Motion displacement estimation using an affine model for image matching

Chiou-Shann Fuh
Petros Maragos
Harvard University
Division of Applied Sciences
Cambridge, Massachusetts 02138

Abstract. A model is developed for estimating the displacement field in spatio-temporal image sequences that allows for affine shape deformations of corresponding spatial regions and for affine transformations of the image intensity range. This model includes the block matching method as a special case. The model parameters are found by using a least-squares algorithm. We demonstrate experimentally that the affine matching algorithm performs better in estimating displacements than other standard approaches, especially for long-range motion with possible changes in scene illumination. The algorithm is successfully applied to various classes of moving imagery, including the tracking of cloud motion.

Subject terms: visual communications; image motion analysis; displacement estimation; block matching; affine models.

Optical Engineering 30(7), 881–887 (July 1991).

CONTENTS

1. INTRODUCTION

Motion detection is a very important problem both in video image coding and in computer vision. In video coding, motion detection is a necessary task for motion-compensated predictive coding and motion-adaptive frame interpolation to reduce the required channel bandwidth. In computer vision systems, motion detection can be used to infer the 3-D motion and surface structure of moving objects with many applications to robot guidance and remote sensing.

Let $I(x,y,t)$ be a spatio-temporal intensity image signal due to a moving object, where $\mathbf{p} = (x,y)$ is the (spatial) pixel vector. A well-known approach to estimating 2-D velocities or pixel displacements on the image plane is the standard *block matching* method, where

$$E(\mathbf{d}) = \sum_{\mathbf{p}\in R} |I(\mathbf{p},t_1) - I(\mathbf{p} + \mathbf{d},t_2)|^2 \qquad (1)$$

is minimized over a small spatial region R to find the optimum *displacement vector* $\mathbf{d}$. Minimizing $E(\mathbf{d})$ is closely related to finding $\mathbf{d}$ such that the correlation $\Sigma_{\mathbf{p}\in R} I(\mathbf{p},t_1)I(\mathbf{p}+\mathbf{d},t_2)$ is maximized; thus, this approach is sometimes called the *area correlation* method. This approach has been criticized because (1) the method is computation-intensive; (2) the method ignores that the region R, which is the projection of the moving object at time $t = t_1$, will correspond to another region R' at $t = t_2$ with deformed shape due to foreshortening of the object surface regions as viewed at two different time instances; and (3) the image signals corresponding to regions R and R' do not only differ with respect to their supports R and R', but also undergo amplitude transformations due to the different lighting and viewing geometries at t_1 and t_2. Nowadays, problem (1) is not critical anymore due to the availability of very fast hardware or parallel computers, but problems (2) and (3) are serious drawbacks. Several researchers have adopted other methods that depend either on (a) constraints among spatio-temporal image gradients or on (b) tracking features (e.g., edges, blobs). However, approach (a) performs badly for medium- or long-range motion and is sensitive to noise. Approach (b) is more robust in noise and works for longer range motion, but feature extraction and tracking is a difficult task and gives sparse motion estimates. By comparison, if problems (2) and (3) can be solved, then the block matching method has the advantages of more robustness than approach (a) and denser motion estimates than approach (b).

After a brief overview of related literature in Sec. 2, we present in Sec. 3 an improved model for block matching that solves problems (2) and (3) by allowing R to undergo affine shape deformations (as opposed to just translations that the block matching method assumes) and by allowing the intensity signal I to undergo affine amplitude transformations. Section 4 provides a least-squares algorithm to find the parameters of this affine model. Then, several experiments are reported in Sec. 5 that demonstrate the superiority of our affine model for image matching and motion detection over other standard approaches.

2. BACKGROUND

There is vast literature on motion detection. Some reviews on this topic include Refs. 1 through 3. Here, we briefly survey a few sample works that contain elements related to our work. The major approaches to computing displacement vectors for corresponding pixels in two time-consecutive image frames can

Invited paper VC-102 received Dec. 24, 1990; revised manuscript received March 11, 1991; accepted for publication March 13, 1991.
©1991 Society of Photo-Optical Instrumentation Engineers.

Downloaded from SPIE Digital Library on 18 Dec 2009 to 140.112.31.82. Terms of Use: http://spiedl.org/terms

be classified as using gradient-based methods, correspondence of motion tokens, or block matching methods.

The gradient-based methods are based on some relationships among the image spatial and temporal derivatives. For example, Horn and Schunck[4] used the optical flow constraint $dI/dt = 0 \Leftrightarrow (\partial I/\partial x)v_x + (\partial I/\partial y)v_y = -\partial I/\partial t$, where v_x, v_y are the x,y velocity components. Cornelius and Kanade[5] extended Horn and Schunck's work to allow for gradual changes in the moving object's appearance and for flow discontinuities at object boundaries. Brockett[6] developed a least-squares approach to approximate optical flow by affine vector fields using shape gramians. A broad class of gradient-based methods are all the pixel-recursive algorithms, popular among video coding researchers. Netravali and Robbins[7] developed a pixel-recursive algorithm to improve the estimation accuracy and to increase the measuring range of displacement. Stuller et al.[8] proposed a gradient search technique for estimating displacement and a luminance change gain. Caffforio and Rocca[9] proposed some improvements on pixel-recursive estimation algorithms. Biemond et al.[10] developed a pixel-recursive algorithm for the estimation of rotation and translation parameters in consecutive image frames. Kalivas et al.[11] proposed two algorithms to estimate the parameters of a 2-D affine motion model; one is based on Taylor series expansion and assumes smooth spatial variation of intensities and the other is a steepest descent algorithm. In general, the gradient methods are analytically tractable and they often make use of iterative solutions. The methods can also give dense displacement estimates, i.e., a displacement vector for each pixel. However, because the methods require derivatives, their use is limited to short-range motion, i.e., at most 2 to 3 pixels. To achieve longer range displacement estimation, multiple resolution gradient methods can be used, but this increases their computational complexity. The derivatives in discrete domain are usually approximated by differences, which introduce errors. In addition, differentiation amplifies high-frequency components and thus the method can be very sensitive to noise.

