Ego-Motion Estimation and 3D Model Refinement in Scenes with Varying Illumination

Amit K Agrawal and Rama Chellappa
Center for Automation Research
University of Maryland
College Park, MD 20770 USA
{aagrawal,rama}@cfar.umd.edu

Abstract

We present an iterative algorithm for robustly estimating the ego-motion and refining and updating a coarse depth map using surface parallax and a generalized dynamic image (GDI) model. Given a coarse depth map acquired by a range-finder or extracted from a Digital Elevation Map (DEM), we first estimate the ego-motion by combining a global ego-motion constraint and a local GDI model. Using the estimated camera motion and the available depth estimate, motion of the 3D points is compensated. We utilize the fact that the resulting surface parallax field is an epipolar field and constrain its direction using the previous motion estimates. We then estimate the magnitude of the parallax field and the GDI model parameters locally and use them to refine the depth map estimates. We use a tensor based approach to formulate the depth refinement procedure as an eigen-value problem and obtain confidence measures for determining the accuracy of the estimated depth values. These confidence measures are used to remove regions with potentially incorrect depth estimates for robustly estimating ego-motion in the next iteration. Experimental results using both synthetic and real data are presented. Comparisons with results obtained using a brightness constancy (BC) model show that the proposed algorithm works significantly better when time-varying illumination changes are present in the scene.

1. Introduction

3D scene reconstruction and ego-motion estimation has been an active area of research over the past few decades. Dynamic scene analysis requires estimation of the relative motion between the camera, scene and the 3D scene structure in the form of a depth map. Motion estimation of a camera moving in an environment is useful for tasks such as navigation, obstacle-detection etc., and recovering the scene structure helps in enhanced visualization and building 3D models of the scene. Several researches have worked on the problem of ego-motion estimation and depth recovery using intensity images. Feature based methods [9][29][28][27][25][3][26][13][22] use features or tokens to get depth information and motion. Flow based methods [15][1] assume that optical flow is available. Direct methods [2][8][6][19][14][4][12][25][24] do not require intermediate steps such as feature extraction or flow computation and work directly with spatio-temporal image gradients. These techniques minimize the deviation from the brightness change model with respect to structure and motion parameters.

The algorithm presented here comes under the category of direct methods. The commonly used brightness constancy (BC) model to compute optical flow or match correspondences, or that used in gradient-based direct methods, is hardly valid in scenes where inter-frame brightness variations are not negligible and time-varying illumination changes are present. Several researchers have worked on overcoming the limitations of the BC model for optical flow computation and structure recovery. Black et. al. [5] build a robust statistical framework in which brightness variations are represented as probabilistic mixtures of different causes. Negahdaripour [18] models the inter-frame brightness variations as a multiplicative and additive field prior to computing the optical flow. In [7], the authors advocated using specific physical models of brightness variations for optical flow computation. In [30], Zhang et. al. proposed a unifying algorithm for estimating optical flow, shape, motion and albedo using a generalized brightness change model. However, their analysis assumes orthographic projection which is a serious limitation. Negahdaripour [20] also proposed a direct solution for estimating depths and motion in scenes with time-varying illumination. They expressed the dynamic image model in terms of scene depth and camera

motion, linearize the resulting equation and obtained a least squares solution. Our work, while similar in spirit to that in [20], is motivated by increased use of range scanners and Digital Elevation Maps (DEM) in 3D modeling. There has been considerable interest in fusing direct depth information with the information from image sequences. The available depth information, however, is often coarse and incomplete (may lack data in certain regions). The algorithm presented here is a parallax based algorithm that incorporates a generalized image model and uses the prior depth information in an iterative procedure to estimate the ego-motion and depths.

Previously proposed approaches based on parallax [14][23][11][12] fall into the category of *plane+parallax* where the 3D structure is recovered relative to a reference plane. Such approaches assume the presence of a dominant plane in the scene. However, the assumption of a dominant planar scene is not valid in several scenarios. In this paper, we show how any non-planar (and non-parametric) surface can be used to recover dense 3D structure by computing general surface parallax, thereby not requiring the assumption that a piecewise planar model or a dominant planar surface be present in the scene. An approach for 3D model refinement using surface parallax is presented in [2] but it requires the presence of a small planar surface in the scene for camera motion estimation, which is a restrictive assumption, as well as the use of the BC model, which has the problems discussed above.

