Dense 3D Interpretation of Image Sequences: A variational Approach Using Anisotropic Diffusion

H. Sekkati and A. Mitiche
INRS-EMT
Place Bonaventure 800 rue de la Gauchetiere O.
Suite 6900 Montreal, Quebec H5A 1K6
sekkati{mitiche}@inrs-telecom.uquebec.ca

Abstract

The purpose of this study is to investigate a new method for recovering relative depth and 3D motion from a temporal sequence of monocular images. The method is direct insomuch as it does not require computation of image motion prior to 3D interpretation. This interpretation is obtained by minimizing a functional with two characteristic terms, one of conformity to the spatiotemporal changes in the image sequence, the other of regularization based on anisotropic diffusion. The Euler-Lagrange equations corresponding to the functional minimization are solved iteratively via the half-quadratic algorithm.

1. Introduction

2D-to-3D conversion of image temporal sequences, where one estimates relative depth and 3D motion from the sequence spatio-temporal changes, is a fundamental, difficult problem in computer vision, and one that serves numerous applications such as robot autonomous navigation, augmented and virtual reality rendering, and animation. In the context of motion analysis, the literature on 2D-to-3D conversion of temporal monocular sequences is relatively scarce [1, 7, 9, 11, 6, 13, 14], particularly when the scene contains moving objects and the camera is allowed to move [10].

The goal of this study is to investigate a new method for 3D interpretation of image sequences. Motion is considered relative as objects are allowed to move as well as the viewing system. The method is based on the minimization of an energy functional with two characteristic terms: (1) a term of conformity to the spatio-temporal changes of the image sequence. Under the assumption that moving objects are rigid, this term is obtained by substituting for optical velocity in the gradient equation [5] its expression in terms of the 3D variables of depth and 3D motion parameters, and (2) a regularization term based on anisotropic diffusion so as to preserve boundaries of the 3D interpretation. Minimization of the energy functional follows from the Euler-Lagrange equations which are solved efficiently using the half-quadratic algorithm [2].

2. Constraint equation of direct methods

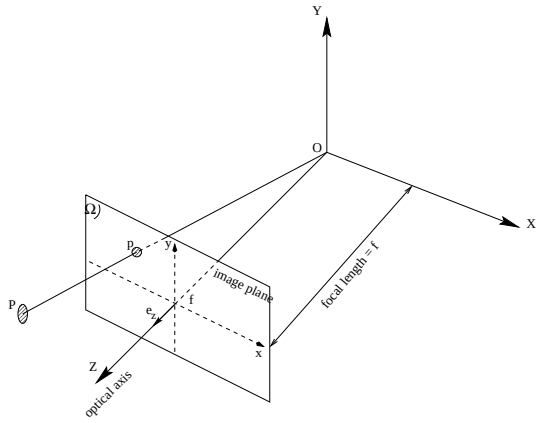

Figure 1. camera model: perspective projection

The image acquisition process is modeled by perspective projection as shown in the Figure 1, in which 3D point $\mathbf{P} = (X, Y, Z)^t$ projects onto 2D image point $\mathbf{p} = (x, y, f)^t$. The coordinates of P and p are related via the transformation:

$$\mathbf{p} = (x, y, f)^t = (f\frac{X}{Z}, f\frac{Y}{Z}, f)^t = f\frac{\mathbf{P}}{Z} \qquad (1)$$

where f is the focal length of the camera. In rigid body motion, the optical velocity (u, v) at coordinates (x, y) on

In Proceedings of International Conference on Image Analysis and Processing, ICIAP-03, Sept. 17-19, 2003, Mantova, Italy

the image plane can be expressed, in terms of 3D translational velocity $\mathbf{T} = (t_1, t_2, t_3)^t$, rotational velocity $\omega = (\omega_1, \omega_2, \omega_3)^t$, and depth Z, as [8],

$$\left\{ \begin{array}{lcl} u & \triangleq & \dot{x} = \frac{1}{Z}(ft_1 - xt_3) - \frac{xy}{f}\omega_1 + \frac{f^2+x^2}{f}\omega_2 - y\omega_3 \\ v & \triangleq & \dot{y} = \frac{1}{Z}(ft_2 - yt_3) - \frac{f^2+y^2}{f}\omega_1 + \frac{xy}{f}\omega_2 + x\omega_3 \end{array} \right. \quad (2)$$

