Incremental Estimation of Dense Depth Maps from Image Sequences[1]

Larry Matthies, Richard Szeliski, and Takeo Kanade
Computer Science Department
Carnegie-Mellon University
Pittsburgh, PA 15213

Abstract

Kalman filtering has recently been proposed as a mechanism for obtaining on-line estimates of depth from motion sequences. Previous applications of Kalman filtering to depth from motion have been limited to estimating depth at the location of a sparse set of features. In this paper, we introduce a new, pixel-based (*iconic*) algorithm that estimates depth and depth uncertainty at each pixel and incrementally refines these estimates over time. We describe the algorithm for translations parallel to the image plane and contrast its formulation and performance to that of a feature-based Kalman filtering algorithm. We compare the performance of the two approaches by analyzing their theoretical convergence rates, by conducting quantitative experiments with images of a flat poster, and by conducting qualitative experiments with images of a realistic outdoor scene model. The results show that the new method is an effective way to extract depth from lateral camera translations and suggest that it will play an important role in low-level vision.

1 Introduction

Using known camera motion to estimate depth from image sequences is important in many applications of computer vision to robot navigation and manipulation. Many applications require an algorithm that operates in an on-line, incremental fashion. Such algorithms require a depth representation that includes not only the current depth estimate, but also an estimate of the uncertainty in the current depth map.

Previous work [3] [5] [7] [11] [17] has identified Kalman filtering as a viable framework for this problem, because it incorporates representations of uncertainty and provides a mechanism for incrementally reducing uncertainty over time. To date, applications of this framework have largely been restricted to estimating the positions of a sparse set of trackable features, such as points or line segments [6] [12]. While this is adequate for many robotics applications, it requires reliable feature extraction and it fails to describe large areas of the image. Another line of work has addressed the problem of extracting denser depth or displacement estimates from image sequences. However, these approaches either have been restricted to two frame analysis [1] or have used batch processing of the image sequence, for example via line fitting [4] or spatio-temporal filtering [8].

In this paper we introduce a new, pixel-based (*iconic*) approach to incremental depth estimation and compare it mathematically and experimentally to a feature-based approach we developed previously [11]. The new approach represents depth and depth variance at every pixel and uses Kalman filtering to extrapolate and update the pixel-based depth representation. The algorithm uses correlation to measure optical flow and to estimate the variance in the flow, then uses the known camera motion to convert the flow field into a depth map. It then generates an updated depth map from a weighted combination of the new measurements and the prior depth estimates. Regularization is employed to smooth the depth map and to fill in underconstrained areas. The resulting algorithm is parallel, uniform, and can take advantage of mesh-connected or multi-resolution (pyramidal) processing architectures.

The remainder of this paper is structured as follows. In the next section, we give a brief review of Kalman filtering and introduce our overall approach to Kalman filtering of depth. We then describe our new, pixel-based depth from motion algorithm and the feature-based algorithm to which it will be compared. We then analyze the theoretical accuracy of both methods, compare them both to the theoretical accuracy of stereo matching, and verify this analysis experimentally using images of a flat scene. We also show the performance of both methods on images of realistic outdoor scene models. In the final section, we discuss the promise and the problems involved in extending the new method to arbitrary motion.

2 Estimation framework

Kalman filtering is a powerful technique for real-time estimation in dynamic systems. The Kalman filter models the *current state* of a system as a vector u_t and uses three separate probabilistic models to generate an estimate of the current state (Table 1). The *system model* describes the evolution over time of the state vector u_t as multiplication by a known transition matrix Φ_t and addition of Gaussian noise with covariance Q_t. The *measurement* (or *sensor*) *model* relates a measurement vector d_t to the current state through a measurement matrix H_t and addition of Gaussian noise with covariance R_t. The *prior model* describes the knowledge about the system state $\hat{u}_0$ and its covariance P_0 before the first measurement is taken.

A Kalman filter algorithm operates in two phases: prediction and update (Table 1 and Figure 1). At time t, the previous

[1]This research was sponsored in part by DARPA, monitored by the Air Force Avionics Lab under contract F33615-87-C-1499 and in part by a postgraduate fellowship from the FMC Corporation. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the funding agencies.

Models	
system model	$u_t = \Phi_t u_{t-1} + \eta_t, \eta_t \sim N(0, Q_t)$
measurement model	$d_t = H_t u_t + \xi_t,\ \xi_t \sim N(0, R_t)$
prior model	$\mathrm{E}[u_0] = \hat{u}_0,\ \mathrm{Cov}[u_0] = P_0$
(other assumptions)	$\mathrm{E}[\eta_t \xi_j^T] = 0$
Prediction phase	
state estimate extrapolation	$\hat{u}_t^- = \Phi_{t-1} \hat{u}_{t-1}^+$
state covariance extrapolation	$P_t^- = \Phi_{t-1} P_{t-1}^+ \Phi_{t-1}^T + Q_{t-1}$
Update phase	
state estimate update	$\hat{u}_t^+ = \hat{u}_t^- + K_t[d_t - H_t \hat{u}_t^-]$
state covariance update	$P_t^+ = [I - K_t H_t] P_t^-$
Kalman gain matrix	$K_t = P_t^- H_t^T [H_t P_t^- H_t^T + R_t]^{-1}$

Table 1: Kalman filter equations

state and covariance estimates, $\hat{u}_{t-1}^+$ and P_{t-1}^+, are extrapolated to predict the current state and covariance, $\hat{u}_t^-$ and P_t^-. The predicted covariance is then used to compute the new Kalman gain matrix K_t and the updated covariance matrix P_t^+. Finally, the measurement residual $d_t - H_t \hat{u}_t^-$ is weighted by the gain matrix K_t and added to the predicted state u_t^- to yield the updated state u_t^+.