We only assume that camera calibration has been computed. The advantages of using our approach are

- The approach can work well with general 3D scenes and does not require the assumption of a dominant planar surface to be present in the scene for alignment or a small planar surface for camera motion estimation.

- We explicitly make use of the parallax direction constraint from the ego-motion estimate, and hence the depth refinement step simplifies to solving a linear system for each pixel.

- Incorporation of a GDI model enables the algorithm to differentiate between the brightness changes due to time-varying illumination with those due to camera motion in the depth refinement phase. In addition, the information obtained from the GDI model is also utilized in estimating the ego-motion, thus leading to better ego-motion estimates.

In Section 2, we present the algorithm in detail, describing the ego-motion estimation and depth refinement procedure using a generalized image model. In Section 3, results on both synthetic and real image sequences are presented. We will provide comparisons with results obtained using a BC model to illustrate the effectiveness of our algorithm. This is followed by conclusions in Section 4.

2. Algorithm

The algorithm uses two intensity images (referred to as *key* and *offset* frames) and an initial coarse and incomplete depth map (referred to as the *reference depth map*) to estimate the ego-motion and the depth map in an iterative fashion (we call these iterations *global iterations*). Let $\mathbf{r} = (x, y)$ denote an image pixel, t denote the time index, $I(\mathbf{r}, t)$ denote the key image and $I(\mathbf{r} - \mathbf{u}\delta t, t - \delta t)$ denote the offset image. Assume a moving camera viewing a rigid scene with no independent motion. The 2D image motion $\mathbf{u}$ for a pixel (x, y) is related to scene depths Z and camera motion by [8]

$$\mathbf{u}(Z, \Theta) = \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} Ah & B \end{bmatrix} \Theta \tag{1}$$

where $B = \begin{bmatrix} \frac{xy}{f} & -(f + \frac{x^2}{f}) & y \\ (f + \frac{y^2}{f}) & -\frac{xy}{f} & -x \end{bmatrix}$, $A = \begin{bmatrix} -f & 0 & x \\ 0 & -f & y \end{bmatrix}$, $\Theta = [T^T, \Omega^T]^T$ (T and Ω denotes the translational and rotational camera velocities) and $h = \frac{1}{Z}$. Based on the linear brightness change model [18], we define our GDI model as

$$M(\mathbf{r}, t) I(\mathbf{r}, t) = I(\mathbf{r} - \mathbf{u}\delta t, t - \delta t) \tag{2}$$

where M denotes the *multiplier field* over the image. If $M(\mathbf{r}, t) = 1$ for all pixels, this reduces to the BC model. Linearizing (2), we get

$$m_t I + \nabla I^T \mathbf{u} + I_t = 0 \tag{3}$$

where $\nabla I = [I_x, I_y]^T$ denotes the spatial image derivatives and I_t denotes the temporal image derivative (here we have used $M = (1 + \delta m)$, $m_t = \lim_{t \to 0} \frac{\delta m}{\delta t}$ as in [21]).

We use Gauss-Newton optimization to estimate the ego-motion and depths using (3). Let us assume that we have some estimate of $\mathbf{u}$, $\mathbf{u}_i = (u_i, v_i)^T$ at the start of i^{th} global iteration (from previous depth and motion estimates). This can be written as a 3-vector $\mathbf{d} = [u_i \delta t, v_i \delta t, \delta t]^T$ in space-time domain. We first derive an equation involving the incremental motion $\delta\mathbf{u}$ given an estimate of $\mathbf{u}$ and then show how to use $\delta\mathbf{u}$ to estimate ego-motion and depths. The gradient of I in the direction $\mathbf{d}$ is

$$\begin{aligned} I_d &= \frac{\mathbf{d}^T}{\| \mathbf{d} \|} \begin{bmatrix} I_x \\ I_y \\ I_t \end{bmatrix} = \frac{1}{\| \mathbf{d} \|} (I_x u_i + I_y v_i + I_t) \delta t \\ &= \frac{1}{\| \mathbf{d} \|} (I_x u_i + I_y v_i - I_x u - I_y v - m_t I) \delta t \\ &= \frac{1}{\| \mathbf{d} \|} (-\nabla I^T \delta\mathbf{u} - \delta m I) \end{aligned} \tag{4}$$