Let $I : \Omega \times]0, T[$ be an image sequence, where Ω is an open subset of $\mathbb{R}^2$ representing the image domain, and $]0, T[$ is the interval of duration of the sequence. The assumption that the brightness from a point on a surface in the scene does not change during motion leads to the gradient equation [5]:

$$\frac{dI}{dt} = I_t + \dot{x}I_x + \dot{y}I_y = 0 \quad (3)$$

where I_x, I_y and I_t are the spatio-temporal derivatives of the image. For a rigid body motion, we substitute (2) in (3) to obtain the constraint equation for rigid body motion [8]

$$I_t + \mathbf{s} \cdot \tau + \mathbf{q} \cdot \omega = 0 \quad (4)$$

where τ, $\mathbf{s}$, and $\mathbf{q}$ are given by

$$\begin{array}{rcl} \tau & = & \frac{\mathbf{T}}{Z} \\ \mathbf{s} & = & \begin{pmatrix} fI_x \\ fI_y \\ -xI_x - yI_y \end{pmatrix} \\ \mathbf{q} & = & \begin{pmatrix} -fI_y - \frac{y}{f}(xI_x + yI_y) \\ fI_x + \frac{x}{f}(xI_x + yI_y) \\ -yI_x + xI_y \end{pmatrix} \end{array} \quad (5)$$

Equation (4) still holds when we scale both Z and T by the same factor. Consequently we can determine only the direction of translational velocity and relative depth of objects.

3. Recovery of relative depth and 3D motion

Allowing translational 3D motions, we seek a 3D interpretation of the image sequence that minimizes the following energy functional:

$$\begin{array}{rcl} E(\tau) & = & \iint_\Omega \left[(I_t + s \cdot \tau)^2 + \lambda \left(\Phi(\|\nabla\tau_1\|) \right.\right. \\ & & \left.\left. + \Phi(\|\nabla\tau_2\|) + \Phi(\|\nabla\tau_3\|) \right) \right] dx\, dy \end{array} \quad (6)$$

A similar functional can be writing for general rigid motion that includes the variables of the rotational components of motion. The following analysis applies to general rigid motion as well. In (6) the first term in the integrand is the term of conformity to data according to (4). The other term is a regularization term. Function Φ is defined as in [2] so that it realizes an anisotropic diffusion to preserve the boundaries of the interpretation, and it is determined from the following conditions [2]

$$\left\{ \begin{array}{rcl} \lim\limits_{s\to 0} \frac{\Phi'(s)}{s} & = & \lim\limits_{s\to 0} \Phi''(s) = \Phi''(0) > 0 \\ \lim\limits_{s\to +\infty} \frac{\Phi'(s)}{s} & = & \lim\limits_{s\to +\infty} \Phi''(s) = 0 \\ \lim\limits_{s\to +\infty} \frac{\Phi''(s)}{\frac{\Phi'(s)}{s}} & = & 0 \end{array} \right. \quad (7)$$

There are several functions that satisfy (7), for instance the Aubert function $\Phi(s) = 2\sqrt{1+s^2} - 2$, which we use here. The Euler-Lagrange partial differential equations corresponding to (6) are:

$$\left\{ \begin{array}{rcl} \lambda\, div\left(\frac{\Phi'(\|\nabla\tau_1\|)}{\|\nabla\tau_1\|}\nabla\tau_1\right) & = & 2s_1(I_t + \mathbf{s}\cdot\tau) \\ \lambda\, div\left(\frac{\Phi'(\|\nabla\tau_2\|)}{\|\nabla\tau_2\|}\nabla\tau_2\right) & = & 2s_2(I_t + \mathbf{s}\cdot\tau) \\ \lambda\, div\left(\frac{\Phi'(\|\nabla\tau_3\|)}{\|\nabla\tau_3\|}\nabla\tau_3\right) & = & 2s_3(I_t + \mathbf{s}\cdot\tau) \end{array} \right. \quad (8)$$

with boundary conditions

$$\frac{\partial\tau_1}{\partial\mathbf{n}} = 0, \frac{\partial\tau_2}{\partial\mathbf{n}} = 0, \frac{\partial\tau_3}{\partial\mathbf{n}} = 0$$

where $\mathbf{n}$ indicates the unit vector normal to the boundary $\partial\Omega$ of Ω.

