A Kalman Filter Approach to Direct Depth Estimation Incorporating Surface Structure

Y.S. Hung, *Member*, *IEEE*, and H.T. Ho

Abstract—The problem of depth-from-motion using a monocular image sequence is considered. A pixel-based model is developed for direct depth estimation within a Kalman filtering framework. A method is proposed for incorporating local surface structure into the Kalman filter. Experimental results are provided to illustrate the effect of structural information on depth estimation.

Index Terms—Depth-from-motion, Kalman filter, gradient method, surface structure, image sequence.

1 Introduction

In this paper, we will consider the depth-from-motion problem, which has applications in mobile robot navigation. Common methods for depth estimation are optical-flow based [1], [3], [11], [15], [13] and feature-based [14], [16]. It is well-known that the estimation of optical flow is an ill-posed problem solvable only by the introduction of additional (e.g., smoothness) constraints [9], and the reconstructed depth and motion are very sensitive to errors in optical flow. Feature-based methods rely on feature correspondence between images and produce only a sparse map at the locations of image features. Another class of methods, namely, gradient-based techniques (also called the direct method) [6], [10], estimates depth directly from the spatiotemporal derivatives of the intensity function and is well-suited for pixel-based (iconic) processing with the advantage of producing a dense depth map. The technique can be used with or without knowledge of camera motion [2], [7], [12].

As depth estimated using two frames is bound to be sensitive to image noise, the Kalman filter is increasingly being used to process a sequence of images [5], [8], [18]. Heel [5] has used the Kalman filter for gradient-based depth estimation where motion is deduced by least-squares estimation. Matthies et al. [8] have investigated Kalman-based algorithms for both optical-flow and feature-based methods for depth-from-motion. In [18], a combination of optical-flow and gradient methods are used to obtain a depth map from known camera motion using the Kalman filter. All these works use image warping techniques in the predictive stage of the Kalman filter.

We will propose in this paper a Kalman-filter-based gradient method for recovering a dense depth map from a sequence of monocular images with known camera motion. Like [5], our method is a direct approach which does not require the estimation of optical flow as an intermediate step, but unlike [5] (and also [8], [18]), we do not use image warping and spatial resampling in our Kalman filter. Instead, our model is based on a local smoothness assumption and the image warping is approximated by first order terms in a Taylor expansion. Furthermore, we exploit the local smoothness condition by incorporating surface structure as an additional "measurement" into the pixel-based Kalman filter. This is in contrast with existing methods which perform smoothing either as a separate process outside the Kalman filter [8], [18], or treat a number of pixels together as an image patch [5]. The advantages of our method are that it is algorithmically simple, and it offers a means for making a direct compromise between measured depth information and a priori known structural information within the same filtering process.

The paper is organized as follows. The direct depth estimation problem is introduced in Section 2. In Section 3, a pixel-based model is developed for depth estimation using the Kalman filter, together with a proposed method for integrating surface structure into the filtering process. Some implementation issues are discussed in Section 4. Experimental results are provided in Section 5. Some concluding remarks are given in Section 6.

2 Direct Depth Estimation

Fig. 1 shows a pinhole camera model with perspective projection. A 3D camera-centered coordinate frame is defined with origin O_c at the centre of projection and the Z-axis along the optical axis of the camera. The image plane, normal to the optical axis at unit focal length $f = 1$, has a 2D coordinate frame with origin on the Z-axis and x and y axes parallel to those of the 3D frame.

Consider an object point $\mathbf{P} = [X\ Y\ Z]^T$ in the 3D frame with a projection $\mathbf{p} = [x\ y]^T$ on the image plane. We will regard the Z-coordinate of the object point $\mathbf{P}$ corresponding to the image point $\mathbf{p}$ as a function of the image coordinates (x, y), and refer to $Z(x, y)$ as the *depth* at the image point $\mathbf{p}$. In our depth estimation problem, we will assume that the camera is moving with known motion in a static environment, and a sequence of monocular images is captured. A moving camera-centered coordinate frame will be adopted. Let $I(x, y, t)$ be the intensity of the image point (x, y) at time t. Our problem is to estimate the depth $Z(x, y, t)$ for all points (x, y) on the image plane using the intensity function $I(x, y, t)$. For notational simplicity, we will suppress the spatial or temporal variables and write $Z(x, y, t)$ as $Z(x, y)$ or $Z(t)$ if appropriate.

Suppose the camera is moving with translational velocity $\tau = [\tau_x\ \tau_y\ \tau_z]^T$ and rotational velocity $\omega = [\omega_x\ \omega_y\ \omega_z]^T$ about O_c. Let I_x, I_y, and I_t denote the partial derivatives of the intensity function $I(x, y, t)$ with respect to x, y, and t, respectively. Horn [6] has shown that depth and motion can be related in terms of spatiotemporal derivatives of the intensity function through the Brightness Change Constraint Equation (BCCE):

$$\frac{\mathbf{s}\cdot\tau}{Z} + \mathbf{q}\cdot\omega = -I_t, \tag{1}$$

where

$$\mathbf{s} = \begin{bmatrix} -I_x \\ -I_y \\ xI_x + yI_y \end{bmatrix},\quad \mathbf{q} = \begin{bmatrix} xyI_x + (1+y^2)I_y \\ -xyI_y - (1+x^2)I_x \\ yI_x - xI_y \end{bmatrix}.$$

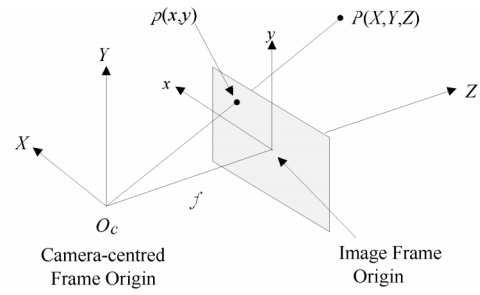

Fig. 1. Camera coordinate system.