where $\delta\mathbf{u} = [(u-u_i)\delta t, (v-v_i)\delta t]^T$ denotes the incremental 2D motion. The gradient of I in the direction $\mathbf{d}$ can also be written as

$$I_d = \frac{1}{\parallel \mathbf{d} \parallel}(I(\mathbf{r},t) - I(\mathbf{r} - \mathbf{u}_i\delta t, t-\delta t)) = \frac{1}{\parallel \mathbf{d} \parallel}\Delta I \quad (5)$$

where ΔI denotes the difference between the key image and the warped offset image according to $\mathbf{u}_i$. Equating I_d in (4) and (5), we get

$$\nabla I^T \delta\mathbf{u} + \Delta I + \delta m I = 0 \quad (6)$$

In what follows, we show how to estimate the ego-motion, δm and depths using (6).

2.1. Ego-Motion estimation given a depth map and multiplier field

Let Z_i denote the current depth map estimate, Θ_i the current ego-motion estimate, and δm_i the current estimate of the multiplier field, where each is obtained from the previous global iteration (for the first global iteration we use $T = [0,0,0]^T, \Omega = [0,0,0]^T$, δm_0 zero all over the image and Z_0 as the reference depth map). Within each global iteration, we refine the ego-motion estimate by performing n local iterations as follows. Let $\delta\Theta_i$ be the incremental ego-motion update. Using (1), we have

$$\delta\mathbf{u} = Ah_i\delta T + B\delta\Omega = \left[\begin{array}{cc} Ah_i & B \end{array}\right]\delta\Theta_i \quad (7)$$

where $h_i = \frac{1}{Z_i}$. Substituting the above equation in (6), we get

$$\nabla I^T \left[\begin{array}{cc} Ah_i & B \end{array}\right]\delta\Theta_i + \Delta I + \delta m_i I = 0 \quad (8)$$

where ΔI is calculated using $\mathbf{u}_i$ obtained from Θ_i and Z_i. This is a linear system in $\delta\Theta_i$ for each pixel and a least square (LS) solution can be obtained by formulating the above equation for $M > 6$ pixels. Although we can use the pixels from the entire image for obtaining the LS solution, we choose only those pixels (denoted as region R) for which the confidence measure C as obtained from the depth refinement step (Section 2.2) is greater than a predefined threshold. Thus, we can remove pixels which can have erroneous depth and multiplier field estimates. Note that information from both the current depth map estimate and the estimate of the multiplier field is utilized to estimate the ego-motion.

The update $\delta\Theta_i$ is added to the current motion estimate Θ_i to get a refined estimate. Thus, at each local iteration, Θ_i is refined, a new value of ΔI is obtained using refined Θ_i and Z_i and (8) is solved to obtain further refinement. The local iterations are performed until the error in least squares fit stops decreasing. Usually, $n \approx 20$ suffices.

2.2. Depth refinement and estimation of multiplier field

We now show how to refine the depth map and estimate the multiplier field δm for each pixel, given an estimate of the ego-motion and the available depth information. Let T_i, Ω_i denote the current ego-motion estimate and Z_i denote the available depth map estimate (initial reference depth map or estimated from the previous global iteration). Let δZ be the incremental depth map estimate and $Z = Z_i + \delta Z$ be the refined depth map. Using (1), the incremental 2D motion for an incremental change in depth can be written as

$$\delta\mathbf{u} = A(h - h_i)T_i \quad (9)$$

where $h = \frac{1}{Z}, h_i = \frac{1}{Z_i}$. Thus, the incremental 2D motion (*surface parallax field*) is in the direction of the focus of expansion (FOE), i.e, it is an epipolar field. Since we have an estimate of the camera motion T_i from the previous ego-motion estimate, we can constrain the direction of the parallax field. In what follows, let f denote the focal length of the camera which we assume is known. First, let $T_z \neq 0$. Defining the focus of expansion (FOE) as $x_f = f\frac{T_x}{T_z}, y_f = f\frac{T_y}{T_z}$, we constrain the direction of the parallax field to lie along the epipolar direction. Thus for each pixel (x,y) we write

$$\delta\mathbf{u}(x,y) = \beta\mathbf{du}(x,y) \quad (10)$$

where $\mathbf{du}(x,y) = [x - x_f, y - y_f]^T$ denotes the parallax direction and β denotes the parallax magnitude. Thus the parallax magnitude is related to the refined depths as

$$\beta\mathbf{du} = A(h-h_i)T_i = T_z\left[\begin{array}{c} x - x_f \\ y - y_f \end{array}\right](h - h_i) \quad (11)$$

