PAPER

Global Motion Parameter Extraction and Deformable Block Motion Estimation

Chi-Hsi SU[†], Hsueh-Ming HANG[†], *and* David W. LIN[†], *Nonmembers*

SUMMARY A global motion parameter estimation method is proposed. The method can be used to segment an image sequence into regions of different moving objects. For any two pixels belonging to the same moving object, their associated global motion components have a fixed relationship from the projection geometry of camera imaging. Therefore, by examining the measured motion vectors we are able to group pixels into objects and, at the same time, identify some global motion information. In the presence of camera zoom, the object shape is distorted and conventional translational motion estimation may not yield accurate motion modeling. A deformable block motion estimation scheme is thus proposed to estimate the local motion of an object in this situation. Some simulation results are reported. For an artificially generated sequence containing only zoom activity, we find that the maximum estimation error in the zoom factor is about 2.8 %. Rather good moving object segmentation results are obtained using the proposed object local motion estimation method after zoom extraction. The deformable block motion compensation is also seen to outperform conventional translational block motion compensation for video material containing zoom activity.

key words: *global motion estimation, local motion estimation, image segmentation*

1. Introduction

Motion estimation plays an important role in video data compression which exploits the high temporal redundancy between successive frames in a video sequence to achieve high compression efficiency. It can also be used for segmenting images into objects moving at different speeds for computer vision applications.

Motion in a video sequence is either due to object movement or due to camera pan and zoom operations. The motion due to object movement is referred to as *local motion* or *object motion*, and the motion due to camera pan and zoom is *global motion*. The most often employed motion estimation technique in video coding, such as that standardized in ITU–T H.261/H.263 and ISO MPEG 1/2, is one of block matching, which gives estimates of the combined local and global motion [1], [2]. Since the global motion is generated by camera movement, it can be represented, in theory, by a few parameters. Hence, the separation of global and local motion may lead to simpler and more efficient motion information representation. Also, the global motion components contained in the motion vectors may confuse an unsophisticated motion-based segmentation algorithm in identification of moving objects. When the global motion components are removed, the remaining local motion information can be more readily used for moving object identification.

Moving object segmentation is essential in object oriented coding, in which each moving or still object forms a basic unit for data compression and object manipulation. Various forms of global motion information extraction and their associated applications have been studied in the past ten years [3]–[10]. Keesman [3] and Tse and Baker [4] demonstrate the advantage in motion information reduction using global motion parameters estimation. To compute the global motion vectors, for example, Konrad and Dubois [5] use a stochastic method and Wu and Kittler [6] and Hoetter [7] use a differential method (based on Taylor series expansion). Their gradient methods are pel recursive and thus are often inaccurate for large global displacement. All the above global motion estimation methods handle camera zoom and rotational pan only. Irani and Anandan [8] propose a unified approach to detect moving objects for both 2D and 3D scenes. Without considering the field depth, their approach does not estimate the camera motion parameters. In [9], Zakhor and Lari propose a seven-parameter camera model to estimate the global motion. In the case of fast zooming, object deformation (enlargement and shrinkage) becomes noticeable. Designed without considering object deformation, their zoom determination algorithm is likely to encounter difficulties as the zoom parameter becomes large. Our approach in this paper is an extended version of [10]. In [10], the estimated motion vectors are at full pixel accuracy. But the zoom-induced motion vectors are in general non-integral. Therefore, the raw motion vectors contain estimation errors. In this work, we account for the object deformation effect in our scheme. We propose a method for zoom factor estimation which can reduce the errors due to integer truncation. And we present an improved model and for local motion compensation following the extraction of the zoom information.

In this paper, we first derive a motion model containing both the local motion due to object movement and the global motion due to camera zoom and pan (both translational and rotational pans). This model

Manuscript received April 3, 1998.

Manuscript revised December 21, 1998.

[†]The authors are with the Department of Electronics Eng. National Chaio-Tung University, 1001 Ta-Hsueh Rd., Hsinchu 300, Taiwan, Republic of China.

originates from image projection geometry that describes the effect of object and camera motion on the apparent motion in recorded video. Our goal is to recover the global motion parameters from the measured (apparent) motion vectors. For any two image pixels belonging to the same moving object, because of the projection geometry in camera imaging, their associated motion vectors should observe a certain relationship. By examining whether the relationship holds for each pair of pixels (or image blocks), we are able to group pixels into objects and, at the same time, identify some global motion information.

This paper is organized as follows. In Sect. 2, we describe the image projection geometry and the consequent model of apparent motion. The aforementioned relationship between the motion vectors of different pixels of the same object is derived in Sect. 3, and the proposed motion component restoration scheme is also presented. Section 4 is devoted to the discussion of experimental results. Some additional details of the proposed schemes are also provided. Section 5 concludes this paper.