- *Y.S. Hung is with the Department of Electrical and Electronic Enginnering, The University of Hong Kong, Pokfulam Road, Hong Kong. E-mail: yshung@eee.hku.hk.*
- *H.T. Ho was with the Department of Electrical and Electronic Enginnering, The University of Hong Kong and is now with KLA-Tencor Corporation, U.S.A.*

Manuscript received 29 Dec. 1997; revised 15 Dec. 1998. Recommended for acceptance by R. Szeliski.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number 107577.

If the camera motion τ and ω are known, (1) can be written as

$$Z = -\frac{\mathbf{s}\cdot\tau}{I_t + \mathbf{q}\cdot\omega}. \quad (2)$$

Hence the depth Z can be estimated directly from the intensity derivatives. The depth estimates given by (2), based on the current frame (and neighboring frames required for time derivative computations), are however bound to be sensitive to image noises and quantization errors. A Kalman filter can be used to integrate the depth information contained in an image sequence.

3 Kalman Filter for Depth Estimation

A state-space model is needed for formulating the depth estimation problem in a Kalman filter framework. We will use the dynamics of the camera motion to formulate the state equation and the BCCE as a measurement equation for the state-space model.

3.1 State-Space Model

For an image of $N_1 \times N_2$ pixels, the system state will consist of the depths of all n ($= N_1N_2$) pixels. Since a full-order Kalman filter is computationally undesirable and pixels not in close proximity of each other are likely to be very weakly coupled, we will assume for simplicity that neighboring pixels are independent of each other. As a result, the full-order n-dimensional Kalman filter can be replaced by n scalar filters attached to individual pixels. Consider a scene point $\mathbf{P}(t) = [X(t)\ Y(t)\ Z(t)]^T$. At time $t + \Delta t$, the position of $\mathbf{P}$ is given by

$$\mathbf{P}(t+\Delta t) \cong \mathbf{P}(t) - \big(\tau + \omega \times \mathbf{P}(t)\big)\Delta t. \quad (3)$$

Denote the interframe translational displacement by $\mathbf{T} = [T_x\ T_y\ T_z]^T$ and rotational displacement by $\Omega = [\Omega_x\ \Omega_y\ \Omega_z]^T$, i.e., $\mathbf{T} = \tau\Delta t$ and $\Omega = \omega\Delta t$. The Z-component of (3) is

$$Z(t+\Delta t) \simeq Z(t) - [T_z + \Omega_x Y(t) - \Omega_y X(t)]. \quad (4)$$

As the projection of the scene point $\mathbf{P}$ on the image plane changes from (x, y) at time t to $(x + \Delta x, y + \Delta y)$ at time $(t + \Delta t)$ due to camera motion, the depths $Z(t)$ and $Z(t + \Delta t)$ of $\mathbf{P}$ are in general system states associated with different image points. More explicitly, we will express $Z(t)$ and $Z(t + \Delta t)$ in (4) as $Z(x, y, t)$ and $Z(x + \Delta x, y + \Delta y, t + \Delta t)$, respectively, and write

$$Z(x+\Delta x, y+\Delta y, t+\Delta t) \simeq Z(x,y,t) - [T_z + \Omega_x Y(x,y,t) - \Omega_y X(x,y,t)] \quad (5)$$

In the image warping approach (e.g., see [5]), the depth map is warped according to the (known or estimated) motion to obtain $Z(x + \Delta x, y + \Delta y, t + \Delta t)$. The warped depth map is then resampled spatially at the image pixel grid to give a predicted value of $Z(x, y, t + \Delta t)$. We will, however, take a different approach here. Assuming local smoothness, a Taylor series expansion in the first two arguments of the left-hand side of (5) gives

$$Z(x+\Delta x, y+\Delta y, t+\Delta t) = Z(x,y,t+\Delta t) + \frac{\partial Z}{\partial x}\Delta x + \frac{\partial Z}{\partial y}\Delta y + \varepsilon(x,y,t+\Delta t) \quad (6)$$

where $\varepsilon(x, y, t + \Delta t)$ represents the approximation error. Equating (5) and (6), and making use of $X(x, y, t) = Z(x, y, t)x$ and $Y(x, y, t) = Z(x, y, t)y$ (under the assumption that $f = 1$) yields

$$Z(x,y,t+\Delta t) = \big[1 - \Omega_x y + \Omega_y x\big]Z(x,y,t) - T_z - \frac{\partial Z}{\partial x}\Delta x - \frac{\partial Z}{\partial y}\Delta y - \varepsilon(x,y,t+\Delta t) \quad (7)$$