Kalman filtering is usually applied to systems with fairly small numbers of state variables. In the domain of motion sequence analysis, it has been used to track sparse features [2] [3] [12], but has not previously been used in conjunction with *dense* fields such as iconic depth maps. We will briefly describe how this estimation framework is used in our depth from motion algorithm. Section 3 describes the details of the implementation for lateral camera motion; extensions to general motion are considered in [13].

In our case, the system state is a representation of the depth at every pixel (x, y) in the current image. We choose to represent the inverse depth $u(x, y) = 1/Z(x, y)$, which we call "disparity", plus the variance in the disparity, $\sigma^2(x, y)$. There are several reasons for representing disparity instead of the actual depth $Z(x, y)$. Disparity can be linearly related to optical flow measurements, it is better conditioned for distant objects, and, for lateral camera motion, the scaled disparity and variance can be used directly to set search limits on correspondence in the subsequent image.

The system model uses the current depth map and an estimate of the camera motion to predict a depth map for the next image in the sequence. This is implemented by using the predicted optical flow to warp the depth map, then resampling the warped map to compute the predicted disparity at each pixel. The measurement model simply produces a measurement of the disparity at every pixel, so that $H_t = I$. The disparity measurement in turn is based on an optical flow measurement obtained by a correlation-based flow estimator. Estimates of the variance $\sigma_d^2(x, y)$ of disparity measurements are computed from the variance of the optical flow. These variance estimates are essential because they characterize the difference between reliable measurements and unreliable measurements, such as the difference between measurements obtained in highly textured areas and those obtained in areas of uniform intensity. The update phase combines the new disparity measurements with the predicted depth map to generate an updated depth map. Finally, the prior model embeds prior knowledge about the scene. In particular, smoothness constraints, which require nearby points to have similar disparity, can be modeled by off-diagonal elements in the prior covariance matrix P_0. This is equivalent to modeling the disparity map as a Markov Random Field [19].

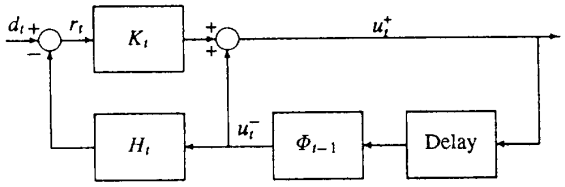


Figure 1: Kalman filter block diagram

3 Iconic depth estimation

Our implementation of this framework consists of four main stages (see Figure 2). The first stage uses correlation to produce a measurement of the disparity at each pixel and an estimate of the associated variance. The second stage integrates this information with the disparity map predicted from the previous time step. The third stage uses regularization-based smoothing to reduce measurement noise and to fill in areas of unknown disparity. The last stage uses known camera motion to predict the disparity field that will be seen in the next frame and re-samples the field to keep it pixel-based. Here we deal only with camera translation parallel to the image plane; in this case, estimating disparity is equivalent to estimating optical flow. Extensions to arbitrary camera motion are described in [13].

3.1 Measurement (correlation)

The problem of extracting optical flow from a sequence of intensity images has been extensively studied in computer vision [1] [8] [9]. The approach used in this paper is a simple version of correlation-based matching known as the Sum of Squared Differences (SSD) method [1]. This technique integrates the squared intensity difference between two shifted images over a small area. For the case of lateral motion, this error measure is

$$e_t(d; x, y) = \int\int w(\lambda, \eta)[f_t(x-d+\lambda, y+\eta) - f_{t-1}(x+\lambda, y+\eta)]^2 \, d\lambda \, d\eta,$$

where f_t and f_{t-1} are the two intensity images and $w(\lambda, \eta)$ is a weighting function. This measure is computed at each pixel for a number of possible disparity values d. In [1], a coarse-to-fine technique is used to limit the range of possible flow values. In our images, the possible range of values is small (since we are using small-motion sequences), so a single-resolution

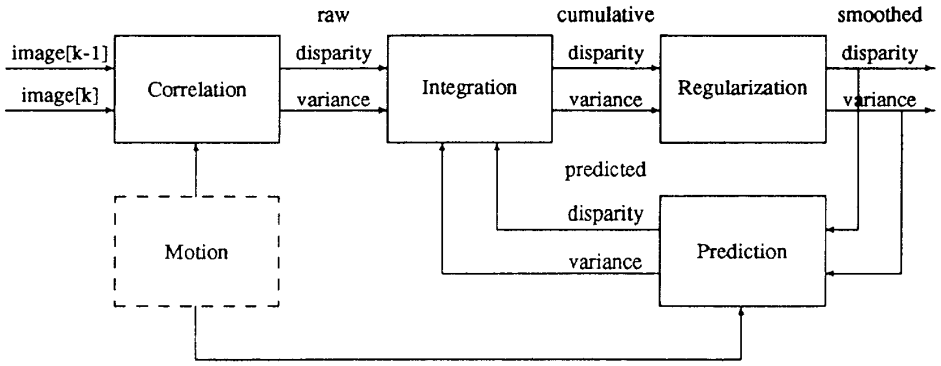


Figure 2: Iconic depth estimation block diagram

algorithm is used. The resulting error surface $e(d; x, y)$ is approximately parabolic around the minimum. The flow value with the lowest error is taken as the estimator output $d(x, y)$ and the second derivative of the error surface is used to compute its variance.

