Fast 3-D Interpretation from Monocular Image Sequences on Large Motion Fields

Jong-Sung Kim and Ki-Sang Hong

Division of Electrical and Computer Engineering, POSTECH, Pohang, Korea
{kimjs,hongks}@postech.ac.kr
http://iip.postech.ac.kr

Abstract. This paper proposes a fast method for dense 3-D interpretation to directly estimate a dense map of relative depth and motion from a monocular sequence of images on large motion fields. The Nagel-Enkelmann technique is employed in the variational formulation of the problem. Diffusion-reaction equations are derived from the formulation so as to approximate the dense map on large motion fields and realize an anisotropic diffusion to preserve the discontinuities of the map. By combining the ideas of implicit schemes and multigrid methods, we present a new implicit multigrid block Gauss-Seidel relaxation scheme, which dramatically reduces the computation time for solving the large-scale linear system of diffusion-reaction equations. Using our method, we perform fast 3-D interpretation of image sequences with large motion fields. The efficiency and effectiveness of our method are experimentally verified with synthetic and real image sequences.

1 Introduction

The 3-D interpretation is one of fundamental problems in computer vision. Reconstructed 3-D information such as scene depth and motion can be used in vision applications. Most methods for 3-D interpretation first compute sparse or dense correspondences such as image features or optical flow. They depend on the accuracy of generally ill-posed matching or estimation process [3,7]. As an alternative, direct methods [2,11,12] have been introduced to estimate the structure and motion of a static scene without prior matching and estimation. However, the requirement of static scenes limits the applicability of the direct methods in a variety of environments, e.g., where viewed objects move independently of viewing systems. This limitation has been first studied as a dense 3-D interpretation problem [8,10], where a dense map of relative depth and motion is estimated from the spatio-temporal change of intensity images due to small-range image motion. It has been formulated within the variational framework and partial differential equation (PDE) systems were solved to obtain the functional minimization. In this framework, each pixel has six variables representing 3-D motion. Thus, there are more than two million variables to evaluate with a 640×480 video image. For this reason, the computation cost for dense 3-D

Y. Yagi et al. (Eds.): ACCV 2007, Part II, LNCS 4844, pp. 497–506, 2007.

interpretation has been too expensive for time-critical applications, even though it has been applied to small motion fields.

In this paper we present a fast method for dense 3-D interpretation. Our method performs fast 3-D interpretation on large motion fields. It was developed in the following way: First, dense 3-D interpretation was formulated with a variational PDE model based on the instantaneous motion model [6] and the constant brightness assumption. The Nagel-Enkelman technique [9] was employed in the variational model to perform our method on large motion fields. This model is used to derive diffusion-reaction equations, which approximate the dense map between two images and realize an anisotropic diffusion to preserve discontinuities in the map. Then, by combining implicit schemes [1,13] and multigrid methods [4,5], a new implicit multigrid block Gauss-Seidel relaxation scheme is devised to quickly solve the diffusion-reaction equations. The efficiency and effectiveness of our method is verified with synthetic and real image sequences.

This paper is organized as follows. Section 2 introduces the variational PDE model for dense 3-D interpretation. Section 3 describes the discretization for numerically solving the variational PDE model. Section 4 explains our numerical scheme based on implicit schemes and multigrid methods. Sections 5 and 6 present experimental results and conclusions.

2 Variational PDE Model

A 3-D point $\boldsymbol{X} = (X, Y, Z)^\top$ on a moving object with linear velocity $\boldsymbol{v} = (v_1, v_2, v_3)^\top$ and angular velocity $\boldsymbol{w} = (w_1, w_2, w_3)^\top$ satisfies the differential equation $\dot{\boldsymbol{X}} = \boldsymbol{v} + [\boldsymbol{w}]_\times \boldsymbol{X}$ and its projection $\boldsymbol{x}_c = (x, y, f)^\top$ through a calibrated camera (focal length f) has the form $\boldsymbol{x}_c = f\,(X/Z, Y/Z, 1)^\top$. The 2-D coordinate vector $(x, y)^\top$ is denoted by $\boldsymbol{x}$ to simplify the notation. Under the instantaneous motion model [6], the optical flow $\boldsymbol{u} = (u, v)^\top$ can be written as

$$\boldsymbol{u} = P\boldsymbol{t} + Q\boldsymbol{w}, \tag{1}$$

where $\boldsymbol{t} = (t_1, t_2, t_3)^\top$ is $Z^{-1}\boldsymbol{v}$ and the matrices P and Q are defined by

$$P = \begin{pmatrix} f & 0 & -x \\ 0 & f & -y \end{pmatrix}, \quad Q = \frac{1}{f}\begin{pmatrix} -xy & f^2 + x^2 & -fy \\ -f^2 + y^2 & xy & fx \end{pmatrix}, \tag{2}$$

respectively. As we see in Eq. (1), only the direction of translational motion and the depth of the scene up to a scaling factor [6,8,10] can be reconstructed by $\hat{\boldsymbol{v}} = ||\boldsymbol{t}||^{-1}\boldsymbol{t}$ and $\hat{Z^{-1}} = ||\boldsymbol{t}||$, where $||\cdot||$ denotes the l_2-norm of a vector.

