2) using the following values: $t = 3.75^{-1}\,s$, $v = 14\,m\,s^{-1}$, $\theta = 50°$, $c = 1/4$, and $\rho = 0$, $1/3$, $2/3$ and 1, and z ranging between 20 and $200\,m$; note that at $z = 17.11\,m$ the number of measurements is $m = 1$.

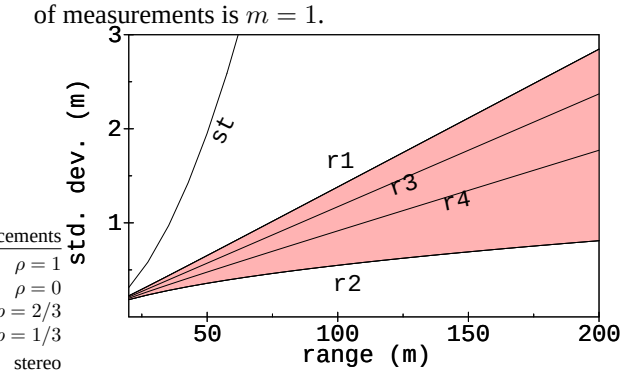

Figure 2: Predicted standard deviations for stereo and multi-baseline as a function of range. The single curve shows range error for stereo when using the smallest baseline (the first two frames in the idealized aerial image sequence) for the values reported in the text. The (red) shaded area shows the range of errors of a multiple-baseline linear estimator for ρ ranging between 0 and 1.

To summarize, by combining multiple measurements and wider baselines we can conceivably obtain between linear and quadratic variance in range estimates as a function of true range, a vast improvement over quartic variance obtained with a stereo pair. In the rest of the paper we describe a recursive algorithm for densely reconstructing terrain, in which we try to achieve this performance.

3 Multi-Frame Planar Parallax

The multi-frame planar parallax (MFPP) method is a generalization of stereo rectification to more than two frames that was first described by Sawhney [7], and was later extended by Irani et al. [8]. Whereas stereo rectification yields images where the disparity (or optical flow) is parallel to scanlines and is inversely proportional to range, MFPP registration yields images such that the ratio of disparities along epipolar lines can be expressed in terms of a view-independent shape parameter that encodes depth.

Suppose a camera takes images $i = 1, \ldots, m$ of a rigid scene. Let the rotation and translation taking the first coordinate system to the i-th one be given by $(\mathsf{R_i}, \boldsymbol{T_i})$, so that $\mathsf{R}_1 = \mathsf{I}$ and $\boldsymbol{T_1} = \mathbf{0}$. We choose a virtual reference plane in the scene and then construct the homographies H_i which transform the i-th frame such that points on the reference plane have zero disparity, and such that $\mathsf{H}_1 = I$ so that the first frame acts as a reference frame. Let $\boldsymbol{N}$ be the unit normal of the reference plane in the coordinate system of the first camera, and let d_i be the perpendicular distance of the i-th viewpoint from the virtual plane. The homographies $\mathsf{H_i}$ which transfer the i-th view to the reference view via the

[2] Note that f and θ are usually coupled with the resolution of the camera, but are constant for a fixed camera.

reference plane are given by:

$$\mathsf{H_i} \;=\; \mathsf{K}\left(\mathsf{R_i} - \frac{1}{d_1}\boldsymbol{T_i}\boldsymbol{N}^T\right)^{-1}\mathsf{K}^{-1}, \qquad (3)$$

where K is the constant intrinsic calibration matrix of the camera. Let $\boldsymbol{E_i} = (e_x^{(i)}, e_y^{(i)}, e_z^{(i)})^T = -\mathsf{K}\mathsf{R_i}^T\boldsymbol{T_i}$ be the image of the i-th viewpoint in the first view, i.e. one of the epipoles in the stereo pair defined by the first and i-th views.

