Proceedings of ICRA 2003
May 2003, Taipei, Taiwan, to appear

Direct Plane Tracking in Stereo Images for Mobile Navigation

Jason Corso, Darius Burschka, and Gregory Hager
Computational Interaction and Robotics Laboratory
The Johns Hopkins University
Baltimore, MD 21218
{jcorso|burschka|hager}@cs.jhu.edu

***Abstract*— We present a novel plane tracking algorithm based on the direct update of surface parameters from two stereo images. The plane tracking algorithm is posed as an optimization problem, and maintains an iteratively re-weighted least squares approximation of the plane's orientation using direct pixel measurements. To facilitate autonomous operation, we include an algorithm for robust detection of significant planes in the environment. The algorithms have been implemented in a robot navigation system.**

I. Introduction

Inferring properties of the real-world through one or many images is fundamental to the field of computer vision. Often it is beneficial to have knowledge of a significant surface or set of surfaces during the inference process. An example of such a surface is a plane.

Many methods have been proposed to solve the problem of planar-surface tracking for monocular [9], [10] and binocular [13], [14], calibrated and uncalibrated cameras (similar to using sequences of images [2], [3], [5]). A common solution involves a disparity map computation. A disparity map is a matrix of correspondence offsets between two images [16]. The disparity map calculation employs expensive neighborhood correlation routines that often yield sparse maps for typical scenes. However, the method makes no assumptions about the environment and has been widely used for the case of general stereo vision.

In contrast to the general solution discussed above, our method exploits a property that planar surfaces exhibit when viewed through a non-verged stereo camera. The disparity is a linear function with coefficients derived from the plane parameters. We use a direct method to perform the tracking. Direct methods use quantities that are calculated directly from image values as opposed to feature-based methods discussed earlier [8], [14]. Our method has descended from earlier image registration and enhancement techniques [11], [12] and from visual tracking [6]. Similarly, it poses the tracking problem as one of objective function minimization. Doing so incorporates a *vote* from many pixels in the image and computes a least-squares estimate of the global motion parameters in question to sub-pixel levels.

The planar tracking algorithm is applied to robot navigation. Multiple tasks on a mobile robot require knowledge about the incremental changes in position during the operation. The task of self-localization in an environment can be divided into two major areas: global localization and local position tracking. While the first deals with the absolute localization in a predefined coordinate system and can be used for initial pose estimation [4], the second computes the relative changes in the position due to the movement of the mobile system. For most purposes (e.g. map generation) the tracking of the local position is enough to guarantee a correct fusion of consecutive sensor readings.

II. Algorithm Description

The key component of our work is the plane tracking algorithm that operates directly in the image domain of the acquired images [1]. In the following discussion, let (x, y, z) be a point in world coordinates and (u, v) be a point in pixel coordinates. In Section II-B, the plane tracking algorithm is discussed followed by the approach used to seed the tracking algorithm (II-C). A brief discussion on localization (II-D) is then presented.

A. Planar Disparities

To efficiently track a plane, we can make use of a property that planar surfaces exhibit when viewed from non-verged stereo cameras. Namely, a plane becomes a linear function that maps pixels in one image to the corresponding pixels on the plane in the other image. Indoor environments have many surfaces that can be approximated with planes $\mathcal{E}$.

$$\mathcal{E} : ax + by + cz = d \tag{1}$$

In a stereo system with non-verged, unit focal length (f=1) cameras, the image planes are coplanar. In this case, the disparity value $D(u, v)$ of a point (u,v) in the image can be estimated from its depth z with

$$D(u, v) = \frac{B}{z}, \tag{2}$$

with B describing the distance between the cameras of the stereo system [16].

We estimate the disparity $D(u, v)$ of the plane $\mathcal{E}$ at an image point (u, v) using the unit focal length camera (f=1) projection as

$$\begin{aligned} \forall z \neq 0: \quad a\frac{x}{z} + b\frac{y}{z} + c &= \frac{d}{z} \\ au + bv + c &= k \cdot D(u,v) \\ \text{with } u = \frac{x}{z},\ v = \frac{y}{z}, \quad k &= \frac{d}{B} \end{aligned} \tag{3}$$

The vector $\mathbf{n} = (a\ b\ c)^T$ is normal to the plane $\mathcal{E}$ and describes the orientation of the plane relative to the camera.