Another class of commonly used motion analysis methods is the correspondence of motion tokens, where important image features are extracted and tracked over consecutive image frames. Various types of tokens can be used, such as isolated points, edges, and blobs. As an example of point tokens, Tsai and Huang[12] used seven correspondence point pairs to determine 3-D motion parameters of curved surfaces from 2-D perspective views. Lee[13] developed an algorithm to recover 2-D affine transformations of planar objects by using moments to find invariant axes. Costa et al.[14] proposed an approach to deal with affine-invariant point matching in the presence of noise. As an example of blob tokens, Fuh and Maragos[15] developed a region matching method where blob-like regions corresponding to intensity peaks and valleys are extracted at each frame and tracked over time. In general, correspondence methods can usually achieve medium or longer range displacement estimates than gradient methods, but they usually give only sparse estimates. They are more robust in the presence of noise, but the correspondence problem is difficult to solve.

In block matching methods, blocks (or subframes) in the previous frame are matched with corresponding blocks in the current frame via criteria such as minimizing a mean-squared (or absolute) error or maximizing a cross-correlation. For example, Jain and Jain[16] proposed a mean-squared error block matching algorithm for estimating interframe displacement of small blocks. Tzou et al.[17] proposed an iterative block matching algorithm, which showed better performance than conventional algorithms to estimate both the displacement and the amplitude ratio. Gilge[18] developed fast algorithms both for motion estimation (by using vector quantization techniques) and for illumination correction (by modeling changes with an additive bias). Finally, Skifstad and Jain[19] presented methods for detecting scene changes in moving imagery with varying illumination.

3. AFFINE MODEL FOR IMAGE MATCHING

We assume that the region R' at $t = t_2$ has resulted from the region R at $t = t_1$ via an affine shape deformation $\mathbf{p} \rightarrow \mathbf{Mp} + \mathbf{d}$, where

$$\mathbf{Mp} + \mathbf{d} = \begin{bmatrix} s_x \cos\theta_x & -s_y \sin\theta_y \\ s_x \sin\theta_x & s_y \cos\theta_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} d_x \\ d_y \end{bmatrix} . \tag{2}$$

The vector $\mathbf{d} = (d_x,d_y)$ accounts for spatial translations, whereas the 2×2 real matrix $\mathbf{M}$ accounts for rotations and scalings (compressions or expansions). That is, s_x,s_y are the scaling ratios in the x,y directions, and θ_x,θ_y are the corresponding rotation angles. These kinds of region deformations occur in a moving image sequence. For example, when objects rotate relative to the camera, the region R also rotates. When objects move closer or farther from the camera, the region R gets scaled (expanded or compressed). Displacements by $\mathbf{d}$ can be caused by translations of objects parallel to the image plane as well as by rotations. In addition, we allow the image intensities to undergo an affine transformation $I \rightarrow rI + c$, where the ratio r adjusts the image amplitude dynamic range and c is the brightness offset. These intensity changes can be caused by different lighting and viewing geometries at times t_1 and t_2.

Thus, given $I(x,y,t)$ at $t = t_1,t_2$, and at various image locations, we select a small analysis region R and find the optimal parameters $\mathbf{M}$, $\mathbf{d}$, r, c that minimize the error functional

$$E(\mathbf{M},\mathbf{d},r,c) = \sum_{\mathbf{p}\in R} |I(\mathbf{p},t_1) - rI(\mathbf{Mp} + \mathbf{d},t_2) - c|^2 . \tag{3}$$

The optimum $\mathbf{d}$ provides us with the displacement vector. As by-products, we also obtain the optimal $\mathbf{M}$, r, c, which provide information about rotation, scaling, and intensity changes. We call this approach the *affine model for image matching*. Note that the standard block matching method is a special case of our affine model, corresponding to an identity matrix $\mathbf{M}$, $r = 1$, $c = 0$. Although $\mathbf{d}$ is a displacement vector representative for the whole region R, we can obtain dense displacement estimates by repeating this minimization procedure at each pixel, with R being a small surrounding region. Note that if R is a square region, its corresponding region R' under the map $\mathbf{p} \rightarrow \mathbf{Mp} + \mathbf{d}$ will generally be a rotated and translated parallellogram. More general shape/intensity transformations can be modeled by a *sum* of affine maps, i.e., $I(\mathbf{p},t_1) \rightarrow c + \Sigma_n r_n I(\mathbf{M}_n\mathbf{p} + \mathbf{d}_n,t_2)$, as developed in Ref. 20.