We perform dense 3-D interpretation between two images $\mathcal{I}_1(\boldsymbol{x}) \equiv \mathcal{I}_1(x, y)$ and $\mathcal{I}_2(\boldsymbol{x}) \equiv \mathcal{I}_2(x, y)$. Just as in optical flow, the Nagel-Enkelman model is applied to dense 3-D interpretation between two frames as follows:

$$\begin{aligned} E(\boldsymbol{t}, \boldsymbol{w}) = \int_\Omega \Big[& \big(\mathcal{I}_1(\boldsymbol{x}) - \mathcal{I}_2(\boldsymbol{x} + P\boldsymbol{t} + Q\boldsymbol{w})\big)^2 \\ & + \alpha \sum_i^3 \left(\nabla t_i^\top D(\nabla \mathcal{I}_1)\, \nabla t_i\right) + \beta \sum_i^3 \left(\nabla w_i^\top D(\nabla \mathcal{I}_1)\, \nabla w_i\right) \Big] d\boldsymbol{x}, \end{aligned} \tag{3}$$

where $D(\nabla\mathcal{I}_1)$ is the regularized projection matrix, defined by

$$D(\nabla\mathcal{I}_1) = \frac{1}{||\nabla\mathcal{I}_1||^2 + 2\nu^2}\left(\begin{pmatrix} -\partial\mathcal{I}_1/\partial y \\ \partial\mathcal{I}_1/\partial x \end{pmatrix}\begin{pmatrix} -\partial\mathcal{I}_1/\partial y \\ \partial\mathcal{I}_1/\partial x \end{pmatrix}^\top + \nu^2 I\right), \quad (4)$$

where I is an identity matrix, ν a contrast parameter which is set to inhibit smoothing of the flow field across the edges of $\mathcal{I}_1$ at locations where $||\nabla\mathcal{I}_1|| \gg \nu$, and α and β smoothing parameters. A piecewise smooth flow field is estimated by using that projection matrix in the regularization. We will describe the details for estimating the translational motion $\boldsymbol{t}$, but our model can be applied to the simultaneous estimation of the translational motion $\boldsymbol{t}$ and rotational motion $\boldsymbol{w}$ without any modification.

We seek $\boldsymbol{t}$ minimizing the energy functional of Eq. (3) by solving a solution of the diffusion-reaction equations derived through the calculus of variations. The diffusion-reaction equations are given by the following PDE system

$$\begin{cases} \alpha\mathrm{div}\left(D(\nabla\mathcal{I}_1)\nabla t_1\right) + \left(\mathcal{I}_1(\boldsymbol{x}) - \mathcal{I}_2(\boldsymbol{x} + P\boldsymbol{t} + Q\boldsymbol{w})\right)\frac{\partial\mathcal{I}_2}{\partial t_1} = 0 \\ \alpha\mathrm{div}\left(D(\nabla\mathcal{I}_1)\nabla t_2\right) + \left(\mathcal{I}_1(\boldsymbol{x}) - \mathcal{I}_2(\boldsymbol{x} + P\boldsymbol{t} + Q\boldsymbol{w})\right)\frac{\partial\mathcal{I}_2}{\partial t_2} = 0 \\ \alpha\mathrm{div}\left(D(\nabla\mathcal{I}_1)\nabla t_3\right) + \left(\mathcal{I}_1(\boldsymbol{x}) - \mathcal{I}_2(\boldsymbol{x} + P\boldsymbol{t} + Q\boldsymbol{w})\right)\frac{\partial\mathcal{I}_2}{\partial t_3} = 0 \end{cases} \quad (5)$$

with boundary conditions $\partial t_1/\partial\boldsymbol{n} = 0$, $\partial t_2/\partial\boldsymbol{n} = 0$, $\partial t_3/\partial\boldsymbol{n} = 0$, where $\boldsymbol{n}$ is the unit vector normal to the boundary $\partial\Omega$ of the image domain Ω. In Eq. (5), the firs terms realize a discontinuity-preserving anisotropic diffusion process and the second terms enforce the constant brightness constraint in the estimation of $\boldsymbol{t}$. The former are called diffusion terms and the latter reaction terms. As we see, the diffusion terms are linear w.r.t. $\boldsymbol{t}$ but the reaction terms are nonlinear. To apply efficient linear methods to this problem, we approximate the nonlinear reaction terms as follows: Given initial estimates $\boldsymbol{t}^0$, current estimates $\boldsymbol{t}$ is $\boldsymbol{t} = \boldsymbol{t}^0 + \Delta\boldsymbol{t}$, where $\Delta\boldsymbol{t}$ is the error between $\boldsymbol{t}$ and $\boldsymbol{t}^0$, i.e., $\Delta\boldsymbol{t} = \boldsymbol{t} - \boldsymbol{t}^0$. Then, our 3-D brightness constraint equations can be written as

$$\mathcal{I}_1(\boldsymbol{x}) - \mathcal{I}_2\big(\boldsymbol{x} + P\boldsymbol{t}^0 + Q\boldsymbol{w}\big) + \boldsymbol{J}\boldsymbol{t}^0 - \boldsymbol{J}\boldsymbol{t} = 0, \quad (6)$$