The implementation of the SSD algorithm is particularly simple for flow parallel to the image raster. Each scanline of the two image frames is first magnified by a factor of 4 by cubic interpolation. The images are then shifted using sub-pixel displacements d_k and the SSD measure e_k is computed using a 5×5-pixel square window. The minimum error $(d_{\hat{k}}, e_{\hat{k}})$ is found and a parabola

$$e(d) = ad^2 + bd + c$$

is fit to this point and its two neighbors $(d_{\hat{k}-1}, e_{\hat{k}-1})$ and $(d_{\hat{k}+1}, e_{\hat{k}+1})$. The minimum of this parabola establishes the flow estimate (to sub-sub-pixel precision). The variance of the flow measurement can be shown to be

$$\mathrm{Var}(d) = 2\sigma^2/a,$$

where σ^2 is the variance of the image noise process [13]. Adjacent flow estimates are correlated over both space and time [13]; the significance of this fact will be considered in Section 5.1.

The raw flow and variance estimates are scaled to units of inverse depth using knowledge of the camera motion and the calibration parameters of a pin-hole camera model. This facilitates the integration of information when the camera motion is not linear, e.g. for widening baseline stereo [23] or for orthogonal camera motions (Section 5.2).

3.2 Update (integration)

The next stage in the iconic depth estimator is the integration of the new disparity measurements with the predicted disparity map. For now, we will assume that each value in the measured or predicted disparity map is not correlated with its neighbors, so that the map updating can be done at each pixel independently.

To update a pixel value, we first compute the variance of the updated disparity estimate

$$p_t^+ = ((p_t^-)^{-1} + (\sigma_d^2)^{-1})^{-1} = \frac{p_t^- \sigma_d^2}{p_t^- + \sigma_d^2}$$

and the Kalman filter gain K

$$K = \frac{p_t^+}{\sigma_d^2} = \frac{p_t^-}{p_t^- + \sigma_m^2}.$$

We then update the disparity value by using the Kalman filter update equation

$$u_t^+ = u_t^- + K(d - u_t^-)$$

where u_t^- and u_t^+ are the predicted and updated disparity estimates and d is the new disparity measurement. This update equation can also be written as

$$u_t^+ = p_t^+ \left(\frac{u_t^-}{p_t^-} + \frac{d}{\sigma_d^2} \right).$$

The latter form shows that the updated disparity estimate is a linear combination of the predicted and measured values, inversely weighted by their respective variances.

3.3 Smoothing (regularization)

The raw depth or disparity values obtained from optical flow measurements can be very noisy, especially in areas of uniform intensity. We employ smoothness criteria to reduce the noise and to "fill in" underconstrained areas. Such methods have been discussed in [1], [9], [15], [16], and [21]. For our application, smoothing is done on the disparity field, using the inverse variance of the disparity estimate as the confidence in each measurement. The smoother we use is the generalized piecewise continuous spline under tension [22], which uses finite element relaxation to compute the smoothed field. The algorithm is implemented with a three-level coarse-to-fine strategy to speed convergence.

The smoothing stage can be viewed as the part of the Kalman filtering algorithm that incorporates prior knowledge

about the smoothness of the disparity map. As shown in [19], a regularization-based smoother is equivalent to a prior model with a correlation function defined by the degree of the stabilizing spline (e.g. membrane or thin plate). The resulting prior covariance matrix contains off-diagonal elements modeling the covariance of neighboring pixels. An optimal implementation of the Kalman filter would require carrying this entire covariance matrix, with non-zero correlations between the depths at neighboring pixels, through the update and prediction stages. This would significantly complicate our algorithm. Our approach of explicitly modeling only the variance at each pixel, with covariance information implicitly modeled in a fixed regularization stage, has worked well in practice.

3.4 Prediction (warping and resampling)

The prediction stage of the Kalman filter must predict both the depth and the depth uncertainty for each pixel in the next image. We will describe the disparity extrapolation first, then consider the uncertainty extrapolation.

Translating the camera laterally shifts each point $\mathbf{x}_t = (x, y)$ in the current image to the point $\mathbf{x}_{t+1} = (x_t + T_x u_t, y)$ in the next image, where T_x is a constant depending on the amount of camera motion and the focal length. This shift can be viewed as warping the disparity map. In general, this warping process will yield estimates of disparity in between pixels in the new image, so we need to resample to obtain predicted disparity at pixel locations. For a given pixel $\mathbf{x}'$ in the new image, we find the pair of extrapolated pixels that overlap $\mathbf{x}'$ and compute the disparity at $\mathbf{x}'$ by linear interpolation.

Uncertainty will increase in the prediction phase due to errors from many sources, including uncertainty in the camera motion, errors in calibration, and inaccurate models of the camera optics. A simple approach to modeling these errors is to lump them together by inflating the current variance estimates by a small multiplicative factor in the prediction stage,

$$p_{t+1}^- = (1 + \epsilon) p_t^+. \tag{1}$$

In the Kalman filtering literature this is known as exponential age-weighting of measurements [14], because it decreases the weight given to previous measurements by an exponential function of time. We use this approach in our implementation. We first inflate the variance in the current disparity map using equation (1), then warp and interpolate the variance map in the same way as the disparity map.