Since (7) now represents the evolution of the depth Z from t to $t + \Delta t$ for a fixed image point (x, y), we will suppress the spatial dependency of Z. Further, taking t and $t + \Delta t$ to be the kth and $(k + 1)$th sampling instant, we will replace t and $t + \Delta t$ by k and $(k + 1)$, respectively. Hence, (7) shows that the depth at (x, y) satisfies a discrete state equation of the form

$$Z(k+1) = \Phi(k)Z(k) + u(k) + \zeta(k), \quad (8)$$

where we have defined

$$\Phi(k) = 1 - \Omega_x y + \Omega_y x \quad (9)$$

$$u(k) = -T_z - \frac{\partial Z}{\partial x}\Delta x - \frac{\partial Z}{\partial y}\Delta y. \quad (10)$$

In (8), the system matrix $\Phi(k)$ and the term $-T_z$ in $u(k)$ can be determined if camera motion is known. $\zeta(k)$ will be taken to include ε as well as error generated when estimating the terms $\frac{\partial Z}{\partial x}\Delta x$ and $\frac{\partial Z}{\partial y}\Delta y$ in $u(k)$. Since $\left(\frac{\partial Z}{\partial x}, \frac{\partial Z}{\partial y}\right)$ is the gradient of the depth map and $(\Delta x, \Delta y)$ represents image motion, $\frac{\partial Z}{\partial x}\Delta x + \frac{\partial Z}{\partial y}\Delta y$ is the depth variation seen at the fixed image point (x, y) due to motion. These terms can be regarded as a first-order approximation to the image warping and resampling operation. We note that the estimation of $\frac{\partial Z}{\partial x}$ and $\frac{\partial Z}{\partial y}$ is computationally simpler than image warping, but these terms can only be sensibly estimated after the depth map has attained some degree of smoothness. Experimental results suggest that the estimation errors for $\left(\frac{\partial Z}{\partial x}, \frac{\partial Z}{\partial y}\right)$ are fairly random. We will therefore assume that ζ is approximately Gaussian white with zero mean and variance $\mathrm{var}(\zeta) = Q$. To complete the state-space model, we need a measurement equation. For this purpose, we introduce a measurement noise η_1 (assumed Gaussian with zero mean) into BCCE and rewrite (1) as

$$-\mathbf{s}\cdot\tau = (I_t + \mathbf{q}\cdot\omega)Z + \eta_1, \quad \mathrm{var}(\eta_1) = R_1. \quad (11)$$

Defining $Y_1(k) = -\mathbf{s}\cdot\tau$ as the measurement and $H_1(k) = I_t + \mathbf{q}\cdot\omega$, (11) can be written as

$$Y_1(k) = H_1(k)Z(k) + \eta_1(k). \quad (12)$$

Equations (8) and (12) together form a state-space model for the depth Z at the image point (x, y). Clearly, there are other ways of defining the measurement. For example, we can use (2) (i.e., $Z = \frac{Y_1}{H_1}$) directly as a depth measurement. This is, however, bound to be unreliable when both Y_1 and H_1 are small. The choice of (12) has the advantage that Z is scaled by H_1 to produce the output Y_1, which helps to reduce the measurement noise when H_1 is small. Although the use of Z instead of the disparity $\left(\frac{1}{Z}\right)$ as the system state may cause conditioning problems when the scene contains points with large Z, but Z has the advantage of providing a more direct appeal when we consider the addition of a structural condition in the Kalman filter.

3.2 Kalman Filter

A set of Kalman filter equations for generating an estimate $\hat{Z}(k)$ for $Z(k)$ is given by [8]:

$$\hat{Z}^-(k) = \Phi(k-1)\hat{Z}(k-1) + u(k-1) \quad (13)$$

$$P^-(k) = \Phi(k-1)\,P(k-1)\Phi^T(k-1) + Q(k-1) \quad (14)$$

$$K(k) = P^-(k)H_1^T(k)\Big[H_1(k)P^-(k)H_1^T(k) + R_1(k)\Big]^{-1} \quad (15)$$

$$\hat{Z}(k) = \hat{Z}^-(k) + K(k)\Big[Y_1(k) - H_1(k)_1\hat{Z}^-(k)\Big] \quad (16)$$

$$P(k) = P^{-}(k) - K(k)H_1(k)P^{-}(k), \tag{17}$$

where $\hat{Z}^{-}(k)$ represents the predicted depth at time k before the arrival of the kth measurement, $P^{-}(k)$ and $P(k)$ are the variances of $\hat{Z}^{-}(k)$ and $Z(k)$, respectively, and $K(k)$ is the Kalman gain.

3.3 Integrating Structure Into Kalman Filter

Because the Kalman filters attached to different image pixels are decoupled, the estimated depths of neighboring pixels, say $Z(x, y)$ and $Z(x + 1, y)$, are not related in any explicit manner. If the intensity of each pixel is corrupted by independent white noise, the estimated depth for each pixel will contain a random component. As a result, a planar surface may appear as a rugged surface with spikes projecting from some mean position of the surface. A common practice (e.g., [8], [18]) is to perform spatial smoothing to remove excessive depth variations.