where $\boldsymbol{J}$ is the Jacobian of $\mathcal{I}_2$ w.r.t. $\boldsymbol{t}$, defined by $\boldsymbol{J} \equiv \partial\mathcal{I}_2/\partial\boldsymbol{t}$. If we assume that $\boldsymbol{t}^0 = 0$ and $\boldsymbol{w} = 0$ in Eq. (6), we can also acquire the 3-D brightness constraint equations for small motion fields,

$$\mathcal{I}_1(\boldsymbol{x}) - \mathcal{I}_2(\boldsymbol{x}) - \boldsymbol{J}\boldsymbol{t} = 0, \quad (7)$$

which have been used in all 3-D interpretation methods [8,10]. Finally, the successively approximated diffusion-reaction equations are given by

$$\begin{cases} \alpha\mathrm{div}\left(D(\nabla\mathcal{I}_1)\nabla t_1\right) + \left(\mathcal{I}_1(\boldsymbol{x}) - \mathcal{I}_2\left(\boldsymbol{x} + P\boldsymbol{t}^0 + Q\boldsymbol{w}\right) + \boldsymbol{J}\boldsymbol{t}^0 - \boldsymbol{J}\boldsymbol{t}\right)J_1 = 0 \\ \alpha\mathrm{div}\left(D(\nabla\mathcal{I}_1)\nabla t_2\right) + \left(\mathcal{I}_1(\boldsymbol{x}) - \mathcal{I}_2\left(\boldsymbol{x} + P\boldsymbol{t}^0 + Q\boldsymbol{w}\right) + \boldsymbol{J}\boldsymbol{t}^0 - \boldsymbol{J}\boldsymbol{t}\right)J_2 = 0 \\ \alpha\mathrm{div}\left(D(\nabla\mathcal{I}_1)\nabla t_3\right) + \left(\mathcal{I}_1(\boldsymbol{x}) - \mathcal{I}_2\left(\boldsymbol{x} + P\boldsymbol{t}^0 + Q\boldsymbol{w}\right) + \boldsymbol{J}\boldsymbol{t}^0 - \boldsymbol{J}\boldsymbol{t}\right)J_3 = 0 \end{cases} \quad (8)$$

which are our diffusion-reaction equations for dense 3-D interpretation. To solve a solution of these equations, final estimates $\boldsymbol{t}$ should be iteratively updated with initial estimates $\boldsymbol{t}^0$ for every pixels. Therefore, the large-scale linear system of Eq. (8) should be solved in each iteration.

3 Discretization

The unknown functions, $t_1(x,y)$, $t_2(x,y)$, and $t_3(x,y)$, are defined on a pixel grid of cell size $h_x \times h_y$. We denote by h the index for the cell size. We denote by t_{1i}, t_{2i}, and t_{3i} the approximation to t_1, t_2, and t_3, respectively, at some pixel i with $i = 1, \ldots, N$. All spatial gradients are approximated by central differences, and the divergences are estimated using the eight neighboring pixels, denoted by j, with $j = 1, \ldots, 8$. We denote by $\mathcal{N}_i$ the set of neighbors of pixel i and d_{mni} the element (m,n) of the projection matrix $D(\nabla \mathcal{I}_{1i})$ for some pixel i. Then, the diffusivity coefficients, c_{ji}, with $j = 1, \ldots, 8$ in some pixel i, can be computed with

$$\begin{array}{l} c_{1i} = \frac{d_{111}+d_{11i}}{2},\ c_{2i} = \frac{d_{112}+d_{11i}}{2},\quad c_{3i} = \frac{d_{223}+d_{22i}}{2},\quad c_{4i} = \frac{d_{224}+d_{22i}}{2} \\ c_{5i} = \frac{d_{125}+d_{12i}}{4},\ c_{6i} = -\frac{d_{126}+d_{12i}}{4},\ c_{7i} = -\frac{d_{127}+d_{12i}}{4},\ c_{8i} = \frac{d_{128}+d_{12i}}{4}. \end{array} \tag{9}$$

We denote by J_{mi} the the element (m) of the Jacobian in some pixel i, defined by $(J_{1i}, J_{2i}, J_{3i}) = (f\partial \mathcal{I}_{1i}/\partial x, f\partial \mathcal{I}_{1i}/\partial y, -fx - fy)$. We describe the finite difference approximation to the diffusion-reaction equations in Eq. (8) by using this discretization.

By assuming that $\boldsymbol{t}^0 = 0$ due to small motion and $\boldsymbol{w} = 0$ in Eq. (8), we obtain the discrete diffusion-reaction equations for small motion, given by the elliptic PDE system

$$\begin{cases} \alpha \sum_{j \in \mathcal{N}_i} c_{ji} \frac{t_{1j} - t_{1i}}{h^2} + J_{1i} e_{si} - J_{1i}^2 t_{1i} - J_{2i} J_{1i} t_{2i} - J_{3i} J_{1i} t_{3i} = 0, \\ \alpha \sum_{j \in \mathcal{N}_i} c_{ji} \frac{t_{2j} - t_{2i}}{h^2} + J_{2i} e_{si} - J_{1i} J_{2i} t_{1i} - J_{2i}^2 t_{2i} - J_{3i} J_{2i} t_{3i} = 0, \\ \alpha \sum_{j \in \mathcal{N}_i} c_{ji} \frac{t_{3j} - t_{3i}}{h^2} + J_{3i} e_{si} - J_{1i} J_{3i} t_{1i} - J_{2i} J_{3i} t_{2i} - J_{3i}^2 t_{3i} = 0, \end{cases} \tag{10}$$

for $i = 1, \ldots, N$, where $e_{si} \equiv \mathcal{I}_1(\boldsymbol{x}_i) - \mathcal{I}_2(\boldsymbol{x}_i)$. This constitutes a large-scale linear system of equations for the $3N$ unknowns t_{1i}, t_{2i}, and t_{3i}.