4. Energy minimization: half quadratique algorithm

A discretization of (8) yields a large system of non linear equations difficult to solve in general. Rather than solving such a system, we use the half-quadratic minimization algorithm proposed in [2] where it was used for image restauration and optical flow estimation. With the half quadratic algorithm, we minimizes (6) via the minimization of another functional which, in our case, is:

$$E^*(\tau, \mathbf{b}) = \iint_\Omega \left((I_t + \mathbf{s}\cdot\tau)^2 + \lambda C^*(\tau, \mathbf{b}) \right) dx\, dy \quad (9)$$

where

$$\begin{array}{rcl} C^*(\tau, \mathbf{b}) & = & b_{\tau_1}\|\nabla\tau_1\|^2 + \psi(b_{\tau_1}) + b_{\tau_2}\|\nabla\tau_2\|^2 + \psi(b_{\tau_2}) \\ & & + b_{\tau_3}\|\nabla\tau_3\|^2 + \psi(b_{\tau_3}) \end{array}$$

$\mathbf{b} = (b_{\tau_1}, b_{\tau_2}, b_{\tau_2})^t$ is a field of auxiliary variables, and ψ is a strictly decreasing convex function. The demonstration for the existence of those variables can be found in the reference indicated above. The advantage of using the functional (9) instead of (6) is two-fold.

- For a fixed τ, $E^*(\tau, \mathbf{b})$ is convex in $\mathbf{b}$

- For a fixed $\mathbf{b}$, $E^*(\tau, \mathbf{b})$ is convex in τ

This property of $E^*(\tau, \mathbf{b})$ allows a two-step minimization algorithm. The first step consist of computing the minimum of E^* with respect to $\mathbf{b}$ by fixing τ, followed by finding the minimum of E^* with respect to τ by fixing $\mathbf{b}$. These two steps are repeated until convergence. The half quadratic algorithm reads as follows:

$\tau^{(0)} \equiv 0$
Repeat
$$\mathbf{b}^{(n+1)} = arg\min_{\{\mathbf{b}\}} E^*(\tau^{(n)}, \mathbf{b})$$
$$\tau^{(n+1)} = arg\min_{\{\tau\}} E^*(\tau, \mathbf{b}^{(n+1)})$$
$$n = n + 1$$
until convergence

Table 1. Minimization by the half quadratic algorithm

The first step, which consist of computing $\mathbf{b}^{(n+1)} = arg\min_{\{\mathbf{b}\}} E^*(\tau^{(n)}, \mathbf{b})$, is equivalent to finding the minimum of

$$\iint_\Omega C^*(\tau, \mathbf{b})\, dx\, dy$$

The minimum of this functional can be computed analytically, and is reached for $\mathbf{b}^F = (b^F_{\tau_1}, b^F_{\tau_2}, b^F_{\tau_3})^t$ where

$$b^F_{\tau_1} = \frac{\Phi'(||\nabla\tau_1||)}{2||\nabla\tau_1||} = \frac{1}{\sqrt{1+||\nabla\tau_1||^2}} = g(||\nabla\tau_1||)$$
$$b^F_{\tau_2} = \frac{\Phi'(||\nabla\tau_2||)}{2||\nabla\tau_2||} = \frac{1}{\sqrt{1+||\nabla\tau_2||^2}} = g(||\nabla\tau_2||)$$
$$b^F_{\tau_3} = \frac{\Phi'(||\nabla\tau_3||)}{2||\nabla\tau_3||} = \frac{1}{\sqrt{1+||\nabla\tau_3||^2}} = g(||\nabla\tau_3||)$$

The second step, which compute $\tau^{(n+1)} = arg\min_{\{\tau\}} E^*(\tau, \mathbf{b}^{(n+1)})$, consist of finding the minimum of

$$\iint_\Omega \left((I_t + \mathbf{s}\cdot\tau)^2 + \lambda C^*(\tau, \mathbf{b}^F)\right)\, dx\, dy$$