Thus

$$\beta = T_z(h - h_i) \quad (12)$$

Now consider the case when $T_z = 0$. The epipolar field in this case is oriented along the 2D direction $[T_x, T_y]^T$. Hence we define $\mathbf{du}(x,y) = [T_x, T_y]^T$. Thus, the parallax magnitude is related to the refined depths as

$$\beta\mathbf{du} = A(h-h_i)T_i = -f\left[\begin{array}{c} T_x \\ T_y \end{array}\right](h - h_i) \quad (13)$$

Thus

$$\beta = -f(h - h_i) \quad (14)$$

A different formulation can be used to avoid the above two cases corresponding to $T_z = 0$ and $T_z \neq 0$ by parameterizing the parallax direction as $\mathbf{du}(x,y) = [x - x_v, y - y_v]^T$ where $[x_v, y_v]^T$ denotes the projection of the vanishing point of the 3D ray corresponding to the pixel (x,y) (intersection of the 3D ray corresponding to (x,y) with the

plane at infinity). Since the projection of the vanishing point does not depend on camera translation, the two cases corresponding to $T_z = 0$ and $T_z \neq 0$ can be dealt simultaneously. However, for results shown in section 3, the former formulation based on epipole was used.

Using (10), (6) can be written as

$$I_p\beta + \Delta I + \delta m I = 0 \tag{15}$$

where $I_p = \nabla I^T \mathbf{du}$ denotes the projection of the intensity gradient along the parallax direction. We estimate β and δm using the above equation and then use β to obtain refined depths h using (14) or (12), depending on whether T_z is zero or not. Thus, for each pixel we need to estimate two parameters (parallax magnitude β and δm) and the generalized aperture problem is encountered here. The solution is regularized by assuming β and δm to be constant within a local neighborhood. We now derive the tensor based solution for estimating β and δm. The tensor formulation results in a total least square (TLS) [10] solution for the problem, considering errors in all variables.

Let $g = [I_p, I, \Delta I]^T$ and $\gamma = [\beta_1, \beta_2, \beta_3]^T$. The tensor analysis minimizes the cost function

$$J = \left\langle [g^T\gamma]^2 \right\rangle \tag{16}$$

with respect to γ where $<>$ defines the mean operator

$$< f(\overline{x}, \overline{y}) >= \int_{-\infty}^{\infty} w(x - \overline{x}, y - \overline{y}) f(x, y) dx dy \tag{17}$$

where w is a windowing function. The size of the window is related to the neighborhood where the assumption of constant β and δm is valid. The parallax magnitude is then given by $\beta = \frac{\beta_1}{\beta_3}$ and the multiplier field is given by $\delta m = \frac{\beta_2}{\beta_3}$. The tensor based solution is given by minimizing (16) with respect to γ. To avoid the trivial solution $\gamma = 0$, the constraint $\gamma^T\gamma = 1$ is imposed. Using Lagrange multipliers, the error function can be written as

$$J = \left\langle \gamma^T g g^T \gamma \right\rangle + \lambda(1 - \gamma^T\gamma) = \gamma^T G \gamma + \lambda(1 - \gamma^T\gamma) \tag{18}$$

where

$$G = \left\langle g g^T \right\rangle = \begin{bmatrix} < I_p^2 > & < I_p I > & < I_p \Delta I > \\ < I_p I > & < I^2 > & < I \Delta I > \\ < I_p \Delta I > & < I \Delta I > & < \Delta I^2 > \end{bmatrix} \tag{19}$$

Differentiating with respect to γ, we get $G\gamma = \lambda\gamma$. This is a simple eigenvalue system. Since G is a 3×3 real symmetric matrix, there will be three valid eigen-value/eigen-vector pairs. Let $\lambda_1 \geq \lambda_2 \geq \lambda_3$ be the valid eigen-values. The eigen-vector corresponding to λ_3 will be the solution for γ from which the parallax magnitude β and δm can be obtained. Thus, for each pixel, the 3×3 matrix G is constructed and is analyzed to obtain β and δm for that pixel.