Large motions are handled by considering the unknowns as the time evolution functions $\boldsymbol{t}(x,y,n)$ with a time variable n. We solve an evolution solution by calculating the asymptotic state $(n \to \infty)$ of Eq. (8). The associated discrete diffusion-reaction equations are given by the parabolic PDE system

$$\begin{cases} \frac{t_{1i} - t_{1i}^0}{\tau} = \alpha \sum_{j \in \mathcal{N}_i} c_{ji} \frac{t_{1j} - t_{1i}}{h^2} + J_{1i} e_{di}\left(\boldsymbol{t}_i^0\right) - J_{1i}^2 t_{1i} - J_{2i} J_{1i} t_{2i} - J_{3i} J_{1i} t_{3i}, \\ \frac{t_{2i} - t_{2i}^0}{\tau} = \alpha \sum_{j \in \mathcal{N}_i} c_{ji} \frac{t_{2j} - t_{2i}}{h^2} + J_{2i} e_{di}\left(\boldsymbol{t}_i^0\right) - J_{1i} J_{2i} t_{1i} - J_{2i}^2 t_{2i} - J_{3i} J_{2i} t_{3i}, \\ \frac{t_{3i} - t_{3i}^0}{\tau} = \alpha \sum_{j \in \mathcal{N}_i} c_{ji} \frac{t_{3j} - t_{3i}}{h^2} + J_{3i} e_{di}\left(\boldsymbol{t}_i^0\right) - J_{1i} J_{3i} t_{1i} - J_{2i} J_{3i} t_{2i} - J_{3i}^2 t_{3i}, \end{cases} \tag{11}$$

for $i = 1, \ldots, N$, where $e_{di}\left(\boldsymbol{t}_i^0\right) \equiv \mathcal{I}_1(\boldsymbol{x}_i) - \mathcal{I}_2\left(\boldsymbol{x}_i + P_i \boldsymbol{t}_i^0 + Q_i \boldsymbol{w}_i\right) - J_i \boldsymbol{t}_i^0$, and τ the size of the time-step. The above system can be solved by iteratively applying the gradient descent method but a small time-step is required at each iteration of the gradient descent method, which drastically increases the computation time. Instead, we present a new efficient numerical scheme for quickly finding an evolution solution to the parabolic PDE system.

4 Efficient Numerical Scheme

The equations in Eq. (10) constitute a linear elliptic PDE system for the $3N$ unknowns t_{1i}, t_{2i}, and t_{3i}, with a sparse matrix structure. Given a linear system $A\boldsymbol{z} = \boldsymbol{b}$ and a decomposition $A = D - L - U$ with diagonal matrix D, lower triangular matrix L, and upper triangular matrix U, then one iteration of the Gauss-Seidel relaxation [4] is $\boldsymbol{z}^{n+1} = (D - L)^{-1}(U\boldsymbol{z}^n + \boldsymbol{b})$, where n is the iteration index of relaxation. The three unknowns of each pixel are coupled with each other. Such a coupling yields to a block Gauss-Seidel relaxation (BGSR). In each iteration, we simultaneously update the three unknowns, t_{1i}, t_{2i}, and t_{3i}, by solving the following 3×3 linear system at pixel i:

$$\boldsymbol{t}_i^{n+1} = Q_{si}^{-1}\boldsymbol{y}_{si}^{n+1/2}, \tag{12}$$

where Q_{si} and $\boldsymbol{y}_{si}^{n+1/2}$ are defined by

$$Q_{si} = \begin{pmatrix} \frac{\alpha}{h^2}\sum_{j\in\mathcal{N}_i} c_{ji} + J_{1i}^2 & J_{2i}J_{1i} & J_{3i}J_{1i} \\ J_{1i}J_{2i} & \frac{\alpha}{h^2}\sum_{j\in\mathcal{N}_i} c_{ji} + J_{2i}^2 & J_{3i}J_{2i} \\ J_{1i}J_{3i} & J_{2i}J_{3i} & \frac{\alpha}{h^2}\sum_{j\in\mathcal{N}_i} c_{ji} + J_{3i}^2 \end{pmatrix} \tag{13}$$