4 Feature-based depth estimation

The dense, iconic depth estimation algorithm described in the previous section can be compared with existing depth estimation methods based on sparse feature tracking [2] [5] [7] [11]. For lateral camera motion, the position of a feature on a scanline is a linear function of the distance moved by the camera, since

$$\Delta x = T_x d_0 \quad \Leftrightarrow \quad x_t = x_0 + t\, T_x\, d_0$$

where x_0 is the position of the feature in the first frame and d_0 is the inverse depth of the feature. The *epipolar plane image* method [4] exploits these characteristics by extracting lines in "space-time" (epipolar plane) images formed by concatenating scanlines from an entire image sequence. However, sequential estimation techniques like Kalman filtering are a more practical approach to this problem because they allow images to be processing on-line by incrementally refining the depth model [3] [11].

The state vector for this approach contains the current image position x_t and depth estimate d_t for each feature. Assuming that the camera motion is exact and that measured feature positions have normally distributed uncertainty with variance σ_e^2, the initial state vector and covariance matrix are expressed in terms of image coordinates as

$$\begin{aligned} x_1 &= \tilde{x}_1 \\ d_1 &= \frac{\tilde{x}_1 - \tilde{x}_0}{T_1} \\ P_1^+ &= \sigma_e^2 \begin{bmatrix} 1 & 1/T_1 \\ 1/T_1 & 2/T_1^2 \end{bmatrix} \end{aligned}$$

where T_1 is the camera translation between the first and second frame. The covariance matrix comes from applying standard linear error propagation methods to the equations for x_1 and d_1 [14].

After initialization, if T_t is the translation between frames $t-1$ and t, the motion equations that transform the state vector and covariance matrix to the current frame are

$$u_t^- = \begin{bmatrix} x_t^- \\ d_t^- \end{bmatrix} = \begin{bmatrix} 1 & T_t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{t-1}^+ \\ d_{t-1}^+ \end{bmatrix} = \Phi_t u_{t-1}^+ \tag{2}$$

$$P_t^- = \Phi_t P_{t-1}^+ \Phi_t^T. \tag{3}$$

The superscript minuses indicate that these estimates do not incorporate the measured edge position at time t. The newly measured edge position $\tilde{x}_t$ is incorporated by computing the updated covariance matrix P_t^+, a gain matrix K, and the updated parameter vector u_t^+:

$$\begin{aligned} P_t^+ &= \{(P_t^-)^{-1} + S\}^{-1} \quad \text{where} \quad S = \frac{1}{\sigma_e^2} \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \\ K &= \frac{1}{\sigma_e^2} P_t^+ \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ u_t^+ &= u_t^- + K[\tilde{x}_t - x_t^-]. \end{aligned}$$

Since these equations are linear, we can see how uncertainty decreases as the number of measurements increases by computing the sequence of covariance matrices P_t, given only the measurement uncertainty σ_e^2 and the sequence of camera motions T_t. This is addressed in Section 5.1.

5 Evaluation

In this section, we compare the performance of the iconic and feature-based depth estimation algorithms in three ways. First, we perform a mathematical analysis of the reduction in depth variance as a function of time. Second, we use a sequence of images of a flat scene to determine the quantitative performance of the two approaches and to check the validity of our analysis. Third, we test our algorithms on images of a realistic scene with complicated variations in depth.

5.1 Mathematical analysis

We wish to compare the theoretical variance of the depth estimates obtained by the iconic method of Section 3 to those obtained by the feature-based method of Section 4. We will also compare the accuracy of both methods to the accuracy of stereo matching with the first and last frames of the image sequence. To do this, we will derive expressions for the depth variance as a function of the number of frames processed, assuming a constant noise level in the images and constant camera motion between frames. For clarity, we will assume that $T_x = 1$.

Iconic approach

For the iconic method, we will ignore process noise in the system model and assume that the variance of successive flow measurements is constant. For lateral motion, the equations developed in Section 2 can be simplified to show that the Kalman filter simply computes the average flow [20]. Therefore, a sequence of flow measurements $\Delta x_1, \Delta x_2, \ldots, \Delta x_t$ is equivalent to the following batch measurement equation

$$\Delta \mathbf{x} = \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \\ \cdot \\ \cdot \\ \cdot \\ \Delta x_t \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ \cdot \\ \cdot \\ \cdot \\ 1 \end{bmatrix} d = \mathbf{H}d.$$

Estimating d by averaging the flow measurements implies that

$$d = \frac{1}{t} H^T \Delta \mathbf{x} = \frac{1}{t} \sum_{i=1}^{t} \Delta x_i. \tag{4}$$

If the flow measurements were independent with variance $2\sigma_n^2/a$, where σ_n is the noise level in the image [13], the resulting variance of the disparity estimate would be

$$\frac{2\sigma_n^2}{ta}. \tag{5}$$

However, the flow measurements are not actually independent. Because noise is present in every image, flow measurements between frames $i-1$ and i will be correlated with measurements for frames i and $i+1$. It can be shown [13] that a sequence of correlation-based flow measurements that track the same point in the image sequence will have the following covariance matrix:

$$P_m = \frac{\sigma_n^2}{a} \begin{bmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & \cdot & & & \\ & & & \cdot & & \\ & & & & \cdot & \\ & & & & 2 & -1 \\ & & & & -1 & 2 \end{bmatrix}$$

where σ^2 is the level of noise in the image and a reflects the local slope of the intensity surface. With this covariance matrix, averaging the flow measurements actually yields the following variance for the estimated flow:

$$\sigma_I^2(t) = \frac{1}{t^2} H^T P_m H = \frac{2\sigma^2}{t^2 a}. \tag{6}$$

This is interesting and rather surprising. Comparing equations (5) and (6), the correlation structure that exists in the measurements means that the algorithm converges faster than we first expected.