Confidence measures based on eigen-values and/or condition number have been proposed in [16][17]. We use $C = (\frac{\lambda_1 - \lambda_3}{\lambda_1 + \lambda_3})^2$ as the confidence measure for the depth refinement procedure. The region R for estimating ego-motion in Section 2.1 is composed of those pixels where C exceeds a pre-defined threshold. The complete algorithm can be described as follows

1. Get the initial reference depth map Z_0, key and offset frames. Set initial multiplier field δm_0 to be zero all over the image, initial ego-motion to be zero and the global iteration index $i = 1$.
2. Estimate the camera motion Θ_i using Z_{i-1} and δm_{i-1} (as described in Section 2.1).
3. Obtain multiplier field δm_i and refined depths Z_i using Θ_i and Z_0 as described in Section 2.2. Obtain confidence measures for depth estimates. Set $i \rightarrow i + 1$.
4. Repeat step 2 (by setting R to those regions in the image where the obtained confidence measure is greater than a pre-defined threshold) and step 3 until ego-motion parameters converge or a pre-specified number of iterations has been reached.

3. Experiments

We present results on both semi-synthetic (i.e. with real textures) and real sequences. All depth maps are color-coded with brighter regions closer to camera. In all depth maps, the region in black corresponds to pixels where estimated depths are either negative or are extremely large ($C = 0$). Also, the shown depth maps are as obtained by the algorithm without any post processing (e.g. for low confidence regions, depths can be interpolated from nearby regions to provide *better looking* estimates). In both experiments, the maximum number of global iterations was set to 10 and the confidence threshold for choosing R for ego-motion estimation was set to 0.3.

3.1. Semi-synthetic images

A semi-synthetic (with real textures) 3D model of an urban environment was rendered in OpenGL. The synthetic 3D model consists of buildings and objects in front of the buildings. We simulate a sequence of images by moving a virtual camera in the scene. The camera motion was in the horizontal direction (1 unit per frame) with no rotation. A directional spotlight (15 degrees spot-cutoff angle) was moved along with the camera. Thus there was significant illumination variation between successive frames. The depth maps were obtained from the OpenGL Z buffer. Figures 1(a) and 1(b) show the key and offset frame respectively. Notice the significant illumination change on the