4. LEAST-SQUARES ALGORITHM

Finding the optimal $\mathbf{M}$, $\mathbf{d}$, r, c is a nonlinear optimization problem. While the problem can be solved iteratively by gradient steepest descent in an 8-D parameter space, this approach cannot guarantee convergence to a global minimum. Alternatively, in our work we propose the following algorithm that provides a

Downloaded from SPIE Digital Library on 18 Dec 2009 to 140.112.31.82. Terms of Use: http://spiedl.org/terms

closed-form solution for the optimal r,c and iteratively searches a quantized parameter space for the optimal $\mathbf{M},\mathbf{d}$. We find first the optimal r,c by setting

$$\frac{\partial E}{\partial r} = -\sum_{\mathbf{p}\in R} I(\mathbf{Mp}+\mathbf{d},t_2)I(\mathbf{p},t_1) + r\sum_{\mathbf{p}\in R} I^2(\mathbf{Mp}+\mathbf{d},t_2) + c\sum_{\mathbf{p}\in R} I(\mathbf{Mp}+\mathbf{d},t_2) = 0\ , \tag{4}$$

$$\frac{\partial E}{\partial c} = -\sum_{\mathbf{p}\in R} I(\mathbf{p},t_1) + r\sum_{\mathbf{p}\in R} I(\mathbf{Mp}+\mathbf{d},t_2) + c\sum_{\mathbf{p}\in R} 1 = 0\ . \tag{5}$$

Solving these two linear equations yields the optimal r^* and c^* as functions of $\mathbf{M}$ and $\mathbf{d}$:

$$r^*(\mathbf{M},\mathbf{d}) = \frac{A\sum_{\mathbf{p}\in R} I(\mathbf{Mp}+\mathbf{d},t_2)I(\mathbf{p},t_1) - \sum_{\mathbf{p}\in R} I(\mathbf{Mp}+\mathbf{d},t_2)\sum_{\mathbf{p}\in R} I(\mathbf{p},t_1)}{A\sum_{\mathbf{p}\in R} I^2(\mathbf{Mp}+\mathbf{d},t_2) - \left[\sum_{\mathbf{p}\in R} I(\mathbf{Mp}+\mathbf{d},t_2)\right]^2}\ , \tag{6}$$

$$c^*(\mathbf{M},\mathbf{d}) = \frac{1}{A}\left[\sum_{\mathbf{p}\in R} I(\mathbf{p},t_1) - r^*(\mathbf{M},\mathbf{d})\sum_{\mathbf{p}\in R} I(\mathbf{Mp}+\mathbf{d},t_2)\right]\ , \tag{7}$$

where A is the area of the region R. Replacing the optimal r^*,c^* into E yields the error functional

$$E^*(\mathbf{M},\mathbf{d}) = E(\mathbf{M},\mathbf{d},r^*,c^*) \tag{8}$$

$$= \sum_{\mathbf{p}\in R} |I(\mathbf{p},t_1) - r^*I(\mathbf{Mp}+\mathbf{d},t_2) - c^*|^2$$

$$= \sum_{\mathbf{p}\in R} I^2(\mathbf{p},t_1) - r^*\sum_{\mathbf{p}\in R} I(\mathbf{p},t_1)I(\mathbf{Mp}+\mathbf{d},t_2) - c^*\sum_{\mathbf{p}\in R} I(\mathbf{p},t_1) \tag{9}$$

$$= \sum_{\mathbf{p}\in R} I_1^2 - \frac{A\left(\sum_{\mathbf{p}\in R} I_1I_2\right)^2 + \sum_{\mathbf{p}\in R} I_2^2\left(\sum_{\mathbf{p}\in R} I_1\right)^2 - 2\sum_{\mathbf{p}\in R} I_1\sum_{\mathbf{p}\in R} I_2\sum_{\mathbf{p}\in R} I_1I_2}{A\sum_{\mathbf{p}\in R} I_2^2 - \left(\sum_{\mathbf{p}\in R} I_2\right)^2}\ , \tag{10}$$

where $I_1 = I(\mathbf{p},t_1)$ and $I_2 = I(\mathbf{Mp}+\mathbf{d},t_2)$. Since the term $\Sigma_{\mathbf{p}}I_1^2$ is independent of $\mathbf{M}$ and $\mathbf{d}$, minimizing $E^*(\mathbf{M},\mathbf{d})$ is equivalent to maximizing the function

$$K(\mathbf{M},\mathbf{d}) = \frac{A\left(\sum_{\mathbf{p}\in R} I_1I_2\right)^2 + \sum_{\mathbf{p}\in R} I_2^2\left(\sum_{\mathbf{p}\in R} I_1\right)^2 - 2\sum_{\mathbf{p}\in R} I_1\sum_{\mathbf{p}\in R} I_2\sum_{\mathbf{p}\in R} I_1I_2}{A\sum_{\mathbf{p}\in R} I_2^2 - \left(\sum_{\mathbf{p}\in R} I_2\right)^2}\ . \tag{11}$$

The function $K(\mathbf{M},\mathbf{d})$ consists of several correlation terms. Now, by discretizing the 6-D parameter space $\mathbf{M},\mathbf{d}$ and exhaustively searching a bounded region, we find the optimal $\mathbf{M},\mathbf{d}$ that maximize $K(\mathbf{M},\mathbf{d})$. (The 2-D parameter subspace $\mathbf{d}$ is inherently discrete because it represents integer pixel coordinates.) After having found the optimal $\mathbf{M}$ and $\mathbf{d}$, we can obtain the optimal r and c from the known functions $r^*(\mathbf{M},\mathbf{d})$ and $c^*(\mathbf{M},\mathbf{d})$.