Equation (3) can be written in the form

$$\begin{aligned} D(u,v) &= \begin{pmatrix} \rho_1 \\ \rho_2 \\ \rho_3 \end{pmatrix} \cdot \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \mathbf{n}^* \cdot \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \\ \text{with} \quad & \rho_1 = \frac{a}{k},\ \rho_2 = \frac{b}{k},\ \rho_3 = \frac{c}{k} \end{aligned} \tag{4}$$

This form uses modified parameters $\{\rho_1, \rho_2, \rho_3\}$ of the plane E relating the image data (u, v) to $D(u, v)$.

B. Plane Tracking

From the observation made in Section II-A, we see that tracking the parameters $\mathbf{p} = \{\rho_1, \rho_2, \rho_3\}$ of the linear map (4) is equivalent to tracking the planar surface. Thus, assuming inter-frame motion is relatively small and both brightness and contrast shifts can be removed, we pose this problem as one of optimization.

1) Parameter Update: Consider a rectified pair of stereo images: L and R. Based on (4), we relate the images with the following formula. Let $D(u, v) = \rho_1 u + \rho_2 v + \rho_3$.

$$L(u,v) = R(u - D(u,v), v) \tag{5}$$

The plane parameters for the current pair are estimated through the minimization of the following least-squares objective function. To enforce the *brightness constancy constraint* [7], we zero-mean the images: given an image I, $\overline{I} = I - \sum_u \sum_v I(u,v)$.

$$E(\mathbf{p}) = \sum (\overline{L}(u,v) - \overline{R}(u - D(u,v), v))^2 \tag{6}$$

Let $\delta\mathbf{p}$ represent the set of offsets, i.e. at iteration i, $\mathbf{p}_{i+1} = \mathbf{p}_i + \delta\mathbf{p}$. Assuming a small magnitude for $\delta\mathbf{p}$ we can solve the minimization by linearizing the expression through a Taylor expansion about $\mathbf{p}$.

$$\begin{aligned} E(\delta\mathbf{p}) \approx \sum (&\overline{L}(u,v) - \overline{R}(u - D(u,v), v) \\ &+ uI_x\delta\rho_1 + vI_x\delta\rho_2 + I_x\delta\rho_3)^2 \end{aligned} \tag{7}$$

Here, I_x refers to the spatial gradient of the right image. We neglect the higher order terms of the Taylor series. We solve the system with the Singular-Value Decomposition (SVD) [15]. It is first convenient to define the error term: $e(u,v) = \overline{R}(u - D(u,v), v) - \overline{L}(u,v)$.

$$\begin{bmatrix} e(u_1,v_1) \\ e(u_1,v_2) \\ \cdots\cdots\cdots \\ e(u_m,v_n) \end{bmatrix} = \begin{bmatrix} u_1 I_{x_{u_1}} & v_1 I_{x_{u_1}} & I_{x_{u_1}} \\ u_1 I_{x_{u_2}} & v_2 I_{x_{u_2}} & I_{x_{u_2}} \\ \cdots\cdots & \cdots\cdots & \cdots\cdots \\ u_m I_{x_{u_m}} & v_n I_{x_{u_m}} & I_{x_{u_m}} \end{bmatrix} \begin{bmatrix} \delta\rho_1 \\ \delta\rho_2 \\ \delta\rho_3 \end{bmatrix} \tag{8}$$

2) The Weighting Matrix: Thus far, we have shown how to optimize the parameters of a plane in a static scene. To extend the approach to a dynamic scene, we incorporate a mask into the framework. The mask is a binary weighting matrix with an entry per-pixel denoting the pixel's inclusion or exclusion from the current tracked plane. Such a mask removes inaccurate and poor pixel matches from the SVD solution, which decreases its processing demand and increases its stability. We note that no explicit plane-boundary is maintained. Equation (7) is extended to (9) incorporating such a mask; $W(u,v)$ corresponds to the value of the mask at (u, v).