We will propose a different approach for handling spatial relationships between the depth of neighboring pixels. Our method is based on the assumption that the depth function $Z(x, y)$ satisfies some local structural property which can be expressed as

$$Z(x, y) = g(Z(x + p_1, y + q_1), \ldots, Z(x + p_s, y + q_s)) - \delta, \tag{18}$$

where g is defined over a mask indexed by p_i, q_i $(i = 1, \ldots, s)$ around the pixel (x, y), and δ represents permissible variations in the local surface structure with variance R_δ. For example, if $Z(x, y)$ is a "continuous" function of (x, y) with bounded variations (less than an upper limit M) between neighboring pixels, we may impose the condition that

$$Z(x,y) = \tfrac{1}{4}\left[Z(x-1,y) + Z(x,y-1) + Z(x+1,y) + Z(x,y+1)\right] - \delta, \tag{19}$$

where $|\delta| \le M$. In (19), $g(\cdot)$ is simply an averaging function. Other functions can be devised for different kinds of surface structures. We wish to recast (18) as a measurement equation. By (18), if the estimated depths of neighboring pixels are known, we can take

$$Y_2 = g\left(\hat{Z}(x + p_1, y + q_1), \ldots, \hat{Z}(x + p_s, y + q_s)\right) \tag{20}$$

as an estimate for $Z(x, y)$ based on a priori known structure of the surface. Let ρ be the error in estimating $Z(x, y)$ arising from the replacement of Z by $\hat{Z}$ in $g(\cdot)$, that is,

$$\rho = g\left(\hat{Z}(x + p_1, y + q_1), \ldots, \hat{Z}(x + p_s, y + q_s)\right) - g\left(Z(x + p_1, y + q_1), \ldots, Z(x + p_s, y + q_s)\right)$$

Subtracting (18) from (20) yields

$$Y_2 = Z(x, y) + \eta_2, \tag{21}$$

where $\eta_2 = \delta + \rho$ represents the total noise in using Y_2 as a "measurement" for $Z(x, y)$. With the extra measurement Y_2, (12) and (21) can be combined to give

$$Y = HZ + \eta, \tag{22}$$

where

$$Y = \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix}, H = \begin{bmatrix} H_1 \\ 1 \end{bmatrix}, \eta = \begin{bmatrix} \eta_1 \\ \eta_2 \end{bmatrix}. \tag{23}$$

The measurement equation (12) of the state-space model will now be replaced by (22). The Kalman filter equations (13) to (17) given in Section 3.2 remain applicable if we replace H_1 by H and R_1 by the noise variance matrix:

$$R = \text{var}(\eta) = diag\{R_1, R_2\}. \tag{24}$$

Note that in (24) we have assumed that η_1 and η_2 are uncorrelated, so that R is diagonal. The parameter $R_2 = \text{var}(\eta_2)$ determines the significance of the surface structure equation in the filtering process. If R_2 is large, indicating a lack of confidence in the surface structure, then (21) will have little effect on the depth estimates. If R_2 is chosen to be small, (21) will have a heavier weighting relative to the BCCE measurement, and the depth estimates will quickly settle down into some surface structure satisfying (21). This is discussed further in the next subsection.

3.4 Compliance of Depth Estimates With Surface Structure

Consider the innovation process

$$\gamma = \begin{bmatrix} \gamma_1 \\ \gamma_2 \end{bmatrix} = \begin{bmatrix} Y_1 - H_1\hat{Z} \\ Y_2 - \hat{Z} \end{bmatrix} \tag{25}$$

associated with the Kalman filter. It is well-known that γ is a whitened process with variance $\text{var}(\gamma) = HPH^T + R$. It follows from (23) and (24) that

$$\text{var}(\gamma_1) = H_1^2 P + R_1 \tag{26}$$

$$\text{var}(\gamma_2) = P + R_2. \tag{27}$$

$\text{var}(\gamma_2)$ can be taken as an indication of how far the depth estimates deviate from the surface structure defined by (21). We shall show that if R_2 is chosen sufficiently small, the depth estimates can be forced to comply with the surface structure equation. By the matrix identity

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(C^{-1} + DA^{-1}B)^{-1}DA^{-1},$$

(15) and (17) can be written

$$K(k) = \frac{P^{-}(k)}{1 + P^{-}(k)H^T R^{-1} H} H^T R^{-1}, \tag{28}$$

$$P(k) = \frac{P^{-}(k)}{1 + P^{-}(k)\left(\frac{H_1^2}{R_1} + \frac{1}{R_2}\right)} \tag{29}$$

From (28) and (29), we see that if $R_2 \to \infty$, then the surface structure plays no role in the filtering process and the Kalman filter reduces to the case when BCCE is the only measurement. Clearly, R_2 should take a small value for the surface structure equation to have a significant effect on the filtering process. Let

$$\beta = \frac{H_1^2}{R_1} + \frac{1}{R_2}.$$

If R_2 is small so that

$$\frac{1}{R_2} >> \frac{H_1^2}{R_1},$$

then $\beta \cong \frac{1}{R_2}$. Making use of (14), (29) can be written as

$$P(k) = \frac{\Phi^2(k-1)P(k-1) + Q}{1 + \beta\left[\Phi^2(k-1)P(k-1) + Q\right]}. \tag{30}$$