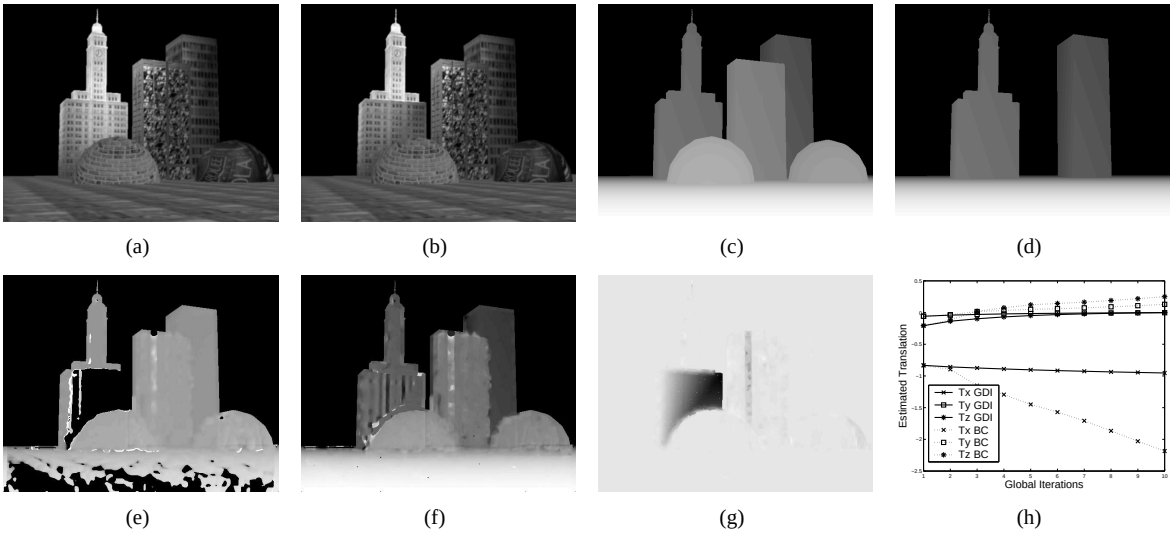

Figure 1. (a) Key frame (b) offset frame (c) true depth map for the key frame (d) reference depth map (e,f) estimated depth maps using BC and GDI models (g) estimated multiplier field (h) ego-motion estimates with iterations

lower part of first building from the left (region behind the sphere) and on the left portions of the the second building from the left. Figures 1(c) and 1(d) show the true depth map and the reference depth map for the scene. Notice that for certain regions (spheres and the building in center), there is no prior depth information in the reference depth map. A constant depth value[1] was chosen for such regions.

Figure 1(h) shows the convergence of ego-motion parameters with global iterations using the GDI and BC models. Observe that the ego-motion parameters were recovered properly using the GDI model but not by using the BC model. Figures 1(e) and 1(f) show the estimated depth map using our algorithm with BC and GDI models respectively. The depth map estimated using the GDI model is better than that obtained by the BC model, especially in regions with large illumination changes. Also notice the depths at the rightmost building which are closer to true depths for GDI model due to better ego-motion estimates. The assumptions of a constant parallax and multiplier field over the neighborhood will not hold at regions where significant intensity and depth discontinuities are present in the scene, thus introducing slight artifacts. The estimated multiplier field using the GDI model is shown in Figure 1(g) (darker values are more negative) which is in accordance with the illumination variation in the key and the offset images. The average intensity values in the key and the offset frames for the lower part of first building from the left (region above the sphere) are 131.04 and 76.26 respectively which gives a multiplier field $\delta m = \frac{76.26}{131.04} - 1 = -0.4180$ (we assume that the induced image motion is small enough for this calculation). The average δm estimated in that region was -0.4246 which is close to the true value.

[1]We assume depths in scene to lie between 0 and Z_{max} and choose $Z_{max}/2$ for all such regions.

3.2. Real images

A video sequence of a scene containing several objects was taken in a lab. The only camera motion was in the X direction with no rotation. A spotlight was rotated along the X axis independent of camera motion (sweeping from top to bottom), thus producing illumination variation across the image. Figures 2(a) and 2(b) show the key and the offset images from the sequence respectively. For the images shown, most of the illumination variation is in the middle one-third region (top to bottom) of the image. For this sequence, we did not have any prior depth information or ground truth for the entire image. Since this is an indoor lab sequence, the variation in the scene depth is small. Therefore, the reference depth map was chosen to be a constant all over the image. Figure 2(f) shows the convergence of ego-motion parameters with global iterations for GDI and BC models. The ego-motion parameters were estimated correctly (since T_y and T_z are zero, depths and T_x can be obtained only up to a scale factor). Figures 2(c) and 2(d) show the estimated depth map using BC and GDI models respectively. We see that by using the BC model, depths can not be obtained reliably, whereas the depth map obtained using the GDI model appears plausible. Notice the finely extracted depth boundaries for different objects. Figure 2(e) show the estimated