$$\begin{aligned} E(\delta\mathbf{p}) \approx \sum \delta_{W(u,v)=1} [(&\overline{L}(u,v) - \overline{R}(u - D(u,v), v) \\ &+ uI_x\delta\rho_1 + vI_x\delta\rho_1 + I_x\delta\rho_2)^2] \end{aligned} \tag{9}$$

In each frame, we fully recompute the mask matrix based on the current parameters (12). We employ two metrics in this computation: a normalized cross-correlation ($\odot$) and horizontal variance.

$$\eta_{u,v} = \overline{L}(u,v) \odot \overline{R}(u - D(u,v), v) \tag{10}$$

Since (9) is only sensitive to pixels that demonstrate horizontal gradients, we also mask pixels with low horizontal variance by sampling the correlation response along the epipolar line.

$$\begin{aligned} \alpha_{u,v} &= \frac{\overline{L}(u,v) \odot \overline{R}(u - D(u,v) + \delta, v)}{\eta_{u,v}} \\ \beta_{u,v} &= \frac{\overline{L}(u,v) \odot \overline{R}(u - D(u,v) - \delta, v)}{\eta_{u,v}} \end{aligned} \tag{11}$$

$$W(u,v) = (\eta_{u,v} > \tau) \wedge (\alpha_{u,v} > \epsilon) \wedge (\beta_{u,v} > \epsilon) \tag{12}$$

In (12), $0 < \tau < 1$ and $\epsilon > 1$. In (11,12), the selection of δ,τ, and ϵ is dependent on the imaging properties of the system and the scene. To increase the robustness of the mask generated in this manner, we also perform a morphological dilation followed by an erosion [6]. This has the effect of removing noisy correlation responses and joining contiguous regions.

C. Identification of Significant Planes

The tracking algorithm discussed in the previous section requires a set of initial guesses for the plane parameters. These estimates are generated from disparity images. It is not prohibitive to include disparity-based seed generation because this module is seldom executed.

Equation (3) shows a linear dependency of the disparity values $D(u,v)$ on the image coordinates (u,v). Therefore, the plane boundaries can be detected as discontinuities in the disparity values $\mathcal{D}^t$ along the u and v axes. We use the sum of the absolute values of the horizontal $\delta_x\mathcal{D}^t$ and vertical $\delta_y\mathcal{D}^t$ gradient images to calculate a contour image $\mathcal{G}^t$ that approximates the gradient magnitude in the disparity image to

$$|\nabla\mathcal{D}^t| \approx \mathcal{G}^t = |\delta_x\mathcal{D}^t| + |\delta_y\mathcal{D}^t|. \tag{13}$$

Equation (13) gives a good approximation of the gradient magnitude $|\nabla\mathcal{D}^t|$ and has the advantage of easy implementation with accelerated image processing routines. The result of this operation is shown in Figure 1.

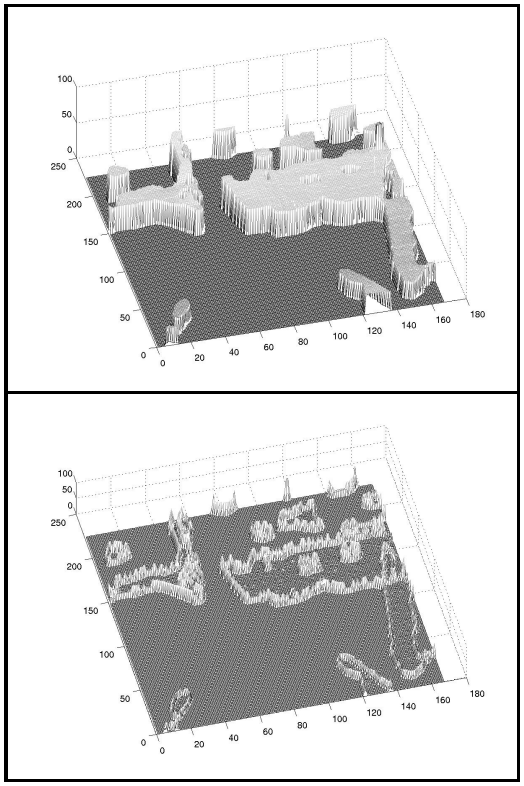

Fig. 1. Contour detection similar to conventional image processing: (top) original disparity image, (bottom) resulting contour image $\mathcal{G}^t$.