In our implementation of the above algorithm we select the image domain regions R to be overlapping squares of size $B \times B$ pixels. (In this paper we set $B = 21$.) The centers of these region blocks form a uniform square grid of $G \times G$ points. The optimum displacement $\mathbf{d}$ is estimated at these region centers. Here, G controls the spatial frequency of estimated displacements. To avoid aliasing, and because we are implicitly using a 2-D rectangular window for our short-space analysis, the distance between two consecutive region centers should not exceed $B/2$ (in each direction). Further, we constrain the action of M so that it performs a uniform rotation by $\theta = \theta_x = \theta_y$ and uniform scaling by $s = s_x = s_y$. We also constrain $\mathbf{d} = (d_x,d_y)$ to be within an $L \times L$ window around $\mathbf{p}$, where $L/2$ is the maximum expected displacement in each direction. To find the optimum scaling s, we discretize and bound its parameter space by searching the finite range between 1 and $\pm$ the maximum scale deviation from unity (which depends upon the specific application) at steps of size 0.1. Similarly, we find the optimum rotation angle θ by bounding its range between 0 and $\pm$ a maximum angle and by searching at steps of 2 deg. For each region, the rotation and scaling are implemented locally by setting their centers at the region center. Thus, overall we search in a bounded finite discrete 4-D parameter space s,θ,d_x,d_y. Finally note that, if $\mathbf{p}$ is an integer pixel vector in R, the vector $\mathbf{p}' = \mathbf{Mp} + \mathbf{d}$ will generally have real-valued coordinates due to the rotation and scaling induced by $\mathbf{M}$. Hence, to be able to assign an intensity value at the location $\mathbf{p}'$ we do bilinear interpolation of the four neighbors of $\mathbf{p}'$ that have integer pixel coordinates.

5. EXPERIMENTS

In this section we describe several experiments that apply the above affine model and least-squares algorithm to 2-D motion detection. Figures 1(a) and 1(b) show an original poster image and a synthetically transformed image according to the affine model with a global translation of $\mathbf{d} = (5,5)$ pixels, rotation by $\theta = 6$ deg, scaling $s = 1.2$, intensity ratio $r = 0.7$, and intensity bias $c = 20$. The center of the synthesized rotation and scaling is at the global center of the image. Figures 1(c) and 1(d) show that the displacement field estimated via the affine matching algorithm (with the maximum scaling and rotation set at 1.2 and 6 deg) is much more robust than that estimated via the standard block matching. Table 1 lists the average values and standard deviations of the recovered affine model parameters and of the displacement estimation errors (in pixels). The averaging was done over $G^2 = 256$ blocks. (Note: due to the global rotation and scaling with respect to the image center, the displacement is not constant over each analysis region.) The numerical results of Table 1 provide evidence about the efficacy of our algorithm to estimate affine changes in image motion and illumination.

As a real motion example, Fig. 2(a) shows a poster image under dim light source, whereas Fig. 2(b) shows the same poster after a small rotation and under much brighter light sources. The scene changes between the images in Figs. 2(a) and 2(b) were induced by physically moving the digitizing camera and changing the scene illumination. As Fig. 2(c) shows, the standard block matching (without affine shape deformation and affine intensity transformation) can result in too many incorrect dis-

Downloaded from SPIE Digital Library on 18 Dec 2009 to 140.112.31.82. Terms of Use: http://spiedl.org/terms

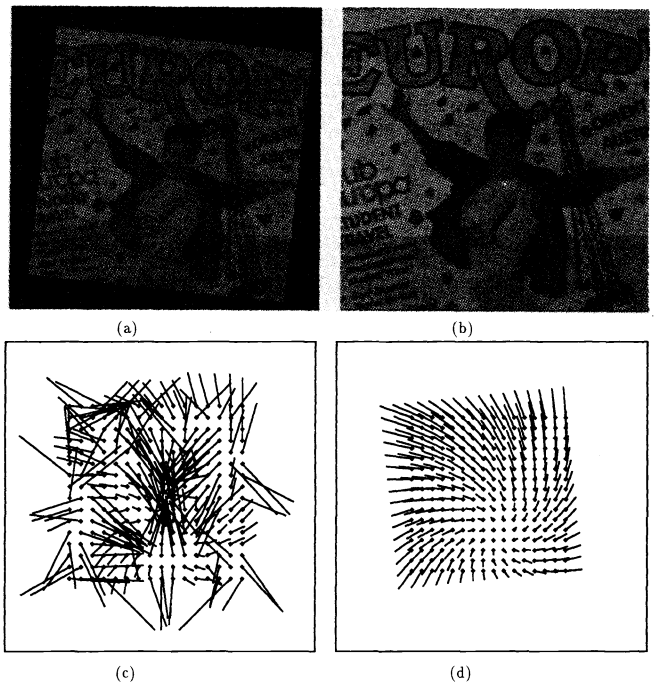


Fig. 1. (a) An affine transformed version of the image in (b) with translation d = (5,5), rotation θ = 6 deg, scaling s = 1.2, intensity ratio r = 0.7, and intensity bias c = 20; (b) the original poster image (242 × 242 pixels, 8 bpp); (c) displacement vectors between the images in (a) and (b) obtained from standard block matching; and (d) displacement vectors from the affine matching algorithm (L = 80 pixels).