Suppose that $\boldsymbol{X} \in \mathbb{R}^3$ is a point in space in the coordinate system of the first camera. Let $\boldsymbol{p_i} = (x_i, y_i)$ for $i = 1, .., m$ be X's projection into each image, and define $\boldsymbol{p} = \boldsymbol{p_1}$. Let $\pi(x, y, z) = (x/z, y/z)$ and $\pi^*(x, y) = (x, y, 1)$. If $\boldsymbol{X}$ lies on the reference plane, then:

$$\boldsymbol{p} \;=\; \underbrace{\pi\left(\,\mathsf{H_i}\,\pi^*(\,\boldsymbol{p_i}\,)\right)}_{\boldsymbol{p_i}'}. \qquad (4)$$

In general, X does not have to lie on the reference plane, so this equation is not necessarily satisfied. Nevertheless, the difference between $\boldsymbol{p}$ and $\boldsymbol{p_i}'$ must be parallel to the epipolar line through $\boldsymbol{p}$ and $\pi(\boldsymbol{E_i})$. Sawhney [7] proves that if $\boldsymbol{p} = (x, y)$, then:

$$\boldsymbol{\delta}_i(\boldsymbol{p}, \gamma) \;=\; \boldsymbol{p} - \boldsymbol{p_i}' \;=\; \frac{-\gamma}{d_i - \gamma\, e_z^{(i)}}\begin{bmatrix} e_z^{(i)}x - e_x^{(i)} \\ e_z^{(i)}y - e_y^{(i)} \end{bmatrix} \qquad (5)$$

where $\gamma = \mathcal{G}(\boldsymbol{p})$ is a view-independent scalar defined at each point in the first image. The difference $\boldsymbol{\delta}_i(\boldsymbol{p}, \gamma)$ is called the *parallax*. In this formulation the set of parallax vectors at a single point $\boldsymbol{p}$ are expressed in terms of the known $\boldsymbol{E_i}$'s and d_i's, and the unknown but view-independent γ. Furthermore, one can show that $\gamma = h/z$, where z is the depth of $\boldsymbol{X}$ in the first view and $h = \boldsymbol{N}^T\boldsymbol{X} + d_1$ is the signed perpendicular distance of $\boldsymbol{X}$ from the reference plane. We can recover z from γ using the fact that $\boldsymbol{X} = z\,\mathsf{K}^{-1}\,\pi^*(\,\boldsymbol{p}\,)$ (see appendix).

4 Non-recursive Cost Function

The geometric model given in equation (5) gives us an image model, an analog to the brightness constancy constraint. We will try to satisfy this constraint by optimizing over the space of functions $\gamma(\boldsymbol{p})$. The brightness constancy constraint is of the form:

$$\mathcal{I}_i^r(\boldsymbol{q}) \;-\; \mathcal{I}_1\big(\boldsymbol{q} + \boldsymbol{\delta}_i(\boldsymbol{p}, \mathcal{G}(\boldsymbol{p}))\big) \;=\; 0, \qquad (6)$$

where $\boldsymbol{\delta}_i(\boldsymbol{p}, \gamma)$ is the parallax generator defined in (5), $\mathcal{G}(p)$ is the function giving a value of γ for each pixel, and $\mathcal{I}_i^r$ is the plane-registered image obtained by warping $\mathcal{I}_i$ by H_i:

$$\mathcal{I}_i^r(\boldsymbol{q}) \;=\; \mathcal{I}_i\big[\,\pi\left(\mathsf{H_i}^{-1}\,\pi^*(\boldsymbol{q})\right)\big], \qquad (7)$$

where functions π and π^* are defined above.

Given the images $\mathcal{I}_i$ and parallax generators $\boldsymbol{\delta}_i$, the goal is to find a function $\mathcal{G}$ such that (6) is true for all values of $\boldsymbol{p}$. Irani et al. [8] proposed to minimize the residual of the brightness constancy constraint over all images and all pixels, as in the following expression:

$$\varepsilon(\mathcal{G}) = \sum_i \iint\limits_{\substack{\boldsymbol{p}\,\boldsymbol{q}\;\in \\ \mathrm{win}(\boldsymbol{p})}} \Big[\,\mathcal{I}_i^r(\boldsymbol{q}) - \mathcal{I}_1\big(\boldsymbol{q} + \boldsymbol{\delta}_i(\boldsymbol{p}, \mathcal{G}(\boldsymbol{p}))\big)\Big]^2 d\boldsymbol{q}\, d\boldsymbol{p}, \qquad (8)$$

where integrals are a convenient notation for sums and $\mathrm{win}(\boldsymbol{p})$ is a $k \times k$ window centered at $\boldsymbol{p}$. They minimize the functional in an iterative fashion, alternating between optimzation over the space of shape functions $\mathcal{G}$ and optimization over the set of epipoles, until convergence. They compute gradients by linearizing the image about an initial estimate.