The goal of these routines is to find regions in the image representing large uniform planes. Therefore, the resulting image $\mathcal{G}^t$ is thresholded and all values above a disparity value of 4 pixels are considered to be on a boundary and set to a significant negative value n_{edge} proportional to the area $\sigma_x \times \sigma_y$ covered by the kernel σ. All gradient values in $\mathcal{G}^t$ resulting from invalid reconstructions in $\mathcal{D}^t$ are set to the same negative value n_{edge}.

The entire image is convolved with a signum kernel σ counting valid gradient values in $\mathcal{G}^t$.

$$\Sigma^t = \sigma * \mathcal{G}^t(u,v) = \sum_{y=v-\frac{\sigma_y}{2}}^{v+\frac{\sigma_y}{2}} \sum_{x=u-\frac{\sigma_x}{2}}^{u+\frac{\sigma_x}{2}} \operatorname{sgn}(\mathcal{G}^t(x,y)+1) \tag{14}$$

During the computation of Σ^t the position (u_Σ, v_Σ) of the maximum value is estimated. It defines a seed for a new plane. The area $\sigma_x \times \sigma_y$ defines the preferred minimum plane size in the image.

1) Estimation of the Plane Parameters: We estimate the plane parameters in 3D space. A set of valid 3D points, $\mathcal{S}_v$, is reconstructed from the disparity image. In the ideal case the set should represent a compact disparity region surrounding the estimated seed (u_Σ, v_Σ) from section II-C. To simplify the search for the points we approximate the circle with a simple cross extending from this point in all directions to the boundary of $\mathcal{G}^t$ or until a negative n_{edge} entry is found. The N resulting 3D-points $P_i = (x_i, y_i, z_i)^T$ are stored in $\mathcal{S}_v$.

We fit an optimal plane through this dataset using the eigenvectors of the covariance matrix:

$$C = \begin{pmatrix} \sigma_{xx}^2 & \sigma_{xy}^2 & \sigma_{xz}^2 \\ \sigma_{yx}^2 & \sigma_{yy}^2 & \sigma_{yz}^2 \\ \sigma_{zx}^2 & \sigma_{zy}^2 & \sigma_{zz}^2 \end{pmatrix} \tag{15}$$

$$\text{with} \quad m = \begin{pmatrix} m_x \\ m_y \\ m_z \end{pmatrix} = \tfrac{1}{N}\textstyle\sum_i P_i \qquad \wedge$$

$$\sigma_{ab}^2 = \tfrac{(a-m_a)\cdot(b-m_b)}{N}$$

The cross-product of the eigenvectors $\{X_1, X_2\}$ associated with the two biggest eigenvalues $\{\lambda_1, \lambda_2\}$ defines the norm vector $n = (a,b,c)^T$ of the reconstructed plane (1). The inner product between this norm vector and the mean, m, on the plane defines the distance d to the plane.

$$n = \frac{X_1 \times X_2}{|X_1 \times X_2|} \quad \text{and} \quad d = n \cdot m \tag{16}$$

The ratio of the third eigenvalue λ_3 to the other eigenvalues describes how well the reconstructed plane represents the cloud of the used 3D points. A smaller value represents a better approximation.

D. Localization

The relative localization between consecutive camera acquisitions is based on localization relative to significant planes in the field of view of the camera. Each plane allows the estimation of three out of the six possible parameters of the pose. A set of two non-coplanar planes $\mathcal{L}^t = \{\mathcal{E}_i, \mathcal{E}_j \mid \forall_{i,j} \, |n_i \cdot n_j| > 0\}$ at time step t allows the estimation of the 2D position in the ground plane of the local area and all rotation angles of the robot. Therefore, relative localization is possible when at least two planes are tracked between frames, $||\mathcal{L}^{t+1} \cap \mathcal{L}^t|| \geq 2$.