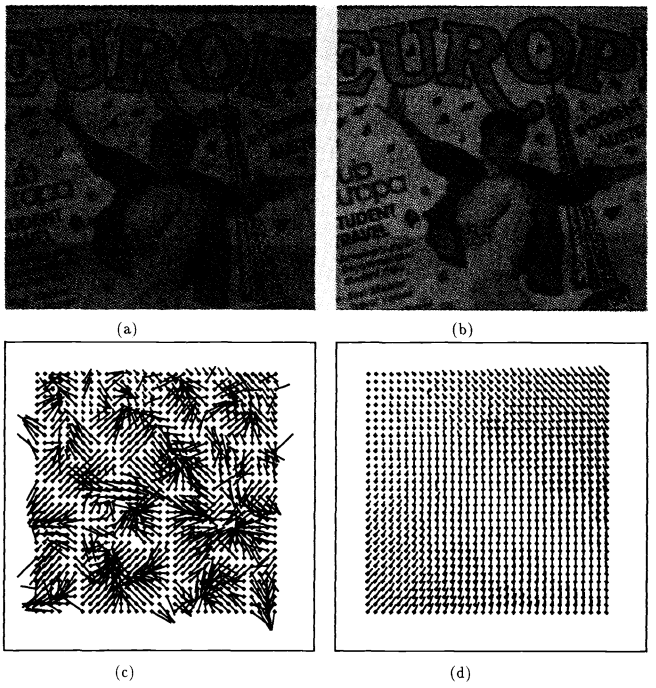


Fig. 2. (a) First frame of a poster image sequence under dim light sources (242 × 242 pixels, 8 bpp); (b) second frame of poster with small rotation and under much brighter light sources; (c) displacement vectors from standard block matching; and (d) displacement vectors from the affine matching algorithm (L = 30 pixels).

placement vectors because every block in Fig. 2(a) tends to match with the dark areas in Fig. 2(b). Figure 2(d) demonstrates the good performance of the affine matching algorithm on Figs. 2(a) and 2(b).

The goal of Figs. 3 and 4 is to compare the affine matching algorithm with other approaches. Figure 3(a) shows the poster image of Fig. 2(b) after the camera zoomed in by moving forward from being 150 cm away to 100 cm (with focus readjusted); hence, the poster image expands. Figure 3(b) shows the same poster image rotated about 23 deg counterclockwise. Thus, Figs. 2(b), 3(a), and 3(b) are frames from a moving image sequence consisting of translation followed by rotation. Figure 3(c) shows that the standard block matching of Figs. 2(b) and 3(a) gives several errors in estimating displacements. Much better is the result of applying the affine matching algorithm, shown in Fig. 3(d), to track the motion between Figs. 2(b) and 3(a). Figure 4 shows the result of estimating the displacement field between Figs. 3(a) and 3(b) by using (a) the standard block matching, (b) the affine matching algorithm, (c) a feature-based displacement estimation algorithm,[15] and (d) a gradient-based optical flow algorithm.[4]

Clearly, the affine matching algorithm has the best performance. However, the superior performance of our affine model comes at a high computational complexity. To quantify this

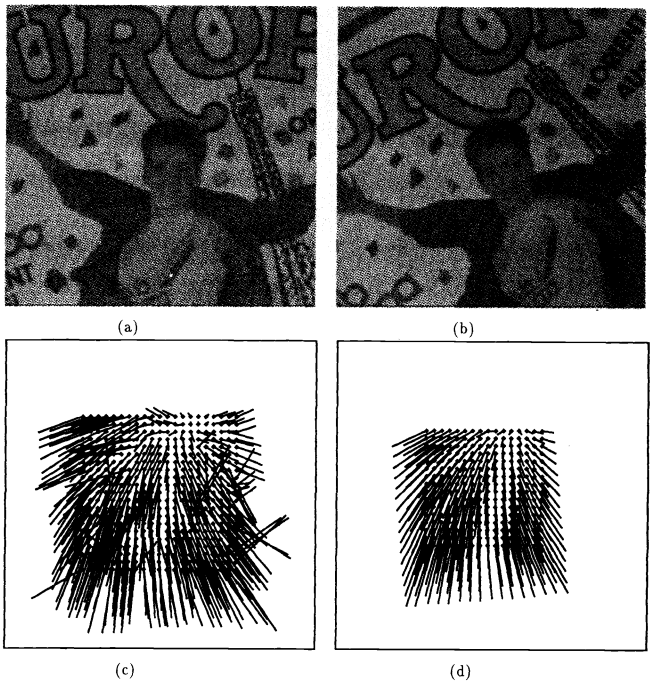


Fig. 3. (a) Third frame of poster after camera moved closer to the object; (b) fourth frame of poster after a 23-deg counterclockwise rotation; (c) displacement vectors from standard block matching of images in Figs. 2(b) and 3(a); and (d) displacement vectors from affine matching algorithm (L = 100 pixels).

Table 1. Recovered affine model parameters.