With correlated measurements, averaging the flow measurements in fact is a sub-optimal estimator for d. The optimal estimator is obtained by substituting the expressions for H and P_m into the batch solution equations [14]

$$d = (H^T P_m^{-1} H)^{-1} H^T P_m^{-1} \Delta \mathbf{x} \tag{7}$$

and

$$\sigma_d^2 = (H^T P_m^{-1} H)^{-1}. \tag{8}$$

This estimator does not give equal weight to all flow measurements; instead, measurements near the center of the sequence receive more weight than those near the end. The variance of the depth estimate is

$$\sigma_I^2(t) = \frac{12\sigma_n^2}{t(t+1)(t+2)a}.$$

The optimal convergence is cubic, whereas the convergence of the averaging method we implemented is quadratic. Developing an incremental version of the optimal estimator requires extending our Kalman filter formulation to model the correlated nature of the measurements. This extension is currently being investigated.

Feature-based approach

For the feature-based approach, the desired variance estimates come from computing the sequence of covariance matrices P_t, as mentioned at the end of Section 4. A closed form expression for this matrix is easier to obtain from a batch line fit solution for x_0 and d_0 than from the Kalman filter formulation and yields an equivalent result. Since we assume that the measured edge positions $\tilde{x}_i$ are independent with equal variance σ_e^2, we find that

$$P_F = \begin{bmatrix} \sigma_x^2 & \sigma_{xd} \\ \sigma_{xd} & \sigma_d^2 \end{bmatrix} = \sigma_e^2 \begin{bmatrix} \sum_{i=0}^{t} 1 & \sum_{i=0}^{t} i \\ \sum_{i=0}^{t} i & \sum_{i=0}^{t} i^2 \end{bmatrix}^{-1}. \tag{9}$$

The summations can be expressed in closed form, leading to the conclusion that

$$\sigma_F^2(t) = \frac{12\sigma_e^2}{t(t+1)(t+2)}. \tag{10}$$

The variance of the displacement or flow estimate d_0 thus decreases as the cube of the number of images. This expression is identical in structure to the optimal estimate for the iconic approach, the only difference being the replacement of the variance of the SSD minimum by the variance of the edge position. Thus, if our estimators incorporate appropriate models of measurement noise, the iconic and feature-based methods theoretically achieve the same rate of convergence. This is surprising, given that the basic Kalman filter for the iconic method maintains only one state parameter (d) for each pixel, whereas the feature-based method maintains two per feature (x_0 and d_0). We suspect that an incremental version of the optimal iconic estimator will require the same amount of state as the feature-based method.

Comparison with stereo

To compare these methods to stereo matching on the first and last frames of the image sequence, we must scale the stereo disparity and its uncertainty to be commensurate with the flow between frames. This implies dividing the stereo disparity by t and its uncertainty by t^2. For the iconic method, we assume that the uncertainty in a stereo measurement will be the same as that for an individual flow measurement. Thus, the scaled uncertainty is

$$\sigma_{IS}^2(t) = \frac{2\sigma_n^2}{t^2 a}.$$

This is the same as is achieved by our incremental algorithm that processes all of the intermediate frames. Therefore, processing the intermediate frames as we do (that is, ignoring the temporal correlation of the measurements) may improve the reliability of the matching, but in this case it does not improve precision.

For the feature-based approach, the uncertainty in stereo disparity is twice the uncertainty σ_e^2 in the feature position; the scaled uncertainty is therefore

$$\sigma_{FS}^2(t) = \frac{2\sigma_e^2}{t^2}.$$

In this case using the intermediate frames helps, since

$$\frac{\sigma_F(t)}{\sigma_{FS}(t)} = \frac{1}{O(\sqrt{t})}.$$

Thus, extracting depth from a small-motion image sequence has several advantages over stereo matching between the first and last frames. The ease of matching is increased, reducing the number of correspondence errors. Occlusion is less of a problem, since it can be predicted from early measurements. Finally, better accuracy is available by using the feature-based method or the optimal version of the iconic method.

5.2 Quantitative experiments: flat scenes

The goals of our quantitative evaluation were to examine the actual convergence rates of the depth estimators, to assess the validity of the noise models, and to compare the performance of the iconic and feature-based algorithms. To obtain ground truth depth data, we used the facilities of the Calibrated Imaging Lab at CMU to digitize a sequence of images of a flat-mounted poster. We used a Sony XC-37 CCD camera with a 16mm lens, which gave a field of view of 36 degrees. The poster was set about 20 inches from the camera. The camera motion between frames was 0.04 inches, which gave an actual flow of approximately two pixels per frame in 480x512 images. For convenience, our experiments were run on images reduced to 240x256 by Gaussian convolution and subsampling. The image sequence we will discuss here was taken with vertical camera motion. This proved to give somewhat better results than horizontal motion; we attribute this to jitter in the scanline clock, which induces more noise in horizontal flow than in vertical flow.

Figure 3 shows the poster and the edges extracted from it. For both the iconic and the feature-based algorithms, a ground truth value for the depth was determined by fitting a plane to the measured values. The level of measurement noise was then estimated by computing the RMS deviation of the measurements from the plane fit. Optical aberrations made the flow measurements consistently smaller near the periphery of the image than the center, so the RMS calculation was performed over only the center quarter of the image. Note that all experiments described in this section did *not* use regularization to smooth the depth estimates, so the results show only the effect of the Kalman filtering algorithm.