1) Estimation of Multiple Planes: For complete localization, we include the estimation of multiple planes in our approach. We extend the significant-plane extraction by using the parameters of the known planes to remove their representation from the disparity image $\mathcal{D}^t$ used in the processing step described in Section II-C.

$$\begin{aligned} \Delta d &= \mathcal{D}^t(u,v) - (\rho_1 u + \rho_2 v + \rho_3) \\ \mathcal{D}^t(u,v) &:= \begin{cases} 0, & \Delta d < \epsilon \cdot \mathcal{D}^t(u,v) \\ \mathcal{D}^t(u,v), & \text{else} \end{cases} \end{aligned} \tag{17}$$

The disparity value $\mathcal{D}^t(u,v)$ is deleted from the disparity image if it matches the expectation calculated from Equation (3) within an uncertainty band ϵ around the reconstructed value. During processing, whenever $||\mathcal{L}^{t+1} \cap \mathcal{L}^t|| < 2$, a significant plane search is executed to re-seed the plane tracking algorithm.

III. Results

Our algorithm has been implemented on a Pentium III 700MHz PC running Linux with an IEEE 1394 stereo camera head. The stereo vision system provides a rectified stream of images at a maximum of 26[Hz]. Our implementation operates in the range of 20-26[Hz] for plane updates. We employ both the XVision2 and Intel Integrated Performance Primitives libraries for video and image processing. For the experiments discussed in this section, we are using a stereo head with 5.18mm lenses, a 92mm baseline, and square pixels 0.12mm wide. The plane being observed, unless otherwise specified, is roughly orthogonal to the viewing axis and at a depth of one-meter.

A. Quality of the Plane Tracking

We analyze the algorithm for maximum guaranteed convergence. Equation (7) was derived under the assumption that inter-frame parameter offsets ($\delta\mathbf{p}$) would be small in magnitude. In discrete terms, we are restricted to changes of ± 1 disparity. To facilitate analysis, we assume an infinite plane. Under lateral translation, the depth and the disparity are constant and the algorithm is guaranteed to converge.

However, under orthogonal translation, the convergence radius is dependent on the current depth; the closer to the camera, the smaller the convergence radius. This is due to the inverse relationship between disparity and depth ($D = \frac{Bf}{z}$). Let ζ be the change in depth for guaranteed convergence.

$$\frac{Bf}{D \pm 1} - z = \zeta \tag{18}$$

Rotations in the plane are also dependent on the current depth. Assuming a uni-axis rotation about the center of projection in the unit focal length camera, the maximum angle of rotation, θ, follows. Figure 2 depicts this formula in a 2D example. We relate the rotation angle to the depth of a projected point, P, in the image Δ pixels away from the optical center.

$$\theta = \arctan\left(\frac{B\Delta}{D \pm 1} - \frac{z}{\Delta}\right) \tag{19}$$

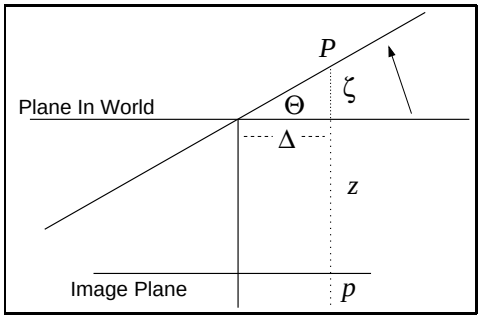

Fig. 2. Depiction of the formula for convergence radius under uni-axis rotation.

In the following sections, experiment results are presented and discussed.

1) Convergence Radius: For a controlled environment with a stationary plane and robot, we calculated an initial guess for the plane parameters and then varied this guess to test the robustness of the tracking algorithm.

In Figure 3, we show the time to convergence when we shift the seed's depth closer to the camera at varying levels. The convergence speed is directly proportional to the magnitude of the introduced error. We note that the convergence speed is only about 5 frames for an error of 10%. In Figure 4, we show the convergence speed while altering the plane's normal about a single axis. We see similar convergence rates to those observed while altering seed depth.