5 Recursive Cost Function

Estimating parallax in real-time necessitates a recursive algorithm, by which we mean an algorithm which has a constant time (per pixel) update, such as the linear Kalman filter. In the formulation above, after every new estimate $\hat{\gamma}^k$ during gradient descent, we need to re-warp all previous images. Furthermore, with each additional frame, we need to perform an additional rewarping at every iteration. We cannot afford such a computation, which grows linearly with the number of frames.

A recursive algorithm is possible because of the following observation. Consider the example from Section 2 of images taken uniformly along a line. For a single point in the scene we have the disparities from the reference image to the k-th image: $\delta_k(z) = kvt/z$. At baselines that are small relative to z, depth estimates are very inaccurate because $\delta_k'(z)$ is small. Conversely, large changes to z induce relatively little variation in δ_k. Though the true z may be far from initial estimates, re-warping is not necessary because it will not result in a "large" change. Said another way, though the depth may be inaccurate, the flow will generally always have the same relatively low error, and the warping will generally be accurate (have low residual compared with the reference image) at a majority of the pixels. However, we count on small changes to effect depth estimates, and so linearizations (intensities and local derivatives) of the past images are maintained in a second order approximation of the cost function. The condition for this working, then, is that updates to the disparity must not exceed the range of the linear approximation of the image at each point. Thus regions of images which are *too* "textured" will pose problems.

The goal is to turn the minimization of the batch functional ε into a recursive procedure, where the addition of

frames results in a warping procedure which iterates only over the new frame, with as little loss of accuracy as possible. To do this, first we decompose ε as defined in (8) into a set of individual pixel cost functions as follows:

$$c_i(\boldsymbol{p},\gamma) \;=\; \int\limits_{\boldsymbol{q}\;\in \mathrm{win}(\boldsymbol{p})} r_i(\boldsymbol{p},\gamma(\boldsymbol{p}))^2\, d\boldsymbol{q}$$

where r_i is the residual:

$$r_i(\boldsymbol{p},\gamma) \;=\; \mathcal{I}_i^r(\boldsymbol{q}) - \mathcal{I}_1\big(\boldsymbol{q} + \boldsymbol{\delta}_i(\boldsymbol{p},\gamma)\big)\big).$$

The total non-recursive cost functional is then the sum over all images and all pixels: $\varepsilon(\mathcal{G}) = \sum_i \int_{\boldsymbol{p}} c_i(\boldsymbol{p}, \mathcal{G}(\boldsymbol{p}))$.

In the recursive formulation we propose a cost function which is linearized in past terms but iterated until convergence on the latest image. We denote by $\mathcal{G}^{(i)}$ the final estimate of $\mathcal{G}$ after the last iteration on the i-th frame. Then, for example after the i-th frame has arrived, we define $c^{(i)}$ to be the per-pixel cost up to and including the i-th frame (not to be confused with the image-specific term c_i):

$$\begin{aligned} c^{(i)}(\boldsymbol{p},\gamma) \;&=\; \sum_{j\le i} c_j(\boldsymbol{p},\gamma) \\ &= c_i(\boldsymbol{p},\gamma) + \sum_{j<i} \underbrace{c_i\Big[\boldsymbol{p}, \mathcal{G}^{(j)}(\boldsymbol{p}) + (\mathcal{G}^{(j)}(\boldsymbol{p}) - \gamma)\Big]}_{\text{compute 2nd order Taylor series in } \gamma \text{ at } \mathcal{G}^{(j)}(\boldsymbol{p})} \\ &\approx c_i(\boldsymbol{p},\gamma) + \Sigma A^{(i-1)}(\boldsymbol{p})\,\gamma^2 + \Sigma B^{(i-1)}(\boldsymbol{p})\,\gamma + \Sigma C^{(i-1)}(\boldsymbol{p}) \end{aligned}$$

where $\Sigma A^{(i)}$, $\Sigma B^{(i)}$, and $\Sigma C^{(i)}$ are respectively the coefficients of γ^2, γ and 1 in the Taylor series expansion. Their expressions, except for $\Sigma C^{(i)}$ which has no bearing on the gradient, are given below:

$$\Sigma A^{(i)}(\boldsymbol{p}) = \Sigma A^{(i-1)}(\boldsymbol{p}) + \int\limits_{\boldsymbol{q}\in\mathrm{win}(\boldsymbol{p})} \big[r_i'\big(\boldsymbol{q}, \mathcal{G}^{(i-1)}(\boldsymbol{p})\big)\big]^2 d\boldsymbol{q} \quad (9)$$