	Scaling s	Rotation θ	Bias c	Ratio r	d_x error	d_y error
Correct	1.2	6.0	20.0	0.7	0	0
Average	1.1988	5.7500	20.4151	0.6902	0.2706	0.2762
St. Dev.	0.0108	0.6847	2.0627	0.0160	0.1671	0.2405

complexity, let the image have height H pixels and width W pixels. Let also N be the number of iterations required by the gradient algorithm in Ref. 4 and let the impulse response of the bandpass filter used in Ref. 15 for region extraction be $K \times K$

Downloaded from SPIE Digital Library on 18 Dec 2009 to 140.112.31.82. Terms of Use: http://spiedl.org/terms

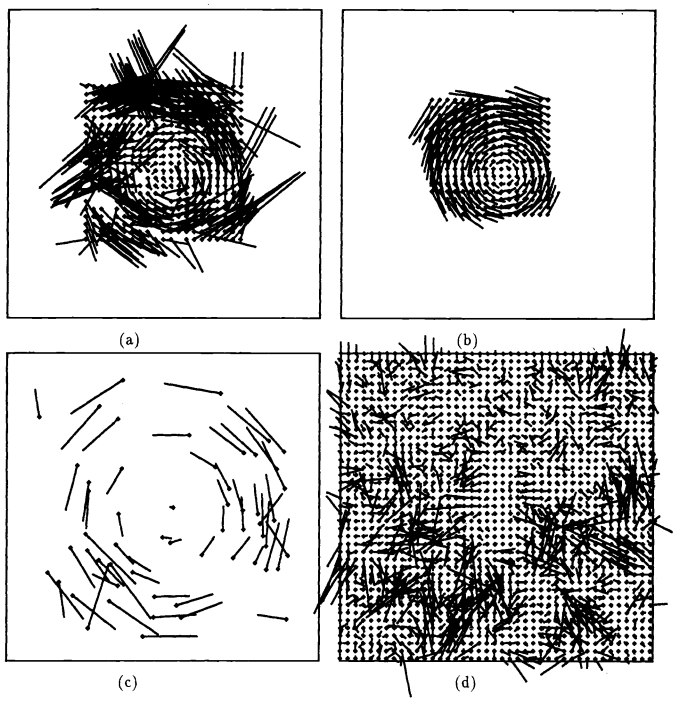


Fig. 4. Displacement vectors between images in Figs. 3(a) and 3(b) from four approaches (L = 100 pixels); (a) standard block matching; (b) affine matching algorithm; (c) a feature-based displacement estimation algorithm[15]; and (d) a gradient-based optical flow algorithm.[4]

pixels. (In Fig. 4, N = 256 and K = 21.) Then, Table 2 lists the computational complexity of the four algorithms compared in Fig. 4. The quantities in Table 2 express the order of magnitude of the number of required operations (multiplications/additions); the multiplicative constant factors involved in these orders of magnitude are different for each algorithm. We have implemented the affine matching algorithm on a parallel computer (MasPar with 1024 processors), and the execution time has been reduced by a ratio of about 20:1 compared with a serial computer (SUN4).

Although the displacement estimates from the affine matching algorithm are mostly robust, there may be a few mismatches, which we view as noise. In this case, additional improvement can be achieved by smoothing the displacement vector field. We exclude the use of linear filtering (e.g., local averaging) because linear smoothing filters have the well-known tendency to blur and shift sharp discontinuities in signals. Sharp discontinuities in the displacement field may indicate object boundaries and, hence, must be preserved. Instead, we chose spatio-temporal vector median filtering because the scalar median filter can eliminate outliers while preserving abrupt edges. Vector median filtering is defined as the x,y componentwise median filtering: $\text{med}\{\mathbf{d}_i\} = (\text{med}\{\mathbf{d}_{x,i}\}, \text{med}\{\mathbf{d}_{y,i}\})$, where $\mathbf{d}_i$, $i = 1,2,\ldots,n$, are the displacement vectors in a spatio-temporal cube surrounding the center of region R and time t_1. We have found this vector median to perform well in smoothing velocity fields; see also Ref. 15. (For a recent theoretical analysis of the vector median see Ref. 21.)

Table 2. Computational complexity of four algorithms; B = block size, G = estimation grid size, L = displacement search window size, S = number of scaling search points, θ = number of rotation search points, W = image width, H = image height, N = number of iterations, and K = filter size.

Standard block matching	$O(B^2 \cdot L^2 \cdot G^2)$
Affine model matching	$O(B^2 \cdot L^2 \cdot G^2 \cdot S \cdot \Theta)$
Region correspondence algorithm [8]	$O(H \cdot W \cdot K^2)$
Gradient optical flow algorithm [11]	$O(H \cdot W \cdot N)$

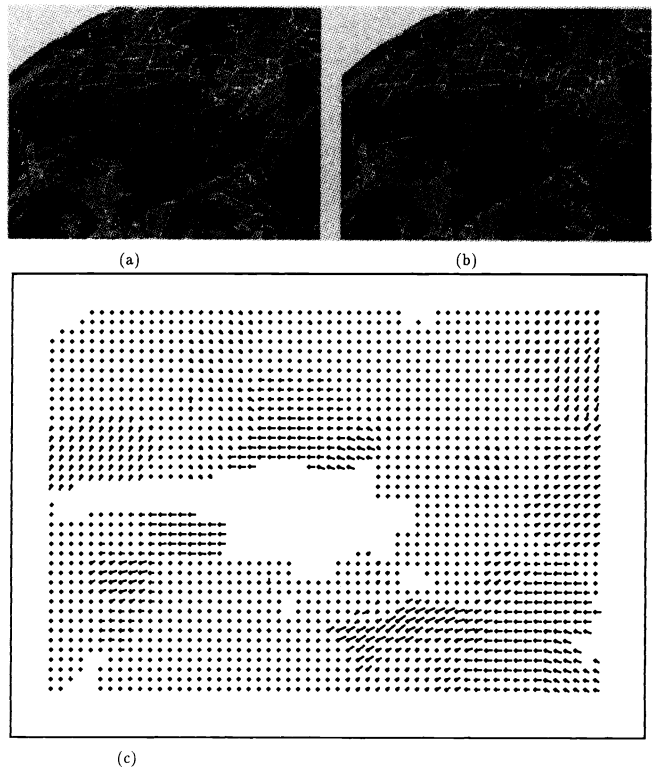


Fig. 5. (a) First frame of an infrared cloud image sequence (240 × 320 pixels, 4 bpp where intensity of each pixel is the altitude of the cloud top); (b) second frame of the cloud sequence (30 min between frames); and (c) displacement vectors (magnified 1.5 times) from the affine matching algorithm, smoothed by a spatio-temporal vector median filter (L = 30 pixels).