It follows that

$$P(k) < P(k-1) \Leftrightarrow \beta\Phi^2 P^2(k-1) + [\beta Q + (1 - \Phi^2)]\, P(k-1) - Q > 0. \tag{31}$$

The right-hand side of (31) is a quadratic form in $P(k-1)$ admitting a positive root at

$$P_+ = \frac{\sqrt{\left[\beta Q + \left(1 - \Phi^2\right)\right]^2 + 4\beta\Phi^2 Q} - \left[\beta Q + \left(1 - \Phi^2\right)\right]}{2\beta\Phi^2}. \tag{32}$$

If $P(k-1) > P_+$, then the sequence $P(k)$ will decrease towards P_+. From (27), we have

$$\text{var}(\gamma_2) \to P_+ + R_2.$$

Hence, if R_2 is chosen sufficiently small (relative to R_1 / H_1^2), then P_+ as well as $\text{var}(\gamma_2)$ can be reduced to some appropriately small

value, thereby forcing the depth estimates from the Kalman filter to comply with the surface structure equation. This provides some guidance as to how $Z(k)$ and $P(k)$ behave in accordance with the choice of R_2.

4 Implementation Issues

4.1 Computation of Y_2

In (20), Y_2 is determined in terms of the current depth estimates of neighboring pixels. In practice, the Kalman filters of individual pixels are updated sequentially, and some neighboring pixels required for computing Y_2 may not have been updated yet. If the local continuity condition (19) is to be integrated into the filtering process, then Y_2 should be computed as

$$Y_2(x,y) = \frac{1}{4}\Big[\hat{Z}(x-1,y) + \hat{Z}(x,y-1) + \hat{Z}(x+1,y) + \hat{Z}(x,y+1)\Big] \quad (33)$$

Suppose the Kalman filters are updated in a row-by-row manner starting from the upper-left corner. Then, the last two terms of (33), namely, $\hat{Z}(x+1,y)$ and $\hat{Z}(x,y+1)$, will not be available at the time when the pixel (x, y) is being updated. To overcome this, one approach is to use a combination of updated estimates $\hat{Z}$ and predicted estimates $\hat{Z}^-$ and modify (33) as

$$Y_2(x,y) = \frac{1}{4}\Big[\hat{Z}(x-1,y) + \hat{Z}(x,y-1) + \hat{Z}^-(x+1,y) + \hat{Z}^-(x,y+1)\Big] \quad (34)$$

Alternatively, we may consider replacing the four-sided continuity condition (33) by a two-sided condition based only on depth estimates which have been updated in the current frame, i.e.,

$$Y_2(x,y) = \frac{1}{2}\Big[\hat{Z}(x-1,y) + \hat{Z}(x,y-1)\Big]. \quad (35)$$

This measurement is, however, spatially biased and may produce diagonal effects in the propagation of depth estimates. To overcome this, each image frame can be filtered four times starting from different corners, producing with four different versions of Y_2, say, Y_2^1, Y_2^2, Y_2^3, and Y_2^4. The final estimate is then taken to be

$$Y_2 = \frac{1}{4}\Big[Y_2^1 + Y_2^2 + Y_2^3 + Y_2^4\Big]. \quad (36)$$

4.2 Estimation of Measurement Noise Variances

As the measurement noise variance R_1 determines the weighting between previous and current measurements in the Kalman filter (see (15) and (16)), identification of R_1 is crucial. It is, however, difficult to determine R_1 prior to filtering because of the complex nature of the measurement noise (including errors in numerical differentiation, image noise, and uncertainties in camera motion). We have adopted an online method for estimating the noise covariance matrix for a general time-varying linear system given in [4], which is based on an innovations sequence obtained by running the Kalman filter with an initial guess for R_1 and then reestimating R_1 using least-squares techniques. It remains to determine the variance R_2 for the surface structure. Assuming that the two components δ and ρ are independent, we have

$$R_2 = \mathrm{var}(\eta_2) = R_\delta + \mathrm{var}(\rho). \quad (37)$$

If the function $g(\cdot)$ is linear in the depth estimates, then $\mathrm{var}(\rho)$ can be expressed in terms of the variances of the estimation errors. For example, for the two-sided continuity condition given by (35),

$$\rho = \frac{1}{2}\Big[\big(\hat{Z}(x-1,y) - Z(x-1,y)\big) + \big(\hat{Z}(x,y-1) - Z(x,y-1)\big)\Big]$$

and hence

$$\mathrm{var}(\rho) = \frac{\lambda}{4}\Big[P(x-1,y) + P(x,y-1)\Big], \quad (38)$$

where the factor λ is introduced to compensate for any underestimation of $\mathrm{var}(\rho)$ due to the assumption that neighboring pixels are decoupled. By empirical means, it is found that a suitable range for λ is 1.6–2.0. Substituting (38) into (37) gives

$$R_2 = R_\delta + \frac{\lambda}{4}\Big[P(x-1,y) + P(x,y-1)\Big]. \quad (39)$$

When the depth estimates of neighboring pixels are accurate (with small P), R_2 is small and the surface structure equation (21) will have a significant effect on the filtering process. However, when the depth estimates of neighboring pixels are noisy (with large P), R_2 is large, thus reducing the effect of the surface structure equation and leaving the BCCE to have a more dominant effect among the two measurement equations. By the space-varying nature of R_2 as given in (39), we note that pixels with small variance will be able to influence their neighbors with large variance through the surface structure equation, but less so the other way around. Hence, the determination of R_2 through (39) has two desirable features. First, it allows the surface structure to be imposed only when the depth estimates are good, and second, it enables the depth estimates to propagate from "good" regions (with small variances) to "bad" regions (with large variances) through the assumed structure of the surface.