$$\begin{aligned} \Sigma B^{(i)}(\boldsymbol{p}) = \Sigma B^{(i-1)}(\boldsymbol{p}) + \int\limits_{\boldsymbol{q}\in\mathrm{win}(\boldsymbol{p})} \Big[2\, r_i'\big(\boldsymbol{q}, \mathcal{G}^{(i-1)}(\boldsymbol{p})\big)\cdot \\ \big(r_i\big(\boldsymbol{q}, \mathcal{G}^{(i-1)}(\boldsymbol{p})\big) - \mathcal{G}^{(i-1)}(\boldsymbol{p}) r_i'\big(\boldsymbol{q}, \mathcal{G}^{(i-1)}(\boldsymbol{p})\big)\big)\Big]^2 d\boldsymbol{q} \end{aligned} \quad (10)$$

were $r_i' = \partial r_i/\partial\gamma$, both of $\Sigma A^{(0)}$ and $\Sigma B^{(0)}$ are zero, and we ignore second-order terms of r_i. Note that the last term in ΣA, for example, is the result of a linearization (of r_i) about a different point from the terms before it, namely whichever was the latest estimate of γ at $\boldsymbol{p}$.

The final result is the following cost functional for the frames up to and including the i-th one:

$$\varepsilon^{(i)}(\mathcal{G}) \;=\; \int_{\boldsymbol{p}} c^{(i)}(\mathcal{G}(\boldsymbol{p}))\, d\boldsymbol{p}$$

which, very roughly, is linear in some sufficient statistics in the first $i-1$ frames, and remains non-linear in the i-th frame. In this sense, the algorithm is implicitly an iterated extended Kalman filter, where the mean γ at each pixel is the minima of the cost function given by $-\Sigma B^{(i)}/\Sigma A^{(i)}$ (pixel-wise), and the variances given by $1/\Sigma A^{(i)}$ (again, pixel-wise).

The procedure, to be fully outlined in Section 7 is this: with $\mathcal{G}^{(i-1)}$ we iteratively minimize $\varepsilon^{(i)}$, in the j-th step arriving at a new intermediate estimate $\mathcal{G}^{(i;j)}$. Upon convergence the last $\mathcal{G}^{(i;j)}$ becomes $\mathcal{G}^{(i)}$.

6 Computation of γ and Flows

Now that we have a cost function to minimize and have found and linearized its gradient, we discuss the recursive computation of the cost coefficients $\Sigma A^{(i)}, \Sigma B^{(i)}$. Given the current estimate of the shape parameter γ for a pixel $\boldsymbol{p} = (x, y)$ in the reference image:

$$\gamma \quad = \quad -\frac{\Sigma B^{(i)}(\boldsymbol{p})}{\Sigma A^{(i)}(\boldsymbol{p})}, \quad (11)$$

the current estimates of the flows at that pixel are determined as in equation (5), i.e. $(u, v) = \boldsymbol{\delta}_i(\boldsymbol{p}, \gamma)$. We warp $\mathcal{I}_i^r$ (as definined in equation (7)) using these flows, e.g. $\mathcal{I}_w(x, y) = \mathcal{I}_i^r(x - u, y - v)$; if the flow estimates are perfect and the brightness constancy constraint holds, then $\mathcal{I}_w$ will be indistinguishable from $\mathcal{I}_1$. Using this warped image and the derivatives $\mathcal{I}_x$ and $\mathcal{I}_y$ of the reference image, we can calculate the addend terms of equations (9)-(10), denoted A and B, by:

$$\mathsf{A}(\boldsymbol{p}) = 2\,\frac{d_i\, a}{b}\left(I_\tau(\boldsymbol{p}) - \frac{e_z^{(i)}\, a\, \gamma^2}{b}\right),\; \mathsf{B}(\boldsymbol{p}) = 2\,\frac{d_i^2\, a^2}{b^2} \quad (12)$$

where

$$\begin{aligned} \mathcal{I}_\tau(\boldsymbol{p}) \;&=\; \mathcal{I}_w(\boldsymbol{p}) - \mathcal{I}_1(\boldsymbol{p}) - \mathcal{I}_x(\boldsymbol{p})\; u - \mathcal{I}_y(\boldsymbol{p})\; v \\ a \;&=\; \left(e_z^{(i)}\, x - e_x^{(i)}\right) \mathcal{I}_x(\boldsymbol{p}) + \left(e_z^{(i)}\, y - e_y^{(i)}\right) \mathcal{I}_y(\boldsymbol{p}) \\ b \;&=\; \left(e_z^{(i)}\, \gamma - d_i\right)^2 \end{aligned}$$

and $\mathcal{I}_x$ and $\mathcal{I}_y$ are the x and y derivatives of $\mathcal{I}_1$, respectively.