and

$$\boldsymbol{y}_{si}^{n+1/2} = \begin{pmatrix} \frac{\alpha}{h^2}\left(\sum_{j\in\mathcal{N}_i^-} c_{ji}t_{1j}^{n+1} + \sum_{j\in\mathcal{N}_i^+} c_{ji}t_{1j}^{n}\right) + J_{1i}e_{si} \\ \frac{\alpha}{h^2}\left(\sum_{j\in\mathcal{N}_i^-} c_{ji}t_{2j}^{n+1} + \sum_{j\in\mathcal{N}_i^+} c_{ji}t_{2j}^{n}\right) + J_{2i}e_{si} \\ \frac{\alpha}{h^2}\left(\sum_{j\in\mathcal{N}_i^-} c_{ji}t_{3j}^{n+1} + \sum_{j\in\mathcal{N}_i^+} c_{ji}t_{3j}^{n}\right) + J_{3i}e_{si} \end{pmatrix}, \tag{14}$$

where $\mathcal{N}_i^- = \{j \in \mathcal{N}_i | i(j) < i\}$ and $\mathcal{N}_i^+ = \{j \in \mathcal{N}_i | i(j) > i\}$. The $i(j)$ returns the corresponding pixel index of j.

The explicit scheme for solving a solution to the resulting parabolic PDE system in Eq. (11) can be written as $\boldsymbol{z}^{n+1} = \boldsymbol{z}^n + \tau(A\boldsymbol{z}^n - \boldsymbol{b}^n)$, where $\boldsymbol{b}^n$ denotes that this source term should be updated at each iteration n, unlike the case for small motion. This explicit scheme is only stable for small time-steps. The implicit scheme is stable no matter how large the time-step is [1,13]. In the implicit scheme, we obtain a solution at each iteration by solving the linear system $(I - \tau A)\,\boldsymbol{z}^{n+1} = \boldsymbol{z}^n - \tau\boldsymbol{b}^n$. Since the matrix $(I - \tau A)$ has a sparse matrix structure equal to the structure of the matrix A, we can employ BGSR as a basic iterative solver for the implicit scheme. Thus, one iteration reads $\boldsymbol{z}^{n+1} = \left(\frac{1}{\tau}I - D + L\right)^{-1}\left(\frac{1}{\tau}\boldsymbol{z}^n - U\boldsymbol{z}^n - \boldsymbol{b}^n\right)$. In each iteration step of the implicit block Gauss-Seidel relaxation (IBGSR), the three coupled unknowns, t_{1i}, t_{2i}, and t_{3i}, in some pixel i at time n, are simultaneously updated by solving the 3×3 linear system:

$$\boldsymbol{t}_i^{n+1} = Q_{di}^{-1}\boldsymbol{y}_{di}^{n+1/2}, \tag{15}$$

where Q_{di} and $\boldsymbol{y}_{di}^{n+1/2}$ are defined by

$$Q_{di} = \begin{pmatrix} \frac{1}{\tau} + \frac{\alpha}{h^2}\sum_{j\in\mathcal{N}_i} c_{ji} + J_{1i}^2 & J_{2i}J_{1i} & J_{3i}J_{1i} \\ J_{1i}J_{2i} & \frac{1}{\tau} + \frac{\alpha}{h^2}\sum_{j\in\mathcal{N}_i} c_{ji} + J_{2i}^2 & J_{3i}J_{2i} \\ J_{1i}J_{3i} & J_{2i}J_{3i} & \frac{1}{\tau} + \frac{\alpha}{h^2}\sum_{j\in\mathcal{N}_i} c_{ji} + J_{3i}^2 \end{pmatrix} \tag{16}$$

and

$$\boldsymbol{y}_{di}^{n+1/2} = \begin{pmatrix} \frac{1}{\tau}t_{1i}^n + \frac{\alpha}{h^2}\left(\sum_{j\in\mathcal{N}_i^-} c_{ji}t_{1j}^{n+1} + \sum_{j\in\mathcal{N}_i^+} c_{ji}t_{1j}^n\right) + J_{1i}e_{di}(\boldsymbol{t}_i^n) \\ \frac{1}{\tau}t_{2i}^n + \frac{\alpha}{h^2}\left(\sum_{j\in\mathcal{N}_i^-} c_{ji}t_{2j}^{n+1} + \sum_{j\in\mathcal{N}_i^+} c_{ji}t_{2j}^n\right) + J_{2i}e_{di}(\boldsymbol{t}_i^n) \\ \frac{1}{\tau}t_{3i}^n + \frac{\alpha}{h^2}\left(\sum_{j\in\mathcal{N}_i^-} c_{ji}t_{3j}^{n+1} + \sum_{j\in\mathcal{N}_i^+} c_{ji}t_{3j}^n\right) + J_{3i}e_{di}(\boldsymbol{t}_i^n) \end{pmatrix}. \quad (17)$$

We can obtain two different solutions to our model by iteratively solving the 3×3 linear system of Eqs. (12) or (15) in each pixel.

The convergence of relaxation is improved by employing the multigrid methods [5]. Let us denote the linear systems described above as $A^h\boldsymbol{z}^h = \boldsymbol{b}^h$. We denote by $\tilde{\boldsymbol{z}}$ the current approximate solution $\boldsymbol{z}$, and $\bar{\boldsymbol{z}}$ the smoothed solution obtained by BGSR or IBGSR. Let us denote by H the new cell size $H_x \times H_y$ on the coarse grid. The prolongation for coarse-to-fine grid transfer is represented by $P^{H\to h}$, and the restriction for fine-to-coarse grid transfer by $R^{h\to H}$. The procedure of the two-grid iteration is composed of three steps; pre-smoothing, coarse-grid correction, and post-smoothing. The details can be described as follows.