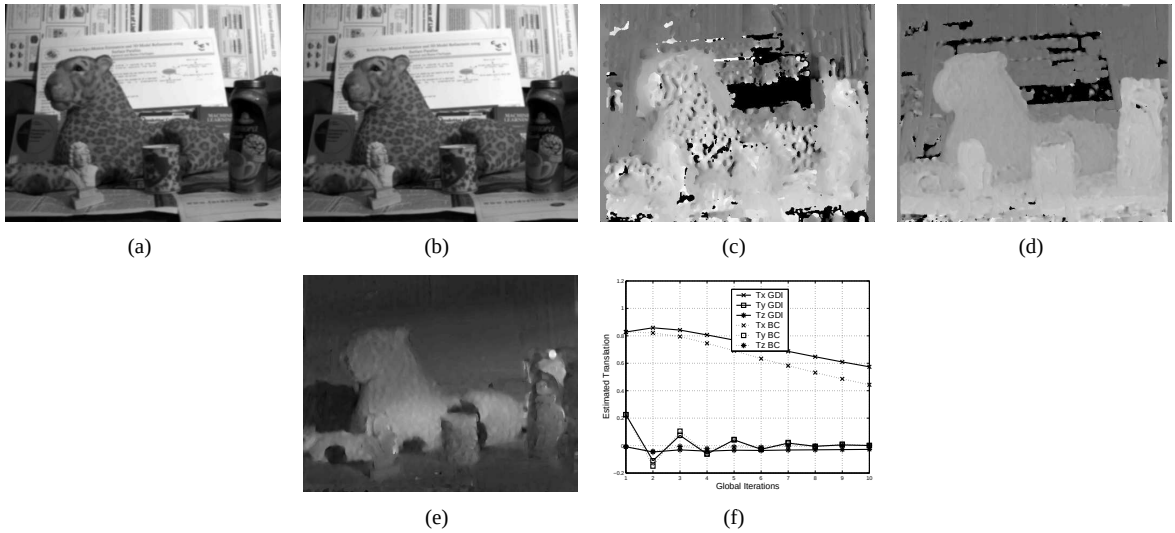

Figure 2. Real example (a) key frame (b) offset frame (c,d) estimated depth maps using BC and GDI models (e) estimated multiplier field (f) ego-motion estimates with iterations

multiplier field (darker values are more negative). Choosing the image region on the tiger between the statue and the cup, the average intensity values in key and offset images in that region were 73.67 and 75.26 respectively which gives a multiplier field $\delta m = 0.0216$. The average δm value estimated in that region is equal to 0.0245.

4. Conclusions

We have proposed a direct iterative algorithm for estimating ego-motion and depths in scenes with varying illumination starting from a coarse and partial depth map using a generalized brightness change model. Using the GDI model and the epipolar constraint from the ego-motion estimate, the depth refinement procedure is formulated as a linear problem which is solved using a tensor based approach. The refined depths, GDI model parameters and confidence measures obtained from depth refinement are used to refine ego-motion. When the depth variations in the scene are small, an initial flat depth can be used without the need for any prior depth information. Results on both real and semi-synthetic images shows the effectiveness of our algorithm. Comparisons with results obtained using a BC model shows that the GDI model performs significantly better.

References

[1] G. Adiv. Determining 3D motion and structure from optical flow generated by several moving objects. *IEEE Trans. Pattern Anal. Machine Intell.*, 7(4):384–401, July 1985.

[2] Y. Aloimonos and C. Brown. Direct processing of curvilinear sensor motion from a sequence of perspective images. In *Proc. Workshop on Computer Vision: Representation and Control*, pages 72–77, 1984.

[3] A. Azarbayejani and A. Pentland. Recursive estimation of motion, structure, and focal length. *IEEE Trans. Pattern Anal. Machine Intell.*, 17:562–575, 1995.

[4] J. Bergen, P. Anandan, K. Hanna, and R. Hingorani. Hierarchical model-based motion estimation. In *Proc. European Conf. Computer Vision*, pages 237–252, Santa Margherita Ligure, Italy, May 1992.

[5] M. Black, D. Fleet, and Y. Yacoob. Robustly estimating changes in image appearance. In *Computer Vision and Image Understanding*, volume 78, pages 8–31, Apr. 2000.

[6] K. Hanna. Direct multi-resolution estimation of ego-motion and structure from motion. In *IEEE Workshop on Motion and Video Computing*, pages 156–162, 1991.

[7] H. Haussecker and D. Fleet. Computing optical flow with physical models of brightness variation. *IEEE Trans. Pattern Anal. Machine Intell.*, 23(6):661–673, June 2001.

[8] B. Horn and E. Weldon. Direct methods for recovering motion. *Int'l J. Computer Vision*, pages 51–76, 1988.

[9] T. Huang and A. Netravali. Motion and structure from feature correspondences: A review. *Proc. IEEE*, 82:252–268, Feb. 1994.

[10] S. Huffel and J. Vandewalle. *The Total Least Squares Problem: Computational Aspects and Analysis*. Society for Industrial and Applied Mathematics, 1991.