Our affine matching algorithm not only performs well on rigid objects undergoing short- or long-range motion and/or changes in scene lighting, but also has satisfactory performance on nonrigid objects, such as moving clouds where the interframe changes of object shapes could be very large. Figures 5(a) and 5(b) show two time frames from a satellite infrared cloud image sequence. Figure 5(c) shows the respective motion displacement field $\mathbf{d}$ that resulted by applying the above affine matching algorithm and smoothing the raw estimates by a spatio-temporal vector median filter. We have applied the affine matching algorithm followed by vector median smoothing to several moving sequences of cloud imagery with an equally good success as in Fig. 5. For example, Fig. 6 shows the same type of motion tracking system applied to a moving cloud sequence obtained during a hurricane; here, the motion is more rapid and inhomogeneous across the image.

6. CONCLUSIONS

We have developed an affine model and a corresponding least-squares algorithm for image matching that shows good performance in estimating 2-D motion for a variety of moving imagery,

Downloaded from SPIE Digital Library on 18 Dec 2009 to 140.112.31.82. Terms of Use: http://spiedl.org/terms

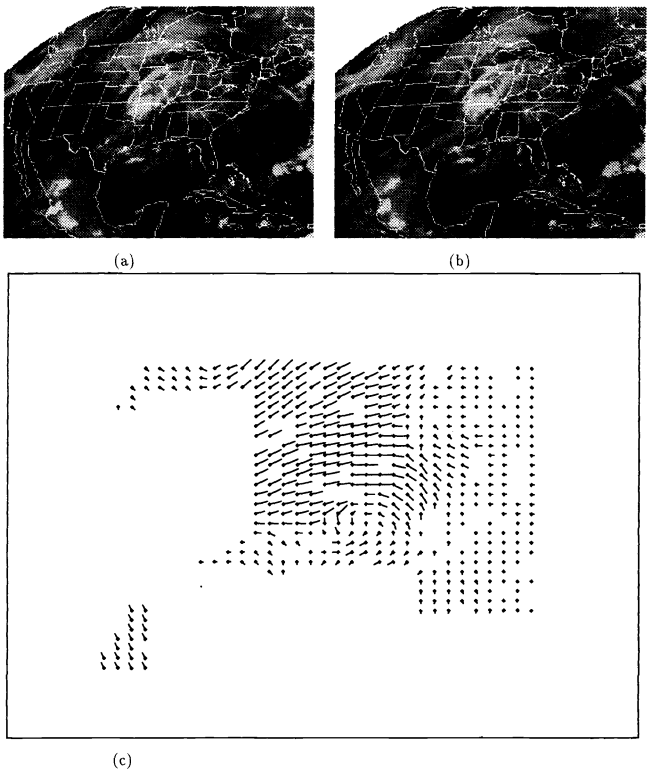


Fig. 6. (a) First frame of a satellite infrared cloud image sequence from a hurricane (240 × 320 pixels, 4 bpp); (b) second frame of the hurricane sequence (30 min between frames); and (c) displacement vectors (magnified 1.5 times) from the affine matching algorithm, smoothed by a vector median filter (L = 60 pixels).

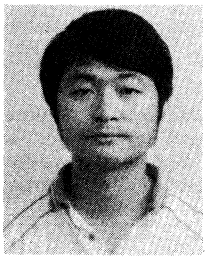
e.g., indoor pictures, outdoor scenes, and clouds. In terms of robust estimation of displacements, the approach outperforms other conventional methods based either on block matching, gradient methods, or on feature tracking, especially for long-range motion and/or illumination changes. However, our method has a somewhat higher computational complexity; in the present day, this no longer presents a problem due to the availability of very fast hardware and parallel computers. Post-smoothing the velocity field via spatio-temporal vector median filtering almost always improves the performance of the matching algorithm. The resulting displacement vectors can also be used as input data to various 3-D models that can provide estimates of the 3-D motion and depth parameters of moving objects.

7. ACKNOWLEDGMENT

This research work was supported by the NSF under Grant MIPS-86-58150 with matching funds from Bellcore, DEC, TASC, and Xerox, and in part by the ARO under Grant DAALO3-86-K-0171 to the Center for Intelligent Control Systems.