2) Accuracy of Parameter Estimation: Assuming a suitably textured scene, the algorithm estimates a plane's parameters with sub-pixel accuracy. However, this estimation accuracy varies with the depth of the plane being tracked. Because the depth-per-disparity increases as the distance to the plane increases, the estimation accuracy

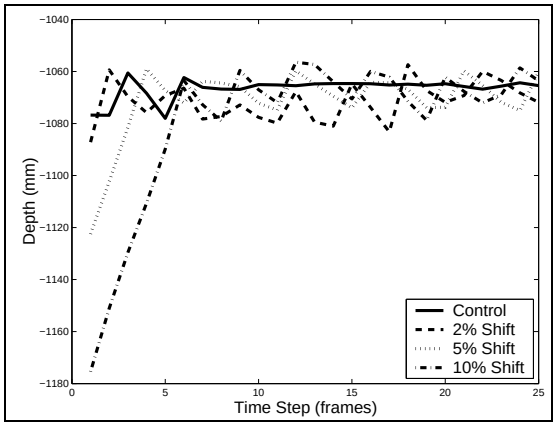

Fig. 3. Graph for convergence while introducing error into the seed's depth by 2, 5 and 10 percent toward the camera.

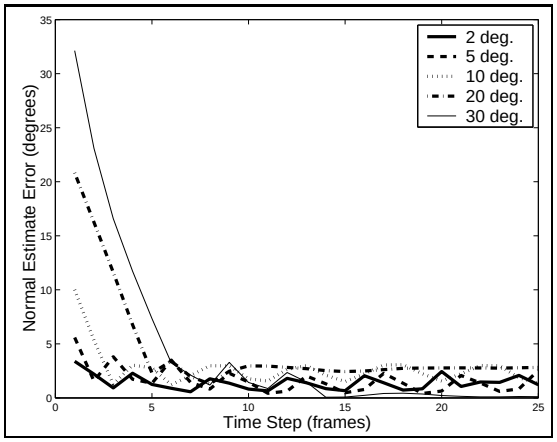

Fig. 4. Convergence rates for error in seed normal.

degrades with depth. Table I shows the statistics for the plane.

For a non-stationary scene, we show the accuracy of our system against the robots internal odometry. Figure 5 shows the robot performing oscillatory rotations in front of a plane (700 mm distance), and Figure 6 show the robot orthogonally translating in a back-and-forth motion. We see that the algorithm performs extremely well for the rotational motion. The estimated orientation lags minimally behind the odometric values; the length of the lag is proportional to the convergence speed. During the translational motion, the algorithm oscillates about the control value during motion. As expected, it is apparent from the figure that the tracking does perform better when the robot is closer to the plane.

TABLE I

PARAMETER ESTIMATION ACCURACY FOR A PLANE AT A DISTANCE OF ONE METER.

	Z Mean	Z Std Dev	Normal Error Std Dev
1	1064.8mm	2.2359mm	0.2947°
2	1065.3mm	1.7368mm	0.2673°
3	1065.2mm	1.5958mm	0.2258°

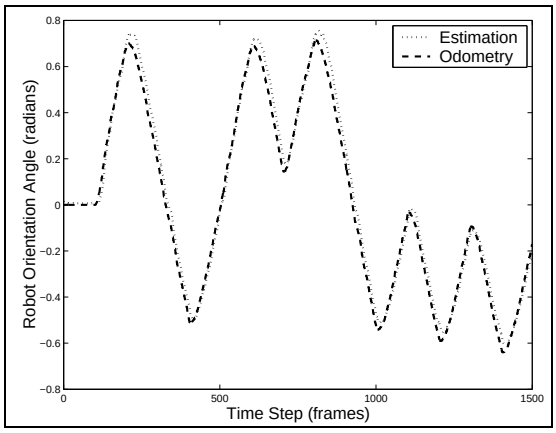

Fig. 5. Accuracy of robot orientation.

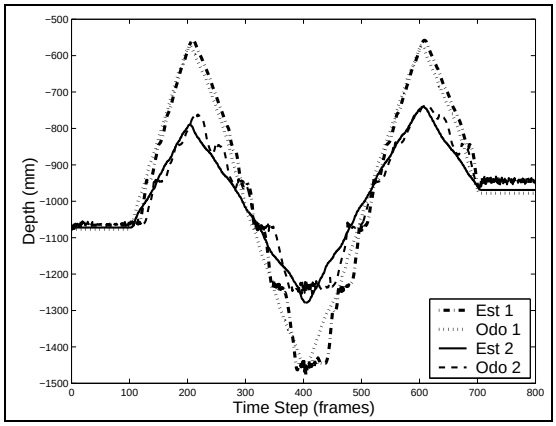

Fig. 6. Accuracy of robot depth.