7 Complete Algorithm

We are now ready to give the complete Recursive Multi-Frame Planar Parallax algorithm. The state variables $\Sigma\mathsf{A}$, $\Sigma\mathsf{B}$, residual, and numValid, each the same dimensions as the images $\mathcal{I}_i$, are initialized to 0. Every image $\mathcal{I}_i$ utilizing the reference image $\mathcal{I}_1$ is processed as follows:

process_image($\mathcal{I}_i$, R_i, $\boldsymbol{T}_i$, $\mathcal{I}_1$, $\Sigma\mathsf{A}$, $\Sigma\mathsf{B}$, residual,

numValid):

1: $\forall\ x, y,\ \mathsf{valid}\,(x,y)\ =\ 1$
2: $\mathcal{I}_i^r$ computed using eq. (7), clear valid for mappings outside of image; erode valid by 2 pixels
3: $\forall\ x, y$ s.t. $\mathsf{valid}\,(x,y)\ ==\ 1$, estimate γ as in eq. (11) using $\{\Sigma\mathsf{A}, \Sigma\mathsf{B}\}$
4: for up to n_{iter} iterations
5: $\{\Sigma\mathsf{AIter}, \Sigma\mathsf{BIter}\}\ =\ \{\Sigma\mathsf{A}, \Sigma\mathsf{B}\}$
6: calculate flows as in eq. (5)
7: $\mathcal{I}_w$ warped from $\mathcal{I}_i^r$ using flows
8: calculate $\{\mathsf{A}, \mathsf{B}\}$ as in eq. (12)
9: $\forall\ x, y$ s.t. $\mathsf{valid}\,(x,y)\ ==\ 1$, $\Sigma\mathsf{AIter}\ +=\ \mathrm{average\_window\_filter}\,(\mathsf{A})$ $\Sigma\mathsf{BIter}\ +=\ \mathrm{average\_window\_filter}\,(\mathsf{B})$
10: $\forall\ x, y$ s.t. $\mathsf{valid}\,(x,y)\ ==\ 1$, estimate γ as in eq. (11) using $\{\Sigma\mathsf{AIter}, \Sigma\mathsf{BIter}\}$
11: if average change in valid region of $\mathcal{G}$ was small, then break
12: end for
13: calculate flows as in eqn (5)
14: $\mathcal{I}_w$ warped from $\mathcal{I}_i^r$ using flows
15: $\forall\ x, y$ s.t. $\mathsf{valid}\,(x,y)\ ==\ 1$, $\mathsf{residual}\,(x,y)\ +=\ |\mathcal{I}_1\,(x,y)\ -\ \mathcal{I}_w\,(x,y)\,|$ $\mathsf{numValid}\,(x,y) + +$
16: $\{\Sigma\mathsf{A},\ \Sigma\mathsf{B}\}\ +=$ last averaged $\{\mathsf{A}, \mathsf{B}\}$ from line 9 (only for valid pixels)

We mention several specific implementation issues below, which are: (1) how to prevent changes to points that go out of view; (2) how to decide whether a pixel is an outlier; (3) how to integrate measurements from multiple runs into a single coherent map; and (4) when to choose a new reference image.

First, the valid matrix prevents changes to the state variables for points that go out of view. A pixel in $\mathcal{I}_1$ is set to invalid for the current image when the homography $\mathsf{H_i}$ maps the corresponding pixel in $\mathcal{I}_i^r$ to a pixel outside of $\mathcal{I}_i$. The matrix numValid counts the number of images in which each pixel in $\mathcal{I}_1$ is valid.