The Euler-Lagrange equations corresponding to this functional are deduced in the same way as (8) and are given by

$$\left\{\begin{array}{rcl} \lambda\ div(b^F_{\tau_1}\nabla\tau_1) &=& 2s_1(I_t + \mathbf{s}\cdot\tau) \\ \lambda\ div(b^F_{\tau_2}\nabla\tau_2) &=& 2s_2(I_t + \mathbf{s}\cdot\tau) \\ \lambda\ div(b^F_{\tau_3}\nabla\tau_3) &=& 2s_3(I_t + \mathbf{s}\cdot\tau) \end{array}\right. \quad (10)$$

with boundary conditions

$$\frac{\partial\tau_1}{\partial\mathbf{n}} = 0, \frac{\partial\tau_2}{\partial\mathbf{n}} = 0, \frac{\partial\tau_3}{\partial\mathbf{n}} = 0$$

5. Discretization

We discretize the image domain Ω into a unit-spacing grid D with points indexed by (i,j) $i = 1,\ldots,N$, $j = 1,\ldots,M$. The divergence terms in (10) can be discretized as in [12]; for each $(i,j) \in D$ we have:

$$\begin{array}{rcl} \left[div(b^F_{\tau_k}\nabla\tau_k)\right]_{i,j} &\simeq& \beta\left[b^N_{\tau_k}\nabla^N\tau_k + b^S_{\tau_k}\nabla^S\tau_k\right. \\ && \left. + b^E_{\tau_k}\nabla^E\tau_k + b^W_{\tau_k}\nabla^W\tau_k\right]_{i,j} \end{array} \quad (11)$$

where $0 \le \beta \le \frac{1}{4}$ is a coefficient for numerical stabilization scheme, N, S, E, W are the mnemonic subscripts for north, south, east and west. These subscripts on the operator ∇, indicate nearest neighbor difference in the corresponding direction,

$$\begin{array}{rcl} \nabla^N\tau_{k(i,j)} &\equiv& \tau_{k(i-1,j)} - \tau_{k(i,j)} \\ \nabla^S\tau_{k(i,j)} &\equiv& \tau_{k(i+1,j)} - \tau_{k(i,j)} \\ \nabla^E\tau_{k(i,j)} &\equiv& \tau_{k(i,j+1)} - \tau_{k(i,j)} \\ \nabla^W\tau_{k(i,j)} &\equiv& \tau_{k(i,j-1)} - \tau_{k(i,j)} \end{array} \quad (12)$$

The coefficients $b^N_{\tau_k}, b^S_{\tau_k}, b^E_{\tau_k}, b^W_{\tau_k}$ can be approximated by the equations

$$\begin{array}{rcl} b^N_{\tau_k(i,j)} &\simeq& g(|\nabla^N\tau_{k(i,j)}|) \\ b^S_{\tau_k(i,j)} &\simeq& g(|\nabla^S\tau_{k(i,j)}|) \\ b^E_{\tau_k(i,j)} &\simeq& g(|\nabla^E\tau_{k(i,j)}|) \\ b^W_{\tau_k(i,j)} &\simeq& g(|\nabla^W\tau_{k(i,j)}|) \end{array} \quad (13)$$

The discretized version of (10) is then obtained by the system

$$\begin{array}{rcl} \alpha\ \left[b^N_{\tau_k}\nabla^N\tau_k + b^S_{\tau_k}\nabla^S\tau_k + b^E_{\tau_k}\nabla^E\tau_k + b^W_{\tau_k}\nabla^W\tau_k\right]_{i,j} &=& \\ \left[s_k(I_t + \mathbf{s}\cdot\tau)\right]_{i,j} & & \end{array} \quad (14)$$

for $k = 1,2,3$ and where $\alpha = \frac{1}{2}\lambda\beta$. With this discretization we have at each $(i,j) \in D$:

$$\left(\begin{array}{ccc} \alpha\overline{b}_{\tau_1} + s_1^2 & s_1s_2 & s_1s_3 \\ s_1s_2 & \alpha\overline{b}_{\tau_2} + s_2^2 & s_2s_3 \\ s_1s_3 & s_2s_3 & \alpha\overline{b}_{\tau_3} + s_3^2 \end{array}\right)\left(\begin{array}{c}\tau_1\\ \tau_2\\ \tau_3\end{array}\right) = \left(\begin{array}{c} \alpha\overline{\tau_1} - s_1I_t \\ \alpha\overline{\tau_2} - s_2I_t \\ \alpha\overline{\tau_3} - s_3I_t \end{array}\right) \quad (15)$$

where the coefficients $\overline{\tau}_k$ and $\overline{b}_{\tau_k}$ are given by

$$\begin{array}{rcl} \overline{\tau}_{k(i,j)} &=& b^N_{\tau_k(i,j)}\,\tau_{k(i-1,j)} + b^S_{\tau_k(i,j)}\,\tau_{k(i+1,j)} \\ && + b^E_{\tau_k(i,j)}\,\tau_{k(i,j+1)} + b^W_{\tau_k(i,j)}\,\tau_{k(i,j-1)} \\ \overline{b}_{\tau_k(i,j)} &=& b^N_{\tau_k(i,j)} + b^S_{\tau_k(i,j)} + b^E_{\tau_k(i,j)} + b^W_{\tau_k(i,j)} \end{array}$$

We notice here that the coefficients $\overline{\tau}_k$ and $\overline{b}_{\tau_k}$ are computed using the neighborhood of (i,j). Solving (15), gives at each $(i,j) \in D$

$$\begin{cases} \tau_1 = \dfrac{1}{\overline{b}_{\tau_1}} (\overline{\tau_1} - s_1 q) \\ \tau_2 = \dfrac{1}{\overline{b}_{\tau_2}} (\overline{\tau_2} - s_2 q) \\ \tau_3 = \dfrac{1}{\overline{b}_{\tau_3}} (\overline{\tau_3} - s_3 q) \end{cases} \quad (16)$$

where

$$q = \frac{s_1 \overline{b}_{\tau_2} \overline{b}_{\tau_3} \overline{\tau_1} + s_2 \overline{b}_{\tau_1} \overline{b}_{\tau_3} \overline{\tau_2} + s_3 \overline{b}_{\tau_1} \overline{b}_{\tau_2} \overline{\tau_3} + \overline{b}_{\tau_1} \overline{b}_{\tau_2} \overline{b}_{\tau_3} I_t}{\alpha \overline{b}_{\tau_1} \overline{b}_{\tau_2} \overline{b}_{\tau_3} + s_1^2 \overline{b}_{\tau_2} \overline{b}_{\tau_3} + s_2^2 \overline{b}_{\tau_1} \overline{b}_{\tau_3} + s_3^2 \overline{b}_{\tau_1} \overline{b}_{\tau_2}}$$

When (16) is written for all $(i,j) \in D$, we obtain a large sparse system of linear equations which can be solved efficiently using the Gauss-Seidel algorithm [3]. Following convergence of the half-quadratic algorithm, the direction of motion at each point is directly given by the three components of τ, and depth is recovered by

$$\frac{1}{Z} = \sqrt{\tau_1^2 + \tau_2^2 + \tau_3^2} \quad (17)$$

6. Experimental results

We will start our experiment by estimating the 3D directions and the depth from a synthetic sequence consisting of a sinusoidal square moving against a fixed background. The first frame of this sequence is given in the Figure (2.a).

The square in the middle of the background is moving in the opposite diagonal ($y = -x$) which means a discrete motion with $T = (-1, -1, 0)^t$. The estimated translational directions and the recovered depth are shown in Figures 2.(c, d, e et f).

Another criteria to evaluate the method for synthetic sequences is the reconstructed optical flow, i.e., computed from (2). We are expecting to obtain for the sinusoidal sequence a field $\{(u,v) = (-1,-1)$ for (x,y) in the moving square$\}$. The result of the optical flow reconstructed from the equations (2) is represented in the Figure (2.b) and is accurate with the expected result.

For real image sequences, where relative depth in the scene is not known, we can evaluate the results in two ways.