To examine the convergence of the Kalman filter, the RMS depth error was computed for the iconic and the feature-based algorithms after processing each image in the sequence. We computed two sets of statistics, one for "sparse" depth and one for "dense" depth. The sparse statistic computes the RMS error for only those pixels where both algorithms gave depth estimates (that is, where edges were found), whereas the dense statistic computes the RMS error of the iconic algorithm over the full image. Figure 4 plots the relative RMS errors as a function of the number of images processed. Comparing the sparse error curves, the convergence rate of the iconic algorithm is slower than the feature-based algorithm, as expected. The relative heights of the two curves will depend on the relative sizes and shapes of the correlation window and the edge operator. In this particular experiment, both methods converged to an error level of approximately 0.5% percent after processing eleven images. Since the poster was 20 inches from the camera, this equates to a depth error of 0.1 inches. Note that the overall baseline between the first and the eleventh image was only 0.44 inches.

To compare the theoretical convergence rates derived earlier to the experimental rates, the theoretical curves were scaled to coincide with the experimental error after processing the first two frames. These scaled curves are also shown in Figure 4. For the iconic method, the theoretical rate plotted is the quadratic convergence predicted by the correlated flow measurement model. The agreement between theory and practice is quite good for the first three frames. Thereafter, the experimental RMS error decreases more slowly; this is probably due to the effects of unmodeled sources of noise. For the feature-based method, the experimental error initially decreases faster than predicted because the implementation required new edge matches to be consistent with the prior depth estimate. When this requirement was dropped, the results agreed very closely with the expected convergence rate. Finally, Figure 4 also compares the RMS error for the sparse and dense depth estimates from the iconic method. The dense flow field is considerably noisier than the flow estimates that coincide with edges, though still just over two percent error by the end of eleven frames. Thus, the iconic method also provides valuable depth information at pixels not containing sharp edges.

5.3 Qualitative experiments: real scenes

We have tested the iconic and feature-based algorithms on complicated, realistic scenes obtained from the Calibrated Imaging Laboratory. Two sequences of ten images were taken with camera motion of 0.05 inches between frames; one se-