Second, we combine several heuristics to choose when a pixel is an outlier:

1. Flows may not extend past the edge of the image.
2. All generated points must be valid in at least 5 images since the last reference change (encoded in numValid); they must have an average absolute residual (residual/numValid) no larger than a fixed constant; and they must be no closer than 2 pixels from the edge of the image.
3. The remaining points are then iteratively filtered for outliers, on each iteration rejecting points that are too many standard deviations above the mean in their world x or y coordinates. This is followed by iteratively discarding all points with (x, y) coordinates in any 25th (5x5 equal-sized blocks) of the portion of the x, y plane spanned by the data that contains less than 0.1% of the points. Finally, points are again iteratively rejected based on being too many standard deviations above the mean in their world z coordinates, or in $\mathrm{Var}\,(z)$.

Third, using the recovered depths each pixel in the reference image is back-projected into world coordinates (see Appendix), resulting in a space point which is integrated into a modular fixed-grid elevation and appearance map. By modular we mean that we store fixed resolution, e.g. 16×16, cells, and fill in cells only when data is available in that location. All values contained in the same grid square (not cell) are optimally combined using their estimated variances.

Fourth, and finally, a new reference image $\mathcal{I}_1$ is set when: (*i*) the percentage of pixels that are valid after aligning the next image is below a given threshold (we use 50%); or (*ii*) after a set maximum number of images since the last reference change to reduce mapping latency (we use 20). Space points are generated upon every change in reference image. To reduce the movement due to parallax, we use a horizontal reference plane (in world coordinates) at the height of the mean terrain elevation in the reference image, an approximation of which is available from a separate motion filter.

8 Results

We tested this algorithm on both synthetic and real image sequences. For the synthetic tests, we developed a closed-loop simulation system for a simple aerial vehicle that includes a simple vehicle dynamics simulator, a trajectory planner for executing a simple search pattern, and a synthetic view generator that renders images from DTED elevation data and Terraserver satellite imagery. Using this framework enables us to compare our reconstructed terrain to the ground truth and perform error analysis.

First we present a comparison of stereo, wide baseline stereo, and RMFPP on synthetic data. Figure 3 shows error results for a single pair of stereo images, or a single reference image and consecutive frames until the next reference change, of a 50 meter/sec flight over mountainous terrain using a camera with a focal length of 251 meters and 240x320 images, alongside ideal standard deviation curves. The stereo images were chosen to be the first two images of the RMFPP sequence and the wide baseline stereo images were chosen to be the first and last images of the RMFPP sequence. Note that the x-axis is absolute height and that the terrain is centered around an elevation of 950 meters, so the relative heights are in the range 550-2350 meters. The ideal curves are only valid up to a global scale based on the properties of the images, but it is clear that the algorithms perform as the expected functions of relative height.

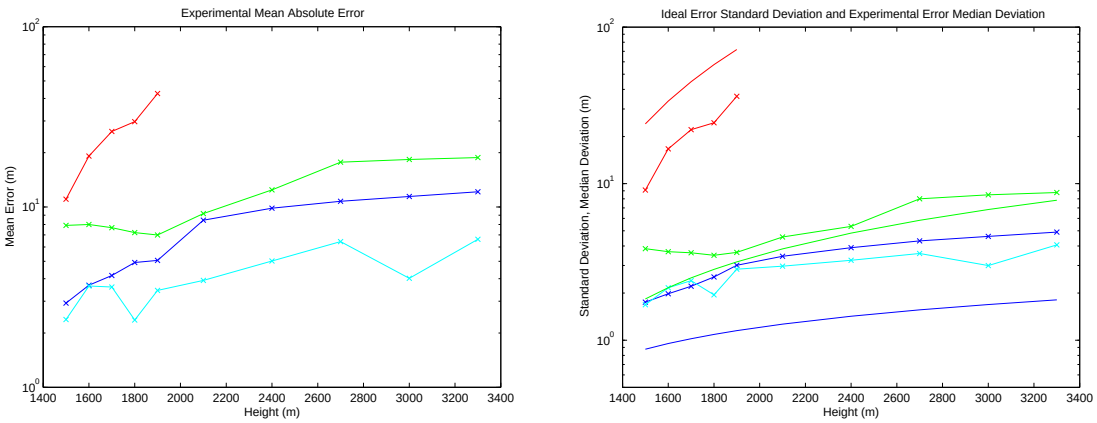

Figure 3: Ideal (solid) and experimental (solid with crosses) error analysis for stereo (red), wide baseline stereo (green), and RMFPP (blue). Left: Experimental mean absolute error vs height. Right: Ideal error standard deviation (up to scale) and experimental error median deviation vs height.