1) A display of a gray-level rendering of depth. This will show graphically the variation of depth that can be compared subjectively to our visual system expectation of this variation. A gray-level rendering of the results for two real image sequences is shown in Figure 3.b (Flower-Garden sequence) and Figure 3.e (Manege sequence).

2) Viewing of an anaglyph stereoscopic image constructed from an image of the monocular sequence and the estimated depth for that image. Construction of an anaglyph image is done as follows. We are given an image, call it I_1, and the corresponding depth map. We construct an image I_2, as if acquired by a fictitious camera as follows. Let S_1 be the viewing system representing the "real" camera which acquired I_1, and S_2 that of the other "fictitious" camera. Both viewing systems are as in Figure 1. S_2 is placed to differ from S_1 by a translation of amount $d \ll f$ along the X-axis. Let P be a point of the real scene. We note by, (X_1, Y_1, Z_1) and (X_2, Y_2, Z_2) the coordinates of P in the systems S_1 and S_2 (resp.), and by (x_1, y_1) and (x_2, y_2) the corresponding image coordinates (resp.). These coordinates are related by:

$$\begin{cases} x_1 = f\dfrac{X_1}{Z_1} \\ y_1 = f\dfrac{y_1}{Z_1} \end{cases}, \begin{cases} x_2 = f\dfrac{X_2}{Z_2} \\ y_2 = f\dfrac{y_2}{Z_2} \end{cases}$$

$$and \begin{cases} X_2 = X_1 - d \\ Y_2 = Y_1 \\ Z_2 = Z_1 \end{cases}$$

Image I_2 is constructed as follows:

$$\begin{aligned} I_2(x_2, y_2) &= I_1(x_1, y_1) \\ &= I_1(x_2 + \frac{df}{Z_1}, y_2) \end{aligned}$$

and given the depth from the equation (17) we obtain

$$I_2(x_2, y_2) = I_1(x_2 + df\sqrt{\tau_1^2 + \tau_2^2 + \tau_3^2}, y_2) \quad (18)$$

The anaglyph images constructed for the Flower-Garden and the Manege are shown in Figure 3.c and Figure 3.f. They are to be viewed with chromatic (red-blue) glasses (the common plastic ones). When viewed this way, we experience a strong sense of depth both for the Flower-Garden and the Manege. The anaglyph was generated using the algorithm in [4], with the permission of Eric Dubois, its inventor. **Note:** we will bring glasses for the audience to view the 3D reconscruction, if this paper is presented.

Note that we can also, indirectly, subjectively evaluate the results by reconstructing the optical flow field from equation (2) and graphically compare the results with our expectation of the variations of this field.

7. Conclusion

We presented a direct method for dense 3D interpretation of temporal sequence of images. This is a variational method which uses anisotropic diffusion to preserve the boundaries of the 3D interpretation. Promizing results have been obtained on several experiments on synthetic and real image sequences. We can easily extend the method to general motion. The energy functional will differ from the one used in (6) by an additional term of regularization for the rotation field ω.