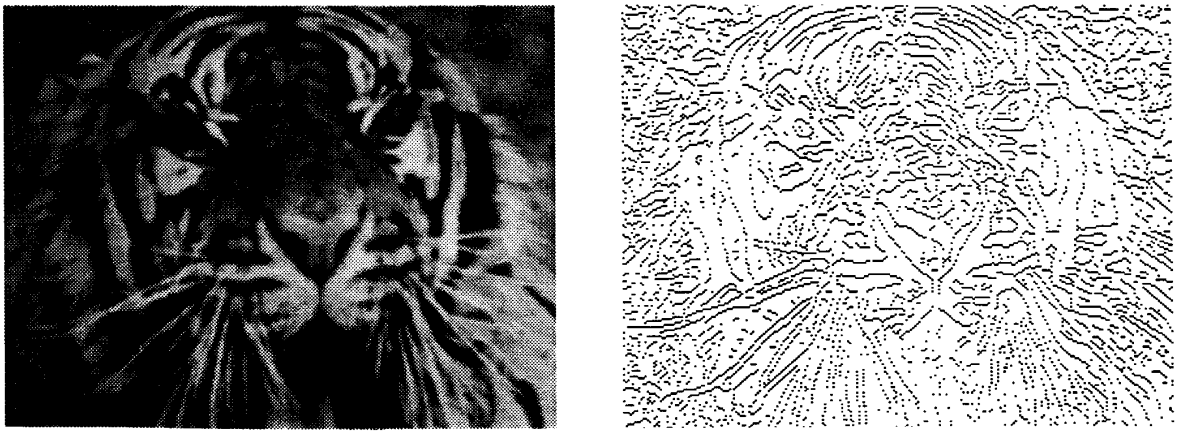


Figure 3: Tiger image and edges

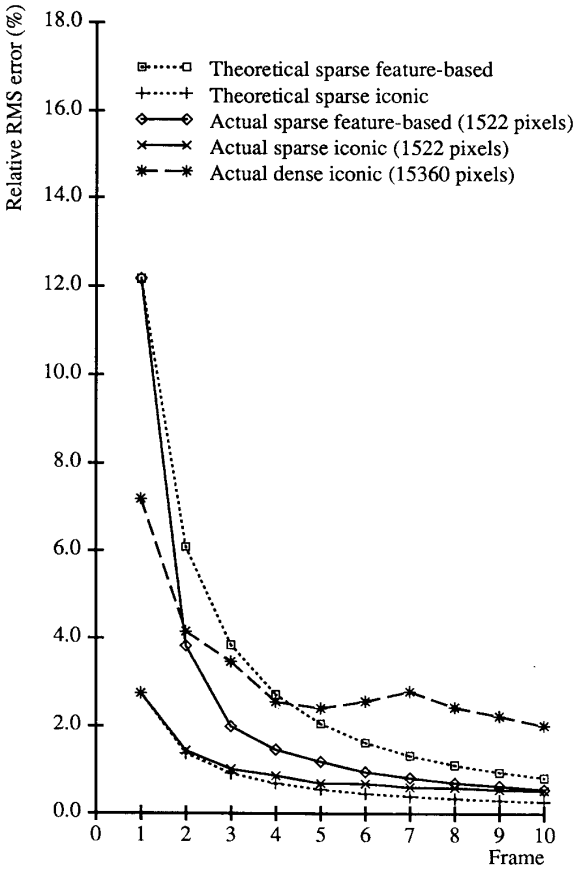


Figure 4: RMS error in depth estimate

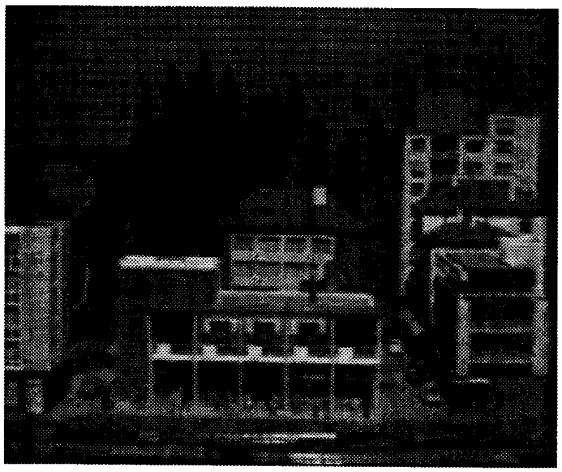
quence moved the camera vertically, the other horizontally. The overall range of motion was therefore 0.5 inches; this compares with distances to objects in the scene of 20 to 40 inches.

One of the images is shown in Figure 5a. Figure 5b shows a depth map that combines the results of the iconic method applied to both the horizontal and the vertical image sequences. Lighter areas in the depth map are nearer. The combined depth map reveals more scene structure than either the horizontal or the vertical sequence would alone. Figures 5c and 5d show 3-D perspective reconstructions obtained from the iconic and the feature-based methods, respectively. These were obtained from depth maps that combined disparity estimates obtained from horizontal and vertical motion, as in Figure 5b. The depth map for the feature-based approach was produced from the sparse depth estimates by regularization. It is difficult to make quantitative statements about the performance of either method from this data, but qualitatively it is clear that both recover the structure of the scene quite well.

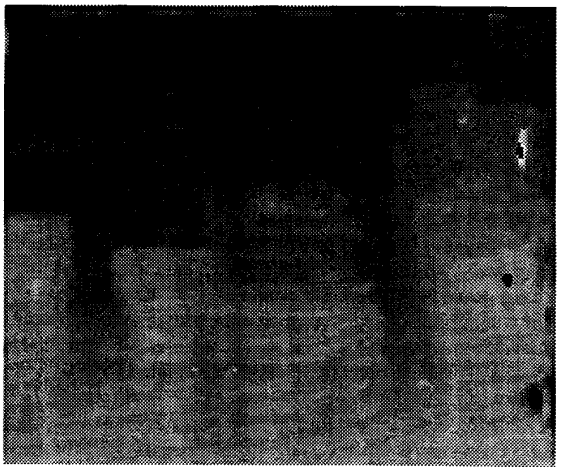
6 Conclusions

This paper has presented a new algorithm for extracting depth from known motion. The algorithm processes an image sequence taken with small inter-frame displacements and produces an on-line estimate of depth that is refined over time. The algorithm produces a dense, iconic depth map and is suitable for implementation on parallel architectures.

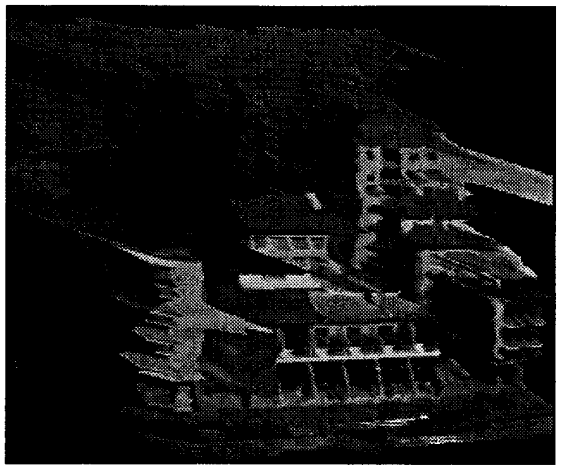
The on-line depth estimator is based on Kalman filtering. A correlation-based flow algorithm measures both the local displacement at each pixel and the confidence (or variance) of the displacement. These two "measurement images" are integrated with predicted depth and variance maps using a weighted least squares technique derived from the Kalman filter. Regularization-based smoothing is used to reduce the noise in the flow estimates and to fill in areas of unknown disparity. The current maps are extrapolated to the next frame by image warping, using the knowledge of the camera motion, and are resampled to keep the maps iconic.



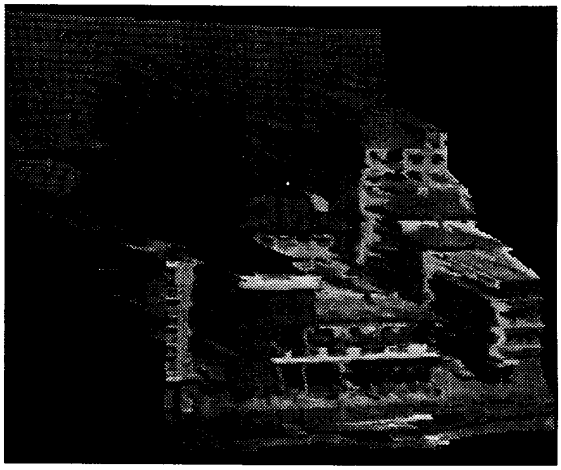
(a)



(b)



(c)



(d)

Figure 5: CIL depth maps

(a) first frame (b) depth map for iconic method

(c) perspective view for iconic method (d) perspective view for symbolic method

The algorithm has been implemented, evaluated mathematically and experimentally, and compared with a feature-based algorithm that uses Kalman filtering to estimate the depth of edges. The mathematical analysis shows that the iconic approach will have a slower convergence rate because it only keeps one element of state per pixel (the disparity), while the feature-based approach keeps both the disparity and the sub-pixel position of the feature. However, an optimal implementation of the iconic method (which takes into account temporal correlations in the measurements) has the potential to equal the convergence rate and accuracy of the symbolic method. Experiments with images of a flat poster have confirmed this analysis and given quantitative measures of the performance of both algorithms. Finally, experiments with images of a realistic outdoor scene model have shown that the new algorithm performs well on images with large variations in depth and intensity.