Two-Grid Iteration Algorithm

1. (**Pre-smoothing**) Compute a smoothed approximate solution $\bar{\boldsymbol{z}}^h$ by applying μ_1 iterations of a relaxation method to a current approximate solution $\tilde{\boldsymbol{z}}^h$.
2. (**Coarse-grid correction**) Correct $\bar{\boldsymbol{z}}^h$ with the error $\boldsymbol{e}^h$ approximated with the coarse grid as follows: (a) Compute the residual on the fine grid $\boldsymbol{r}^h$ with $\boldsymbol{r}^h = \boldsymbol{b}^h - A^h\bar{\boldsymbol{z}}^h$. (b) Restrict the residual $\boldsymbol{r}^h$ to the coarse grid as $\boldsymbol{r}^H = R^{h\to H}\boldsymbol{r}^h$. (c) Compute the error on the coarse grid $\boldsymbol{e}^H$ by solving a linear system $A^H\boldsymbol{e}^H = \boldsymbol{r}^H$. (d) Prolong the error $\boldsymbol{e}^H$ to the fine grid as $\boldsymbol{e}^h = P^{H\to h}\boldsymbol{e}^H$. (e) Correct the approximate solution $\bar{\boldsymbol{z}}^h$ by $\bar{\boldsymbol{z}}'^h = \boldsymbol{z}^h + \boldsymbol{e}^h$.
3. (**Post-smoothing**) Compute the new approximation $\tilde{\boldsymbol{z}}'^h$ by applying μ_2 iterations of the relaxation method to $\bar{\boldsymbol{z}}'^h$.

The two-grid iteration is repeated recursively down to some coarsest grid. The recursive procedure is the multigrid method. One iteration of a multigrid method is called a cycle, and the exact structure of a cycle depends on the the value of γ, the number of two-grid iterations at each immediate step. The case $\gamma = 1$ is called a multigrid V-cycle (MV), while $\gamma = 2$ is called a multigrid W-cycle (MW). Let us denote by h' the index of cell size on the current grid and by H' the index of cell size on some coarsest grid. Then, the multigrid iteration can be defined recursively by the following two steps:

Multigrid Iteration Algorithm

1. If $h' = H'$, then return an exact solution $\bar{\boldsymbol{z}}_{H'}$ obtained by solving $\bar{\boldsymbol{z}}_{H'} = A_{H'}^{-1}\boldsymbol{b}_{H'}$.
2. Otherwise, return an $\bar{\boldsymbol{z}}_{h'}$ obtained by recursively applying γ iterations of a multigrid iteration, from h' to H'.

Table 1. Comparison of the block Gauss-Seidel relaxation and the implicit block Gauss-Seidel relaxation

	α	n	ζ^n	Time [s]	Speedup
Block Gauss-Seidel	10	1356	0.499	59.8	1
	1	107	0.499	4.6	13
Block Gauss-Seidel	10	37	0.499	8.7	7
(V(1,1))	1	3	0.485	0.7	85
Block Gauss-Seidel	10	37	0.499	8.6	7
(W(1,1))	1	3	0.485	0.7	85
Implicit block Gauss-Seidel	10	302	0.499	19.0	3
	1	36	0.499	2.3	26
Implicit block Gauss-Seidel	10	5	0.496	1.3	46
(V(1,1))	1	2	0.418	0.5	120
Implicit block Gauss-Seidel	10	2	0.464	0.6	100
(W(1,1))	1	1	0.489	0.3	200

Table 2. Comparison of different multigrid implementations of the implicit block Gauss-Seidel relaxation

	Yosemite			Flower Garden			Rubik Cube		
	n	ζ^n	Time [s]	n	ζ^n	Time [s]	n	ζ^n	Time [s]
V(1,1)	19	0.267	4.8	16	0.282	4.3	17	0.211	6.4
V(2,2)	17	0.263	5.7	11	0.285	4.2	10	0.229	5.7
W(1,1)	17	0.266	5.3	11	0.281	3.7	4	0.287	1.9
W(2,2)	16	0.262	6.7	9	0.283	4.0	4	0.299	2.5

5 Experimental Results

All numerical experiments were carried out on a desktop PC with a 3.2-GHz Intel P4 CPU executing C/C++ code. The qualitative analysis of dense 3-D interpretation was conducted by using a display of gray value rendering of a dense map of relative depth as in [8,10]. To quantitatively evaluate the accuracy of 3-D interpretation, we have computed the error reduction factor defined by $\zeta^n = \sqrt{e^n/e^0}$, where e^n denotes the sum of squared intensity errors at iteration n. We used four grids for multigrid V-cycle or W-cycle iterations. We denote by V(1,1) or W(1,1) the V-cycle or W-cycle with one pre-smoothing and one post-smoothing iteration at each grid. The contrast parameter of our model, ν was set to 2. In all experiments, we set the field of view of camera to 45°.