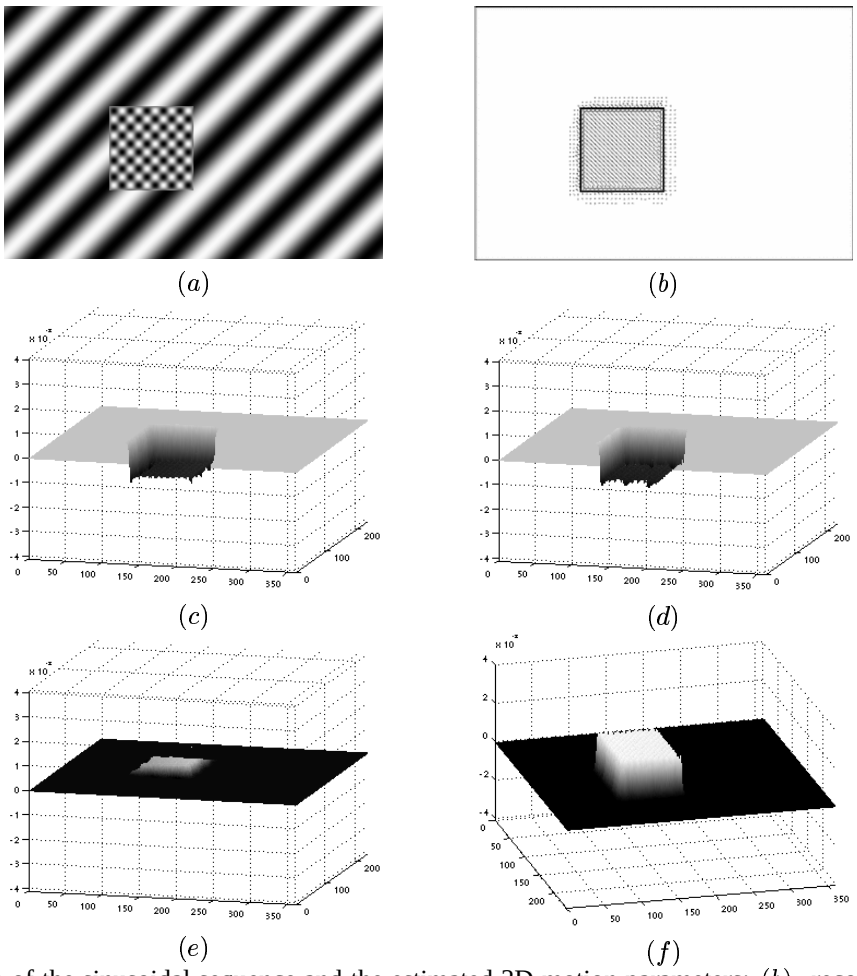

Figure2. (a) A frame of the sinusoidal sequence and the estimated 3D motion parameters: (b) reconstructed optical flow (c) $\tau_1 = \{-10^{-3}, \forall(i,j)$ in the moving square $\}$, (d) $\tau_2 \approx \tau_1$, (e) $\tau_3 \approx 0$ and the relative depth (f) $\frac{1}{Z} = \{10^{-3}, \forall(i,j)$ in the moving square $\}$

References

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

[2] G. Aubert R. Deriche and P. Kornprobst. Computing optical flow via variational thechniques. *SIAM Journal of Applied Mathematics*, 60(1):156–182, 1999.

[3] P. Ciarlet. *Introduction a l'analyse numerique matricielle et a l'optimisation*. Masson, fifth edition, 1994.

[4] E. Dubois. A projection method to generate anaglyph stereo images. volume III, pages 1661–1664. Proc. ICASP, 2001.

[5] B. Horn and B. Schunk. Determining optical flow. *Artif. Intell.*, (17):185–203, 1981.

[6] Y. Hung and H. Ho. A kalman filter approach to direct depth estimation incorporating surface structure. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21(6):570–575, 1999.

[7] R. Laganiere and A. Mitiche. Direct bayesian interpretation of visual motion. *Journal of Robotics and Autonomous Systems*, (14):247–254, 1995.

[8] H. Longuet-Higgins and K. Prazdny. The interpretation of a moving retinal image. pages 385–397. Proc. R. Soc. London, Ser. B 208.

[9] F. Martinez, J. Benois-Pineau, and D. Barda. Extraction of the relative depth information of objects in video sequences. pages 948–952. In Proccedings of International conference on Image Precessing, 1998.

[10] A. Mitiche and S. Hadjres. Mdl estimation of a dense map of depth and 3d motion from a temporal sequence of images. *Pattern Analysis and Applications*, 2003 (to appear).

[11] F. Morier, H. Nicolas, J. Benois, D. Barda, and H. Hanson. Relative depth estimation of video objects for image interpolation. pages 953–957. In Proccedings of International conference on Image Precessing, 1998.

[12] P. Perona and J. Malik. Scale space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(7):629–639, 1981.

[13] S. Stein and M. Shashua. Model-based brightness bonstraints: on direct estimation of structure and motion. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(9):993–1005, 2000.