8. REFERENCES

1. J. K. Aggarwal and N. Nandhakumar, "On the computation of motion from sequences of images—a review," *Proc. IEEE* 76, 917–935 (Aug. 1988).
2. E. C. Hildreth and C. Koch, "The analysis of visual motion: from computational theory to neuronal mechanisms," A.I.L. memo 919, M.I.T. (1986).
3. H. G. Musmann, P. Pirsch, and H.-J. Grallert, "Advances in picture coding," *Proc. IEEE*, Vol. 73, 523–548 (1985).
4. B. K. P. Horn and B. G. Schunck, "Determining optical flow," in *Artificial Intelligence*, Vol. 17, 185–203 (1981).
5. N. Cornelius and T. Kanade, "Adapting optical flow to measure object motion in reflectance and x-ray image sequences," in *Motion Representation and Perception*, N. I. Badler and J. K. Tsotsas, Eds., ACM (1983).
6. R. Brockett, "Gramians, generalized inverses, and the least-squares approximation of optical flow," *J. Visual Commun. Image Repres.* 1(9), 3–11 (1990).
7. A. N. Netravali and J. D. Robbins, "Motion compensated television coding—part I," *Bell Syst. Tech. J.* 58(3), 631–670 (1979).
8. J. A. Stuller, A. N. Netravali, and J. D. Robbins, "Interframe television coding using gain and displacement compensation," *Bell Syst. Tech. J.* 59(9), 1227–1240 (1980).
9. C. Cafforio and F. Rocca, "The differential method for image motion estimation," in *Image Sequence Analysis*, T. S. Huang, Ed., Springer-Verlag, pp. 104–124 (1983).
10. J. Biemond, J. N. Driessen, A. M. Geurtz, and D. E. Boekee, "A pel-recursive Wiener-based algorithm for the simultaneous estimation of rotation and translation," in *Visual Communications and Image Processing*, T. R. Hsing, Ed., Proc. SPIE 1001, 917–924 (1988).
11. D. S. Kalivas, A. A. Sawchuk, and R. Chellappa, "Segmentation and 2-D motion estimation of noisy image sequences," in *Proc. IEEE Intl. Conf. Acoust., Speech, Signal Process.*, pp. 1076–1079, New York (1988).
12. R. Y. Tsai and T. S. Huang, "Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surfaces," *IEEE Trans. Pattern Anal. Mach. Intellig.* PAMI-6(1), 13–27 (1984).
13. M. Lee, "Recovering the affine transformation of images by using moments and invariant axes," in *Image Understanding and Machine Vision*, Technical Digest Series, Vol. 14, 2–5, Washington, D.C., Optical Society of America (1989).
14. M. S. Costa, R. M. Haralick, and L. G. Shapiro, "Optimal affine invariant point matching," in *Proc. IEEE Intl. Conf. on Pattern Recognition*, pp. 233–236 (1990).
15. C. S. Fuh and P. Maragos, "Region-based optical flow estimation," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 130–135 (1989).
16. J. R. Jain and A. K. Jain, "Displacement measurement and its application in interframe coding," *IEEE Trans. Commun.* COM-29(12), 1799–1808 (1981).
17. K. H. Tzou, T. R. Hsing, and N. A. Daly, "Block-recursive matching algorithm (BRMA) for displacement estimation of video images," in *Proc. IEEE Intl. Conf. Acoust., Speech, Signal Process.*, pp. 359–362 (1985).
18. M. Gilge, "Motion estimation by scene adaptive block matching (SABM) and illumination correction," in *Image Processing Algorithms and Techniques*, Proc. SPIE 1244, 355–366 (1990).
19. K. Skifstad and R. Jain, "Illumination independent change detection for real-world image sequences," in *Computer Vision, Graphics, and Image Processing*, Vol. 46, 387–399 (1989).
20. P. Maragos, "Affine morphology and affine signal models," in *Image Algebra and Morphological Image Processing*, P. D. Gader, Ed., Proc. SPIE 1350, 31–43 (1990).
21. J. Astola, P. Haavisto, and Y. Neuvo, "Vector median filters," *Proc. IEEE*, 78, 678–689 (April 1990).



Chiou-Shann Fuh received the BS degree in information engineering from the National Taiwan University, Taiwan, China, in 1983, the MS degree in computer science from the Pennsylvania State University, University Park, in 1987, and the MS degree in applied sciences from Harvard University, Cambridge, in 1989. From 1983 to 1985, he was a second lieutenant communication officer in the Taiwan Air Force. From 1985 to 1987, he was a research assistant at the VLSI Lab, Department of Computer Science, the Pennsylvania State University. Since 1987, he has been a research assistant at the Robotics Lab, Division of Applied Sciences, Harvard University, where he is currently working toward a Ph.D. degree. His current research interests include motion analysis of computer vision and mathematical morphology.

Downloaded from SPIE Digital Library on 18 Dec 2009 to 140.112.31.82. Terms of Use: http://spiedl.org/terms

Petros Maragos received the Diploma degree in electrical engineering from the National Technical University of Athens, Greece, in 1980, and the M.S.E.E. and Ph.D. degrees from the Georgia Institute of Technology, Atlanta, in 1982 and 1985, respectively. In 1985, he joined the faculty of the Division of Applied Sciences at Harvard University, Cambridge, where he is currently an associate professor of electrical engineering. His general research interests are in signal processing and its applications to computer vision and computer speech. Some of his current research focuses on morphological signal processing, fractal signal/image analysis, and nonlinear modeling of speech production. He is currently serving as an associate editor for the IEEE Transactions on Signal Processing, and on the editorial board for the Journal of Visual Communication and Image Representation. He received a Sigma Xi research award in 1983; a National Science Foundation Presidential Young Investigator Award in 1987; and the IEEE Acoustics, Speech, and Signal Processing Society's 1988 Paper Award for a publication in the Society's Transactions.

Downloaded from SPIE Digital Library on 18 Dec 2009 to 140.112.31.82. Terms of Use: http://spiedl.org/terms