In the first experiment, we compared the performance of BGSR and IBGSR. The previous methods in [8,10] used BGSR for performing dense 3-D interpretation. In each test run, the algorithm was stopped when the error reduction factor was below 0.5. We also evaluated the performance of each relaxation with V(1,1) or W(1,1). The first two frames of the *Yosemite* sequence were tested for this experiment. Table 1 shows the required number of iterations, n, to reach the

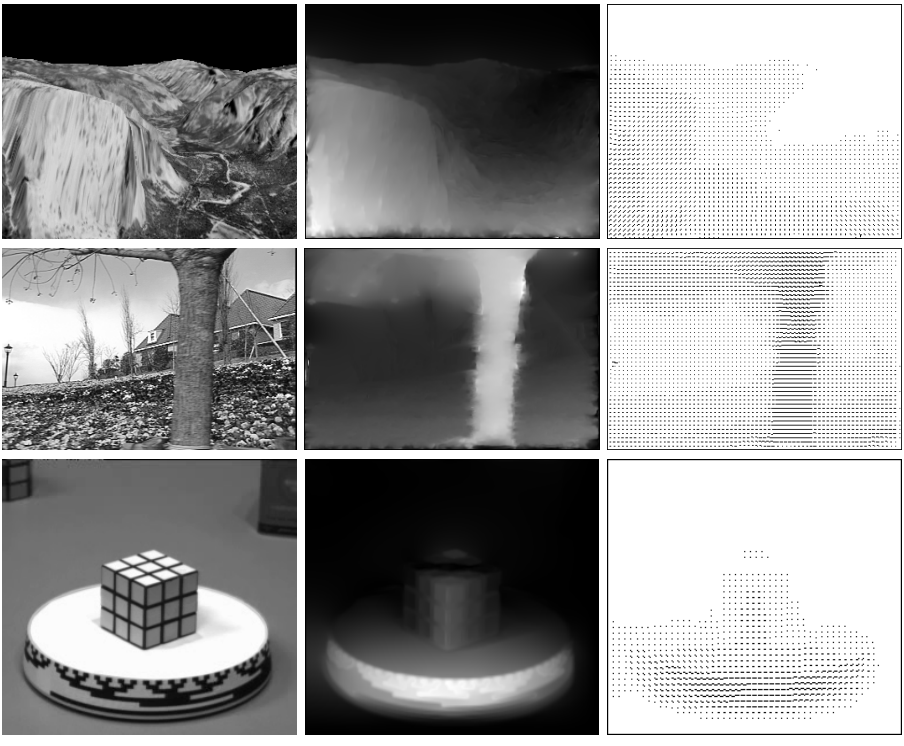

Fig. 1. *Top*: Result of the *Yosemite* sequence (ν=2.0, α=1, 4.8 s with 19 V(1,1) cycles). *Middle*: Result of the *Flower Garden* sequence (ν=2.0, α=1, 3.7 s with 10 W(1,1) cycles). *Bottom*: Result of the *Rubik Cube* sequence (ν=2.0, α=10, 1.9 s with 4 W(1,1) cycles).

desired error reduction of $\zeta^n < 0.5$, the computation time in seconds, and the speedup factor with respect to BGSR. While IBGSR computed the source term $\boldsymbol{b}^n$ in each iteration, it was about 3 times faster than BGSR. While multigrid methods provided a significant acceleration to both relaxation methods, IBGSR reached the desired error reduction faster than BGSR with V(1,1) or W(1,1).

In our second experiment, different multigrid implementations of IBGSR were tested. We performed tests with the first two frames of the *Yosemite* sequence, the first two frames of the *Flower Garden* sequence of 352×240 pixels, and the first and the fifth frames of the *Rubik Cube* sequence of 256×240 pixels. The *Rubik Cube* sequence has a rotating object, and thereby we have estimated the rotational motion $\boldsymbol{w}$ as well as the translational motion $\boldsymbol{t}$. The algorithm was stopped when $\zeta^n - \zeta^{n-1} \leq 10^{-3}$. IBGSR was applied to each test set with 4-grid V-cycle or W-cycle. Table 2 shows the required number of V or W cycles n, the error reduction factor ζ^n, and the computation time for each multigrid implementation. To demonstrate the quality of 3-D interpretation, we display

Table 3. Comparison of the previous methods (MDL [8], Diffusion [10]) and our method (IBGSR, V(1,1)) for the *Yosemite* sequence

	n	ζ^n	κ [%]	ϕ [°]	Time [s]
MDL	10000	0.459	35.06	13.91	6061.4
Diffusion	10000	0.478	36.64	17.17	989.5
Our method	19	0.267	23.65	4.11	4.8