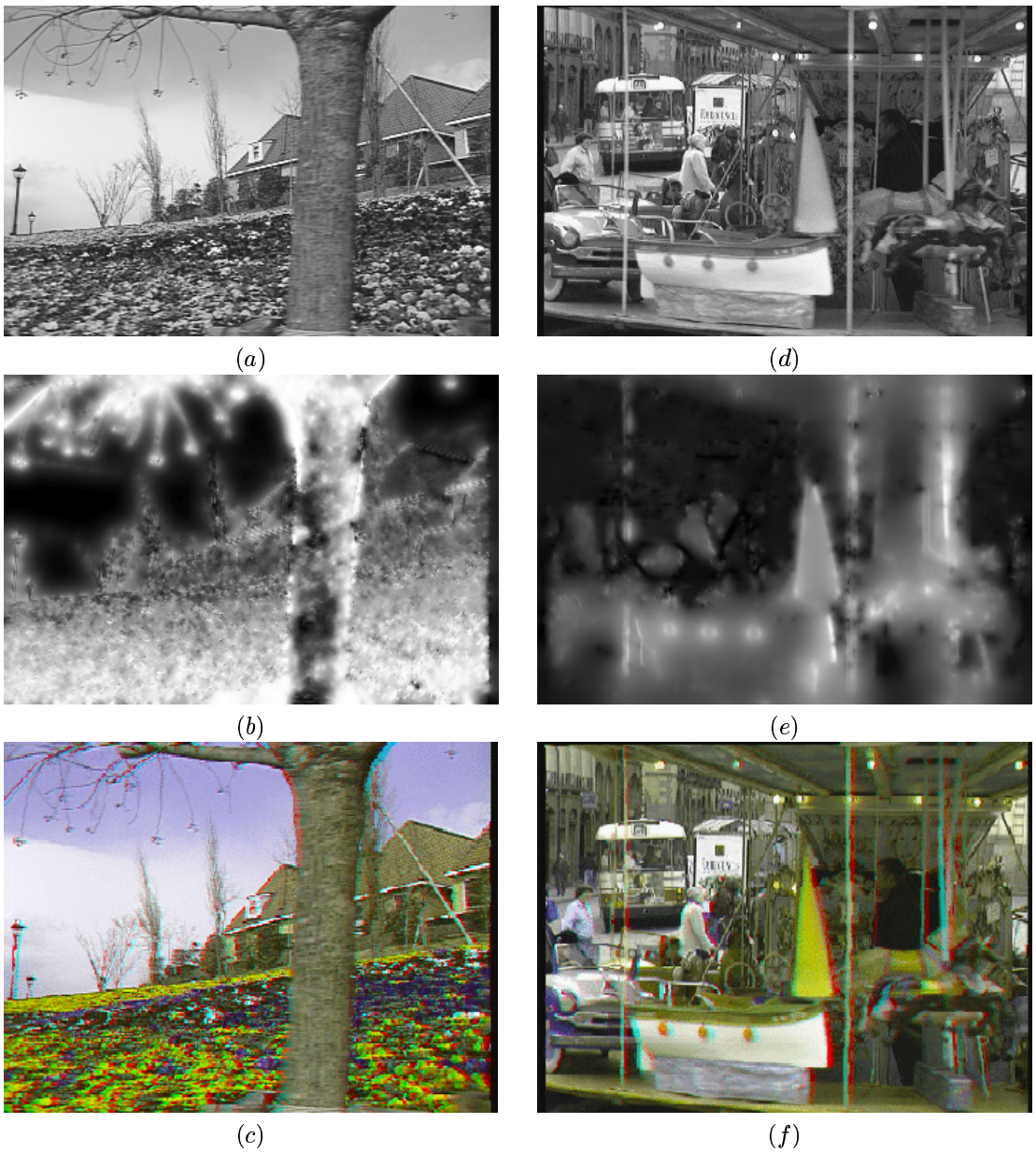

Figure 3. A frame of the Flower-garden sequence (a) and the corresponding grey level rendering of the estimated depth (b) and constructed color anaglyph of stereoscopic images (must be viewed with red/blue glasses)(c). The same representations are given for the sequence Manege (d) in (e) and f

[14] P. Torr, R. Szeiliski, and P. Anandan. An integrated bayesian approach to layer extraction from image sequences. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(3):297–303, 2001.