IV. CONCLUSIONS AND FUTURE WORK

In this paper, we presented a novel plane-tracking algorithm that maintained an iteratively re-weighted least-squares approximation of the plane parameters with sub-pixel accuracy. The method has been applied to the task of mobile navigation. Our implementation of the approach typically executes at or near frame rate and maintains accurate parameter estimate.[1] We presented a set of experiments that analyzed the algorithm's convergence radius and the accuracy of the parameter estimation: these results extend to multiple planes because each plane is tracked independently.

[1]The implementation executes two iterations of the optimization routine every frame. This operating rate is in contrast to a disparity calculation based system which typically runs at half frame-rate on the same system.

Accurate estimation is highly dependent on the optics of the imaging system and the environment. These dependencies are fundamental to the set of approaches employed in stereo vision, however, not specifically our approach. Currently the parameters of our algorithm are chosen on an environment basis. For instance, the δ in (11) is dependent on the texture frequency. However, the imaged texture frequency may vary from one plane to another and will vary with large depth changes. Such operation is considered passive in nature. The exploration of algorithms that take more active approaches (i.e. projecting a suitable pattern depending on the current environment) or more intelligent approaches (i.e. altering operating parameters and thresholds autonomously by analyzing important scene qualities) are needed for further advancement of the field.

V. Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No. 0112882. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

This work has also been supported by the MARS project.

VI. References

[1] Jason Corso and Gregory D. Hager. Planar surface tracking using direct stereo. Technical report, The Johns Hopkins University, 2002. CIRL Lab Technical Report.

[2] F. Dellaert, S. Seitz, C. Thorpe, and S. Thrun. Structure from motion without correspondence. Technical Report CMU-RI-TR-99-44, Carnegie Mellon University, 1999.

[3] F. Dellaert, C. Thorpe, and S. Thrun. Super-resolved texture tracking of planar surface patches. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robotic Systems*, 1998.

[4] Frank Dellaert, Dieter Fox, Wolfram Burgard, and Sebastian Thurn. Monte carlo localization for mobile robots. In *Proc. IEEE Int. Conf. on Robotics and Automation*, 1999.

[5] V. Ferrari, T. Tuytelaars, and L. Van Gool. Real-time affine region tracking and coplanar grouping. In *IEEE Computer Soc. Conf. on Computer Vision and Pattern Recognition*, 2001.

[6] G. Hager and P. Belhumeur. Efficient region tracking with parametric models of geometry and illumination. *IEEE Transactions of Pattern Analysis and Machine Intelligence*, 20(10):1125–1139, 1998.

[7] B. Horn. *Robot Vision*. The MIT Press, 1986.

[8] M. Irani and P. Anandan. About direct methods. In *Vision Algorithms: Theory and Practice (International Workshop on Vision Algorithms)*, 1999.

[9] K. Kanatani. Detection of surface orientation and motion from texture by stereological technique. *Artificial Intelligence*, 24:213–237, 1984.

[10] K. Kanatani. Tracing planar surface motion from a projection without knowing the correspondence. *Computer Vision, Graphics, And Image Processing*, 29:1–12, 1985.

[11] D. Keren, S. Peleg, and R. Brada. Image sequence enhancement using sub-pixel displacements. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 1988.

[12] B. Lucas and T. Kanade. An iterative image registratoin technique with an application to stereo vision. In *Proceedings DARPA Image Understanding Workshop*, 1981.

[13] S. Pei and L. Liou. Tracking a planar patch in three-dimensional space by affine transformation in monocular and binocular vision. *Pattern Recognition*, 26(1):23–31, 1993.

[14] G. Stein and A. Shashua. Direct estimation of motion and extended scene structure from a moving stereo rig. *IEEE Conference on Computer Vision and Pattern Recognition*, 1998.

[15] G. Strang. *Linear Algebra and Its Applications*. Saunders HBJ, 1988.

[16] Emanuele Trucco and Alessandro Verri. *Introductory Techniques for 3-D Computer Vision*. Prentice Hall, 1998.