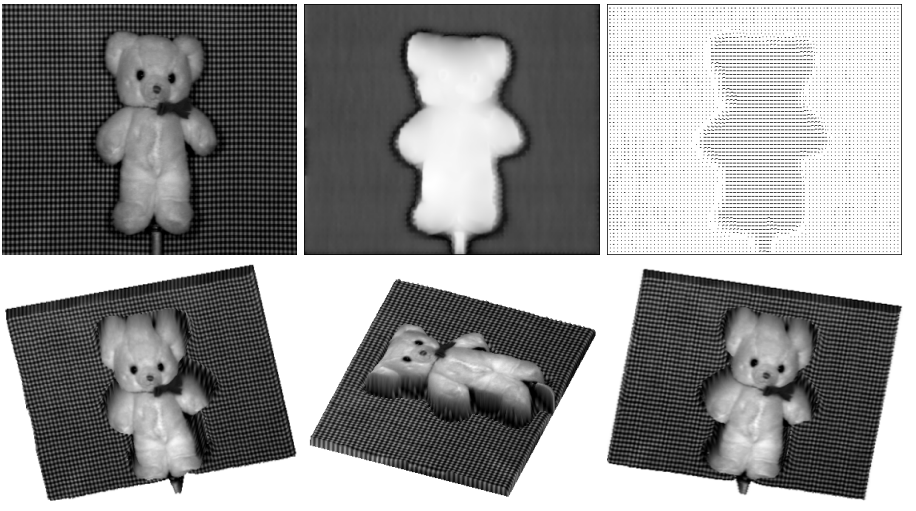

Fig. 2. *Top*: Result of the *Teddy* sequence (ν=2.0, α=5, 3.8 s with 5 W(2,2) cycles). *Bottom*: Different views of the reconstructed 3-D structure by our method.

the first frame, the gray value rendering result of the estimated depth map, and the optical flow computed by Eq. (1) for three test set in Figure 1. Note that our method has output the full 3-D information of each test set in a few seconds.

In Table 3, we show the performance of our method compared to the previous methods, MDL [8] and Diffusion [10]. For the quantitative analysis, we calculated the average angular error ϕ and the relative depth error κ, defined by $\kappa = ||a\hat{\boldsymbol{d}} + b - \boldsymbol{d}||/||\boldsymbol{d}||$, where $\boldsymbol{d}$ denotes the ground truth depth, $\hat{\boldsymbol{d}}$ the estimated depth, and a and b the scale and the bias factors of the estimated depth map, respectively. The maximum number of iterations was set to 10^4. As we see, the previous methods could not reach the same error reduction as our method.

In the last experiment, we experimentally verified that our method allows the movement of both viewing system and viewed objects. This was demonstrated using the *Teddy* sequence, where an object moves laterally and a textured background moves in the opposite direction. We used the first two frames after resizing to 400×340 pixels and set ν=2 and α=5 for our model parameters. W(2,2) was selected for the multigrid implementation. Figure 2 shows the frame 1 of the *Teddy* sequence, the estimated depth map between the first two frames, and the

computed optical flow field. The computation time was 3.8 seconds. That figure also displays different views of the reconstructed 3-D structure by our method.

6 Conclusions

We have presented a fast method for dense 3-D interpretation method without prior estimation of optical flow. The proposed method can conduct fast 3-D interpretation on large motion fields. The Nagel-Enkelman technique was employed in the variational model of our problem, and we have derived diffusion-reaction equations. The idea of implicit schemes and multigrid methods has been combined to develop a new implicit multigrid block Gauss-Seidel relaxation scheme for solving our dense 3-D interpretation problem quickly on large motion fields. We have verified the efficiency and effectiveness of our method through the experimental results with synthetic and real image sequences.

References

1. Alvarez, L.: Images and PDE's. ICAOS 1996 Images, Wavelets and PDE's 219, 3–14 (1996)
2. Cohen, M., Irani, M., Anandan, P.: Direct recovery of planar-parallax from multiple frames. IEEE Trans. on Pattern Anal. and Mach. Intell. 24(11), 1528–1534 (2002)
3. Chellappa, R., Qian, G., Srinivasan, S.: Invited paper: Structure from motion: sparse versus dense correspondence methods. In: Proc. IEEE Conf. on Image Processing, pp. 492–499. IEEE Computer Society Press, Los Alamitos (1999)
4. Hackbusch, W.: Applied mathematical sciences, vol. 95. Springer, New York (1993)
5. Henson, V.E., Briggs, W.L., McCormick, S.F.: A multigrid tutorial. In: SIAM (2000)
6. Longuet-Higgins, H.C.: A computer algorithm for reconstructing a scene from two projections. Nature 293, 133–135 (1987)
7. Mitiche, A.: Computational analysis of visual motion. Plenum Press, New York (1994)
8. Mitiche, A., Hadjres, S.: MDL estimation of a dense map of relative depth and 3d motion from a temporal sequence of images. Pattern Analysis and Applications 6, 78–87 (2003)
9. Nagel, H.H., Enkelmann, W.: An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences. IEEE Trans. Pattern Anal. and Mach. Intell. 8, 565–593 (1986)
10. Sekkati, H., Mitiche, A.: Dense 3D interpretation of image sequences: A variational approach using anisotropic diffusion. In: Proc. Int. Conf. on Image Analysis and Processing (2003)
11. Stein, S., Shashua, M.: Model-based brightness constraints: On direct estimation of structure and motion. IEEE Trans. on Pattern Anal. and Mach. Intell. 22(9), 993–1005 (2000)
12. Vedula, S., Baker, S., Rander, P., Collins, R., Kanade, T.: Three-dimensional scene flow. IEEE Trans. on Pattern Anal. and Mach. Intell. 27(3), 475–480 (2005)
13. Weickert, J., Romeny, B.M., Viergever, M.A.: Efficient and reliable schemes for nonlinear diffusion filtering. IEEE Trans. Image Process. 7, 398–410 (1998)