[11] M. Irani and P. Anandan. Parallax geometry of pairs of points for 3D scene analysis. In *Proc. European Conf. Computer Vision*, pages 17–30, Cambridge, UK, Apr. 1996.

[12] M. Irani, P. Anandan, and M. Cohen. Direct recovery of planar-parallax from multiple frames. *IEEE Trans. Pattern Anal. Machine Intell.*, 24(11):1528–1534, 2002.

[13] R. Koch, M. Pollefeys, and L. Gool. Realistic surface reconstruction of 3D scenes from uncalibrated image sequences. *J. Visualization and Computer Animation*, 11:115–127, 2000.

[14] R. Kumar, P. Anandan, and K. Hanna. Direct recovery of shape from multiple views: a parallax based approach. In *Proc. 12th IAPR Int'l Conf. Pattern Recognition*, volume 1, pages 685–688, Jerusalem, Israel, 1994.

[15] H. Liu, R. Chellappa, and A. Rosenfeld. A hierarchical approach for obtaining structure from two-frame optical flow. In *Proc. Workshop on Motion and Video Computing*, pages 214–219, Orlando, FL, Dec. 2002.

[16] H. Liu, R. Chellappa, and A. Rosenfeld. Accurate dense optical flow estimation and segementation using adaptive structure tensors and a parametric model. *IEEE Trans. Image Processing*, 12(10):1170–1180, 2003.

[17] H. Liu, T. Hong, M. Herman, and R. Chellappa. A general motion model and spatio-temporal filters for computing optical flow. *Int'l J. Computer Vision*, 22:141–172, 1997.

[18] S. Negahdaripour. Revised definition of optical flow: Integration of radiometric and geometric cues for dynamic scene analysis. *IEEE Trans. Pattern Anal. Machine Intell.*, 20(9):961–979, Sept. 1998.

[19] S. Negahdaripour, N. Kolagani, and B. Hayashi. Direct motion stereo for passive navigation. In *Proc. Conf. Computer Vision and Pattern Recognition*, pages 425–431, June 1992.

[20] S. Negahdaripour and J. Lanjing. Direct recovery of motion and range from images of scenes with time-varying illumination. In *Proc. Int'l Symposium Computer Vision*, pages 467–472, Nov. 1995.

[21] S. Negahdaripour and C. H. Yu. A generalized brightness change model for computing optical flow. In *Proc. Int'l Conf. Computer Vision*, pages 2–11, 1993.

[22] J. Oliensis. A multi-frame structure-from-motion algorithm under perspective projection. *Int'l J. Computer Vision*, 34:1–30, 1999.

[23] H. Sawhney. 3D geometry from planar parallax. In *Proc. Conf. Computer Vision and Pattern Recognition*, pages 929–934, 1994.

[24] G. Stein and A. Shashua. Model based brightness constraints: On direct estimation of structure and motion. *IEEE Trans. Pattern Anal. Machine Intell.*, 22:992–1015, Sept. 2000.

[25] R. Szeliski and S. Kang. Recovering 3D shape and motion from image streams using non-linear least squares. *J. Visual Computation and Image Representation*, 5:10–28, 1994.

[26] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization method. *Int'l J. Computer Vision*, 9:137–154, 1992.

[27] J. Weng, N. Ahuja, and T. S. Huang. Optimal motion and structure estimation. *IEEE Trans. Pattern Anal. Machine Intell.*, 15:864–884, Sept. 1993.

[28] J. Weng, T. Huang, and N. Ahuja. 3D motion estimation, understanding, and prediction from noisy image sequences. *IEEE Trans. Pattern Anal. Machine Intell.*, 9:370–389, 1987.

[29] G. S. Young and R. Chellappa. 3D motion estimation using a sequence of noisy stereo images: Models, estimation, and uniqueness results. *IEEE Trans. Pattern Anal. Machine Intell.*, 12:735–759, Aug. 1990.

[30] L. Zhang, B. Curless, A. Hertzmann, and S. Seitz. Shape and motion under varying illumination: Unifying structure from motion, photometric stereo, and multi-view stereo. In *Proc. 9th IEEE Int'l Conf. Computer Vision*, pages 618–625, Oct. 2003.