4.3 Occluding Boundaries

As occlusion is characterized by discontinuity in depth, we can detect occluding boundaries by estimating the local gradient $\left(\frac{\partial Z}{\partial x}, \frac{\partial Z}{\partial y}\right)$ for each pixel and thresholding the magnitude of the gradient. For the Kalman filter incorporating structure, the structural relationship should be regarded as broken across an occluding boundary. In practice, this can be readily implemented by suppressing the structural relationship (in the direction of large gradient) in the Kalman filter at pixels detected to be on an occluding boundary. We note however that occlusion detection and spatial smoothing are opposing objectives, as the former relies on depth discontinuities whereas the latter tends to smooth out discontinuities in depth. Within the proposed framework, the gradient thresholding can be expected to successfully detect occluding boundaries having large jumps in depth, but boundaries with a small step change in depth may be missed and the boundary smoothed by the Kalman filter.

5 Experimental Results

The Kalman filter incorporating surface structure is evaluated using an image sequence captured from a scene shown in Fig. 2a consisting of a soda can and a small box placed in front of a planar poster featuring the drawing *Relativity* by M.C. Escher. The planar poster is located at Z = 1,200 mm. The soda can is placed with its nearest point at a distance of 1,100 mm from the camera. The box, of depth 120 mm, is placed against the poster. Fig. 2b shows the depth map of the ground truth. A 30-frame image sequence is captured as the camera undergoes uniform lateral translation of 0.65 mm per frame. The images are digitized to 180×300 pixels in 256 gray levels.

The first 20 frames of the image sequence is processed using the Kalman filter without the surface structure equation. The following initial values and filter parameters are adopted:

$$\hat{Z}(0) = 10^4,\ P(0) = 10^8,\ Q = 100,\ R_1(0) = 10^6. \quad (40)$$

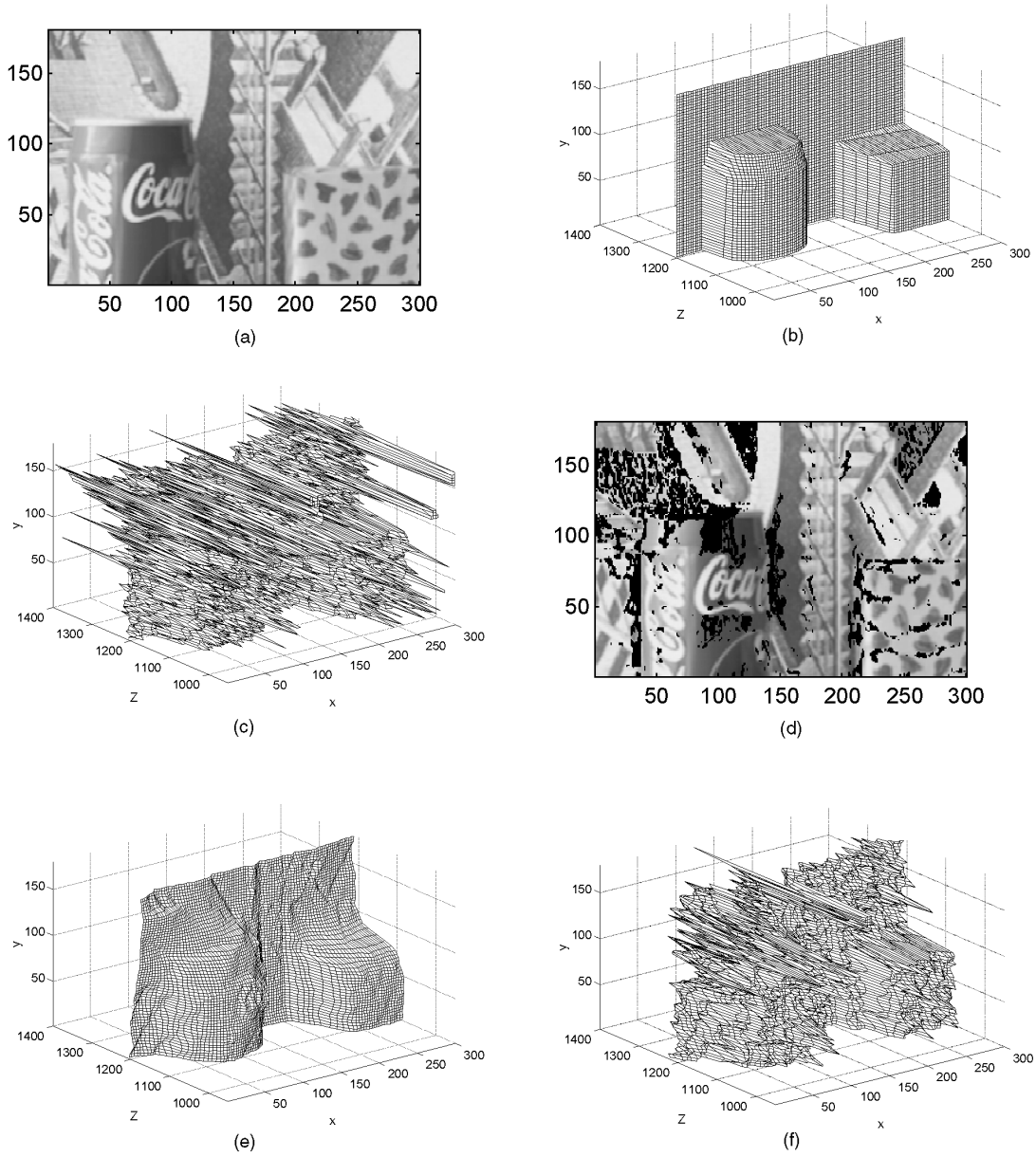

Fig. 2. Experimental results. (a) Frame 1 of the image sequence. (b) Ground truth at frame 20. (c) Depth map at frame 20 using Kalman filter without smoothness condition. (d) Frame 20 showing points (marked in black) with relative error greater than 5 percent. (e) Depth map at frame 28 using Kalman filter with smoothness condition. (f) Depth map at frame 28 with smoothness outside the Kalman filter.

Fig. 2c shows a 3D plot of the estimated depth map at frame 20. The reconstructed depth map is very rugged with large spikes and the 3D structure of the scene is hardly recognizable. Using knowledge of the ground truth, regions of poor estimation with a relative error greater than 5 percent, that is,

$$\left|\hat{Z}(x,y) - Z(x,y)\right| > 0.05Z(x,y),$$

are identified and marked in "black" at frame 20 of the sequence, as shown in Fig. 2d. The area above the soda can is poorly estimated because of the lack of intensity variations. A direct verification shows that there is a good match between the marked points shown in Fig. 2d and points where $P(x,y)^{\frac{1}{2}}$ is large. Thus $P(x,y)^{\frac{1}{2}}$ serves as an indication of the reliability of the depth estimates.

We next proceed to incorporate surface structure into the Kalman filter. Since it is not sensible to impose surface structure during the initial stage of the filtering process when depth estimates fluctuate wildly, the Kalman filter is first run with the BCCE as the only measurement equation generating the results given in Fig. 2c, and the surface structure is then incorporated into the Kalman filter starting from frame 21. The observation Y_2 is computed according to (35) and (36), and the parameters associated with smoothing are chosen to be

$$R_\delta = 100, \quad \lambda = 2. \tag{41}$$

Since $\frac{\partial Z}{\partial x}, \frac{\partial Z}{\partial y}$ can only be reasonably estimated after smoothing has been performed, the terms $-\frac{\partial Z}{\partial x}\Delta x - \frac{\partial Z}{\partial y}\Delta y$ in (10) are included in the model only after frame 22. Fig. 2e shows the reconstructed surface at frame 28. It can be seen that the spatial smoothing within the Kalman filtering has been effective in reconstructing the surface structure of the 3D scene with the shape of the box and the soda can reasonably recovered. Some further remarks follow.

Remark 1: Choice of Filter Parameters

The parameters that need to be chosen are listed in (40) and (41). Q was chosen to reflect the expected errors in the estimation of $\left(\frac{\partial Z}{\partial x}, \frac{\partial Z}{\partial y}\right)$. Our experiments suggest that the results are not sensitive to the choice of Q by up to an order of magnitude. The choice of $P(0)$ (provided that it is large) and $\hat{Z}(0)$ has little influence on the outcome either. R_1 is determined by an online algorithm as discussed in Section 3.2. R_δ is a user parameter that requires tuning to achieve the desired smoothing. The effect of R_δ is very apparent in the Kalman filtering process as it directly controls the compromise between the BCCE measurement and structural information. It should be noted that if R_δ is chosen to be small, the variance P can become correlated dominantly with the smoothness condition, and it may be desirable to reset P to reinstate its correlation with the BCCE.

Remark 2: Computation of I_x, I_y, and I_t

It is necessary to perform a preliminary spatial smoothing to remove noise before computing these derivatives. We have used a 5×5 Gaussian mask to filter the image before applying a central difference equation based on a fourth-order polynomial interpolation to compute the spatial and temporal derivatives. We have found that a higher (sixth-) or a lower (second-) order polynomial interpolation does not make any significant difference to the results.

Remark 3: Comparison With Other Methods

The depth map for the above image sequence was estimated using two other techniques. First, we consider replacing the system equation (8) by an image warping and resampling model. We note that the warping and resampling operation tends to produce false occlusions at the initial stage of depth estimation when the depth map contains noisy spikes. Hence, it seems sensible to perform image warping in combination with some form of smoothing. If the image warping and resampling model is used after frame 21 together with the proposed smoothing scheme, then the estimated depth map at frame 28 is similar to that shown in Fig. 2e. Second, we remove the smoothness equation from the Kalman filter and instead perform the smoothing outside the filter. In the smoothing operation, $Z(x, y)$ is replaced by the weighted average

$$\hat{Z}(x,y) = \sum_{i,j}\left[P(i,j)^{-\frac{1}{2}} Z(i,j)\right] \Big/ \sum P(i,j)^{-\frac{1}{2}},$$

where the summation is taken over the pixel (x, y) plus its four neighbors. As in the case of smoothing within the Kalman filter, the smoothing is started at frame 21. The resultant depth map at frame 28 is shown in Fig. 2f, which shows that smoothing outside the Kalman filter is less effective in removing spikes in areas of poor estimation. With a small R_δ, the depth estimates can converge in fewer number of frames if the smoothing is performed within the Kalman filter rather than outside. In terms of computation time, our Matlab implementation running on a 200-MHz Pentium II processor requires $\sim$2 seconds to process one image frame without smoothing, but filtering with the smoothness condition is more expensive at $\sim$30 seconds for each pass over the entire image plane.