In the experimental error distributions we only count points about which the algorithms are certain (we do not penalize for holes or for pixels that are removed by filtering prior to map integration), although we note that RMFPP produced a result with few holes while the wide baseline stereo only produced results in about half of the image due to the reduced overlap of its views.

Figure 4 shows the result of RMFPP on images rendered from an outwardly spiraling synthetic flight at 100 meters/sec and 1500 meters absolute elevation (the terrain is in the range 775-1025 meters) using the same camera parameters as in the previous experiment. The reconstructed appearance is overlayed on the reconstructed elevation. Note that the black regions in the center of the reconstruction are invalid regions that are not explored by the trajectory. We also include a histogram of elevation errors and a plot of the correlation between elevation errors at neighboring pixels. The correlation between errors at neighboring pixels shows that the method captures the relative height of the terrain even when its absolute estimate has error.

Figure 5 shows the result of RMFPP on images captured from a real autonomous flight at FIXME meters/sec and FIXME meters absolute elevation (the terrain is in the range FIXME meters) using the a camera with a focal length of FIXME capturing 320x240 images. FIXME: say something else.

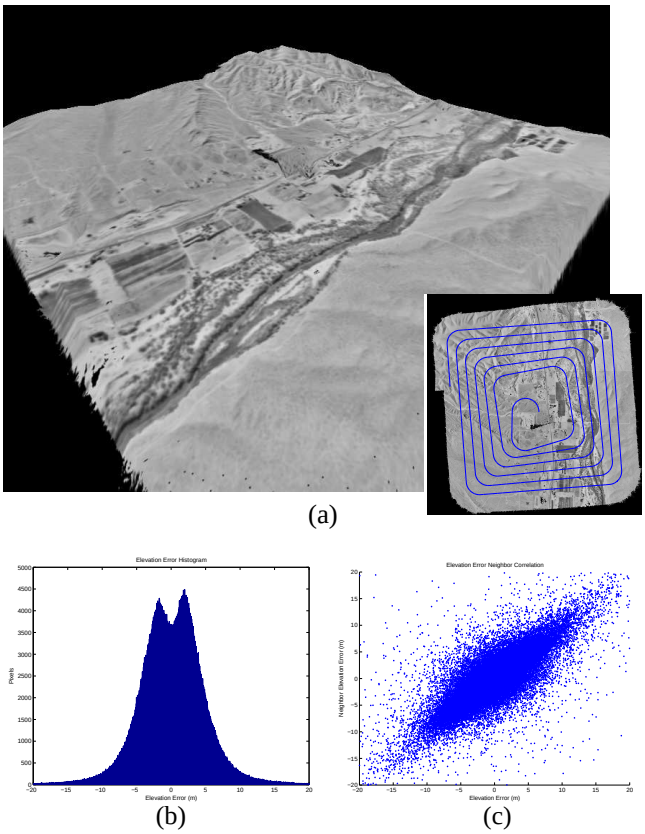

Figure 4: Synthetic images experiment: (a) The reconstructed appearance draped over the reconstructed elevation. The inset shows the simulated vehicle trajectory. (b) Histogram of height errors in meters. (c) Correlation between height errors at adjacent pixels.

9 Conclusion

This paper has introduced the Recursive Multi-Frame Planar Parallax algorithm, which is a direct, dense, accurate, and recursive method for recovering shape from a monocular sequence of images with known motions. These capabalities were desirable from the point of view of an aerial

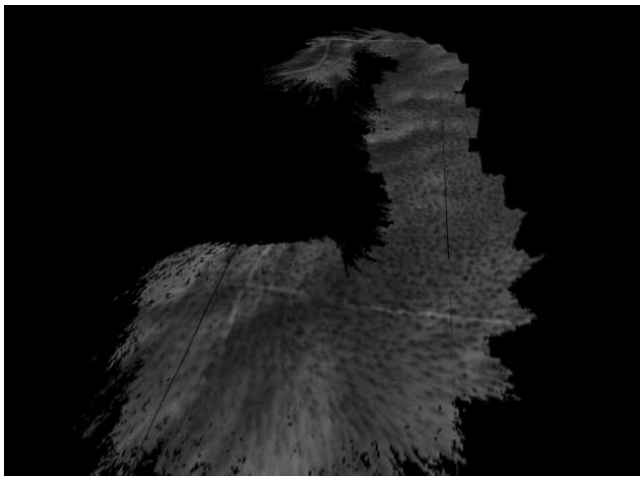

Figure 5: Real images experiment: The reconstructed appearance draped over the reconstructed elevation.

vehicle requiring on-board real-time terrain analysis. We have demonstrated the algorithm on both synthetic and real image sequences, and have shown that its performance is close to that of the batch method, and to the theoretical performance of linear variance derived for pure translation.