The algorithms described in this paper can be extended in several ways. The most straightforward extension is to the case of non-lateral motion [13]. This can be accomplished by designing a correlation-based flow estimator that produces two-dimensional flow vectors and an associated covariance matrix estimate [1]. This approach can also be used when the camera motion is uncertain or when the camera motion is variable (e.g. for widening baseline stereo [23]).

More research is required into the behavior of the correlation based flow and confidence estimator. In particular, we have observed that our current estimator produces biased estimates in the vicinity of intensity step edges. The correlation between spatially adjacent flow estimates, which is currently ignored, should be integrated into the Kalman filter framework. More sophisticated representations for the intensity and depth fields are also being investigated [18].

Finally, the incremental depth from motion algorithms we have developed can be used to initiate stereo fusion. Work is currently in progress investigating the integration of depth-from-motion and stereo [10]. We believe that the framework presented in this paper will prove to be useful for integrating information from multiple visual sources and for tracking such information in a dynamic environment.

References

[1] P. Anandan. Computing dense displacement fields with confidence measures in scenes containing occlusion. In *IUS Workshop*, pages 236–246, DARPA, December 1985.

[2] N. Ayache and O. D. Faugeras. Maintaining representations of the environment of a mobile robot. In *International Symposium of Robotics Research 4*, MIT Press, 1987.

[3] H. H. Baker. Multiple-image computer vision. In *Proceedings of the 41st Photogrammetric Week*, pages 7–19, Stuttgart Institute for Photogrammetry, Stuttgart, West Germany, September 1987.

[4] R. C. Bolles, H. H. Baker, and D. H. Marimont. Epipolar-plane image analysis: an approach to determining structure from motion. *International Journal of Computer Vision*, 1:7–55, 1987.

[5] T. J. Broida and R. Chellappa. Kinematics and structure of a rigid object from a sequence of noisy images. In *Proc. Workshop on Motion: Representation and Analysis*, pages 95–100, IEEE, May 1986.

[6] O. D. Faugeras, N. Ayache, B. Faverjon, and F. Lustman. Building visual maps by combining noisy stereo measurements. In *IEEE International Conference on Robotics and Automation*, pages 1433–1438, IEEE, San Francisco, California, April 1986.

[7] J. Hallam. Resolving observer motion by object tracking. In *International Joint Conference on Artificial Intelligence*, 1983.

[8] D. J. Heeger. Optical flow from spatiotemporal filters. In *First International Conference on Computer Vision*, pages 181–190, IEEE Computer Society Press, June 1987.

[9] B. K. P. Horn and B. G. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[10] L. H. Matthies. *Motion and Depth from Stereo Image Sequences*. PhD thesis, Carnegie Mellon University, (in preparation) 1988.

[11] L. H. Matthies and T. Kanade. The cycle of uncertainty and constraint in robot perception. In *Proc. International Symposium on Robotics Research*, MIT Press, August 1987.

[12] L. H. Matthies and S. A. Shafer. Error modeling in stereo navigation. *IEEE Journal of Robotics and Automation*, 239–248, June 1987.

[13] L. H. Matthies, R. Szeliski, and T. Kanade. *Kalman Filter-based Algorithms for Estimating Depth from Image Sequences*. Technical Report CMU-CS-87-185, Computer Science Department, Carnegie Mellon University, December 1987.

[14] P. S. Maybeck. *Stochastic Models, Estimation, and Control*. Volume 1, Academic Press, New York, NY, 1979.

[15] H.-H. Nagel and W. Enkelmann. An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(5):565–593, September 1986.

[16] T. Poggio, V. Torre, and C. Koch. Computational vision and regularization theory. *Nature*, 317(6035):314–319, 26 September 1985.

[17] P. Rives, E. Breuil, and B. Espiau. Recursive estimation of 3d features using optical flow and camera motion. In *Proceedings Conference on Intelligent Autonomous Systems*, pages 522–532, Elsevier Science Publishers, December 1986. (also appeared in Proc. 1987 IEEE Int'l Conf. on Robotics and Automation).

[18] R. Szeliski. *Bayesian Modeling of Uncertainty in Low Level Vision*. PhD thesis, Carnegie Mellon University, (in preparation) 1988.

[19] R. Szeliski. Regularization uses fractal priors. In *Proceedings AAAI-87*, pages 749–754, Morgan Kaufmann Publishers, Seattle, Washington, July 1987.

[20] Technical Staff, The Analytic Sciences Corporation. *Applied Optimal Estimation*. MIT Press, Cambridge, MA, 1974.

[21] D. Terzopoulos. Image analysis using multigrid relaxation methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(2):129–139, March 1986.

[22] D. Terzopoulos. Regularization of inverse visual problems involving discontinuities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(4):413–424, July 1986.

[23] G. Xu, S. Tsuji, and A. Minoru. Coarse-to-fine control strategy for matching motion stereo pairs. In *Proceedings of IJCAI*, pages 892–894, 1985.