6 Conclusion

We have developed a pixel-based model using the gradient method for direct depth estimation within a unified Kalman filtering framework. A method is proposed for incorporating local surface structural information into the filtering process without sacrificing the simplicity of the iconic filtering algorithm. The example shows that the surface structure equation can make a significant contribution to the recovery of surface structure from an ensemble of depth estimates of individual pixels. We have assumed perfect knowledge of camera motion. In practice, camera motion, even if measured, could be noisy. It would be appropriate to include the estimation of camera motion in the Kalman filtering. Possible approaches would be to formulate the depth estimation as an extended Kalman filter problem to include motion parameters, or use least-squares techniques for motion estimation (e.g., see [5], [17]). An automatic and adaptive method for choosing the parameter R_δ could be another area for further investigation.

Acknowledgments

The authors would like to thank the reviewers for valuable comments which help improve the paper in many respects and the Hong Kong Research Grants Council for financial support (Ref No. HKU 7043/98E).

References

[1] G. Adiv, "Determining Three-Dimensional Motion and Structure From Optical Flow Generated by Several Moving Objects," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 7, no. 4, pp. 384–401, 1985.

[2] J.Y. Aloimonos, I. Weiss, and A. Bandyopadhyay, "Active Vision," *Int'l J. Computer Vision*, vol. 1, pp. 333–356, 1987.

[3] Y. Aloimonos and Z. Duric, "Estimating the Heading Direction Using Normal Flow," *Int'l J. Computer Vision*, pp. 33–56, 1994.

[4] P.R. Belanger, "Estimation of Noise Covariance Matrices for a Linear Time-Varying Stochastic Process," *Automatica*, vol. 10, pp. 267–274, 1974.

[5] J. Heel, "Direct Dynamic Motion Vision," *Proc. IEEE Conf. Robotics and Automation*, pp. 1,142–1,147, 1990.

[6] B.K.P. Horn and E.J. Weldon, "Direct Methods for Recovering Motion," *Int'l J. Computer Vision*, vol. 2, pp. 51–76, 1988.

[7] L. Huang and Y. Aloimonos, "How Normal Flow Constrains Relative Depth for an Active Observer," *Image and Vision Computing*, pp. 435–445, 1994.

[8] L. Matthies, T. Kanade, and R. Szeliski, "Kalman Filter-Based Algorithms for Estimating Depth From Image Sequences," *Int'l J. Computer Vision*, vol. 3, pp. 209–236, 1989.

[9] H.-H. Nagel and W. Enkelmann, "An Investigation of Smoothness Constraints for the Estimation of Displacement Vector Fields From Image Sequences," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 8, pp. 565–593, 1986.

[10] S. Negahdaripour and B.K.P. Horn, "Direct Passive Navigation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 9, no. 1, pp. 168–176, 1987.

[11] K. Prazdny, "Determining the Instantaneous Direction of Motion From Optical Flow Generated by a Curvilinearly Moving Observer," *Computer Vision, Graphics and Image Processing*, vol. 17, pp. 238–248, 1981.

[12] G. Sandini and M. Tistarelli, "Active Tracking Strategy for Monocular Depth Inference Over Multiple Frames," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 12, no. 1, pp. 13–27, 1990.

[13] D. Sinclair, A. Blake, and D. Murray, "Robust Estimation of Egomotion From Normal Flow," *Int'l J. Computer Vision*, pp. 57–69, 1994.

[14] R.Y. Tsai and T.S. Huang, "Uniqueness and Estimation of Three-Dimensional Motion Parameters of Rigid Objects With Curved Surfaces," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 6, no. 1, pp. 337–351, 1984.

[15] A.M. Waxman, B. Kamgar-Parsi, and M. Subbarao, "Closed-Form Solutions to Image Flow Equations for 3D Structure and Motion," *Int'l J. Computer Vision*, vol. 1, pp. 239–258, 1987.

[16] J. Weng, T.S. Huang, and N. Ahuja, *Motion and Structure From Image Sequences*. New York: Springer-Verlag, 1993.

[17] Y. Xiong and S.A. Shafer, "Dense Structure From a Dense Optical Flow Sequence," *Int'l Symp. Computer Vision*, pp. 1–6, Coral Gables, Fla., 1995 (also Carnegie Mellon University Technical Report CMU-RI-TR-95-10).

[18] H. Zhuang, R. Sudhakar, and J. Shieh, "Depth Estimation From a Sequence of Monocular Images With Known Camera Motion," *Robotics and Autonomous Systems*, vol. 13, pp. 87–95, 1994.