References

[1] E. Sokolowsky, H. Mitchell, and J. de La Beaujardiere, "Nasa's scientific visualization studio image server," in *Proc. IEEE Visualization 2005*, p. 103, 2005.

[2] S. K. Jenson. and J. O. Domingue, "Extracting topographic structure from digital elevation data for geographic information systems analysis," *Photogrammetric Engineering and Remote Sensing*, vol. 54, no. 11, pp. 1593 – 1600, 1988.

[3] B. Sofman, J. Bagnell, A. Stentz, and N. Vandapel, "Terrain classification from aerial data to support ground vehicle navigation," Tech. Rep. CMU-RI-TR-05-39, Robotics Institute, Carnegie Mellon University, 2006.

[4] J. McMichael and M. Francis, "Micro air vehicles–toward a new dimension in flight," tech. rep., DARPA, 1997.

[5] O. Faugeras, *Three-Dimensional Computer Vision*. The MIT Press, 1993.

[6] L. Matthies and S. A. Shafer, "Error modeling in stereo navigation," *J. of Robotics and Automation*, vol. RA-3, June 1987.

[7] H. S. Sawhney, "3d geometry from planar parallax," in *Proceedings of Computer Vision and Pattern Recognition*, June 1994.

[8] M. Irani, P. Anandan, and M. Cohen, "Direct recovery of planar-parallax from multiple frames," *Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, November 2002.

[9] M. Irani and P. Anandan, "About direct methods," in *Proc. International Workshop on Vision Algorithms*, September 1999.

[10] P. H. S. Torr and A. Zisserman, "Feature based methods for structure and motion estimation," in *Proc. International Workshop on Vision Algorithms*, September 1999.

[11] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, "Bundle adjustment – a modern synthesis," in *Proc. International Workshop on Vision Algorithms*, September 1999.

[12] M. Okutomi and T. Kanade, "A multiple-baseline stereo method," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 15, pp. 353–363, April 1993.

[13] G. P. Stein and A. Shashua, "Direct estimation of motion and extended scene structure from a moving stereo rig," Tech. Rep. AIM-1621, Massachussetts Institute of Technology, 1997.

[14] Y. Xiong and L. Matthies, "Error analysis of a real-time stereo system," in *Proceedings of Computer Vision and Pattern Recognition*, June 1997.

[15] M. Zucchelli and H. I. Christensen, "Recursive flow based structure from parallax with automatic rescaling," in *British Machine Vision Conference*, September 2001.

[16] L. Matthies, T. Kanade, and R. Szeliski, "Kalman filter-based algorithms for estimating depth from image sequences," *International Journal of Computer Vision*, vol. 3, September 1989.

Appendix: World Coordinates

Combining the equation for height above the reference plane in the frame of the reference camera and the definition of γ, the point $\boldsymbol{p} = (x, y)$ in the reference image has z coordinate in the frame of the reference camera

$$z = -\frac{d_1}{\boldsymbol{N}^T \boldsymbol{X'} - \gamma}, \tag{13}$$

where $\boldsymbol{X'} = \mathsf{K}^{-1}\pi^*(\boldsymbol{p})$. Back-projecting into 3-dimensional space, the point $\boldsymbol{X}$ and its covariance in the frame of the reference camera $\mathrm{Cov}(\boldsymbol{X})$ are given by:

$$\begin{aligned} \boldsymbol{X} &= -\frac{d_1}{\boldsymbol{N}^T \boldsymbol{X'} - \gamma}\boldsymbol{X'} \\ \mathrm{Cov}(\boldsymbol{X}) &= \mathsf{J}\,\mathrm{Var}(\gamma)\,\mathsf{J}^T \\ \text{where} \quad \mathrm{Var}(\gamma) &= \frac{1}{\Sigma A} \\ \text{and} \quad \mathsf{J} &= \frac{d_1}{\left(\boldsymbol{N}^T \boldsymbol{X'} - \gamma\right)^2}\boldsymbol{X'}. \end{aligned} \tag{14}$$

To construct an elevation map over multiple reference frames, the point and its covariance can be transformed into the world coordinate system using the known location and orientation of the reference camera.