2. Modeling of Apparent Motion in Recorded Video

In accordance with the imaging mechanism of typical video cameras, we employ the central projection geometry to model object motion in the recorded images. This geometry is illustrated in Fig. 1. P is a point of interest on an object. Let F_1 be the z-coordinate of the image plane in the object-space. Then, based on similarity between the two triangles $\triangle OPR$ and $\triangle OP'S$, we have

$$Y_1 = F_1\frac{y}{z} \text{ and } X_1 = F_1\frac{x}{z}. \tag{1}$$

A measured motion vector may contain zoom, pan, and object motion as illustrated in Fig. 2. In our discussion, an "object" is defined to be something which has a planar surface parallel to the xy plane (same z coordinate) and which are projected into a group of pixels on the image plane that share the same motion vector. Let V_{obj} be the motion vector of object point P in the original object space and V_{ox}, V_{oy}, and V_{oz} be its x, y and z components. A zoom motion occurs when the camera changes its focal length from F_1 to F_2. Assume, in addition, the camera does a translational motion with a displacement vector V_t and it also rotates around the y-axis by an angle θ_y. Let (x'', y'', z'') be the coordinate of P in the moved coordinate system, and let (X_2, Y_2) be the projection of P on the moved image plane. Then

$$\begin{cases} X_2 = F_2\dfrac{x''}{z''} \\ Y_2 = F_2\dfrac{y''}{z''}. \end{cases} \tag{2}$$

For two video frames separated by 0.1 sec or less, we usually have small θ_y, small V_t, and small V_{obj}. In addition, objects in a video scene are often at a reasonable distance from the camera, Hence, we usually have $|z| \gg |V_{oz}| + |\theta_y(x + V_{ox} - V_{tx})|$. Thus we may use z in place of z'' and rewrite Eq. (2) as [10]

$$\begin{cases} X_2 = \dfrac{F_2}{z}(x + V_{ox} - V_{tx}) + \theta_y F_2 \\ Y_2 = \dfrac{F_2}{z}(y + V_{oy} - V_{ty}). \end{cases} \tag{3}$$

Therefore, the apparent motion vector $(V_x, V_y) = (X_2 - X_1, Y_2 - Y_1)$ on the image plane due to the combined effect of object and camera movement is

$$\begin{cases} V_x = \left(1 - \dfrac{F_1}{F_2}\right) X_2 + \dfrac{F_1}{z}V_{ox} - \dfrac{F_1}{z}V_{tx} + F_1\theta_y \\ V_y = \left(1 - \dfrac{F_1}{F_2}\right) Y_2 + \dfrac{F_1}{z}V_{oy} - \dfrac{F_1}{z}V_{ty}. \end{cases} \tag{4}$$

Equation (4) shows that all the objects in the image plane have the same zoom factor $(1 - \frac{F_1}{F_2})$. For simplicity, we rewrite Eq. (4) as

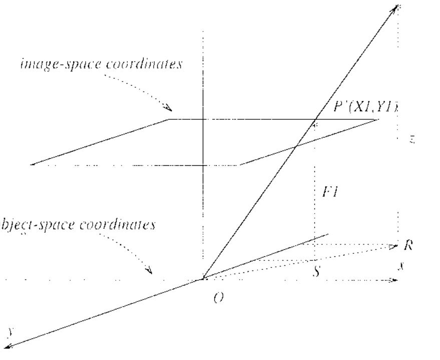


Fig. 1 Central projection imaging.

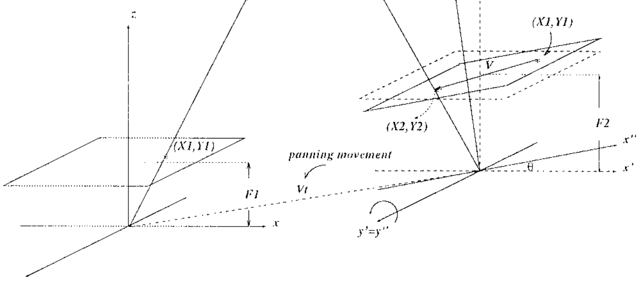


Fig. 2 General motion including zoom, pan, and object movement.

$$\left\{ \begin{aligned} V_x &= \mathcal{Z}X_2 + \mathcal{V}_{ox} + \mathcal{P}V_{tx} + \Theta \\ V_y &= \mathcal{Z}Y_2 + \mathcal{V}_{oy} + \mathcal{P}V_{ty}, \end{aligned} \right. \tag{5}$$

where $\mathcal{Z} = (1 - \frac{F_1}{F_2})$, $\mathcal{P} = -\frac{F_1}{z}$, $\mathcal{V}_{ox} = \frac{F_1}{z}V_{ox}$, $\mathcal{V}_{oy} = \frac{F_1}{z}V_{oy}$, and $\Theta = F_1\theta_y$.

The first term ($\mathcal{Z}$) on the right hand side of Eq. (5) is due to camera zoom. The second term is the projection of object motion on the image plane and the last two terms are due to camera translational pan and rotational pan, respectively. Theoretically, the global motion is characterized by F_1, F_2, z, V_{tx}, V_{ty}, and θ_y. In reality, however, we often cannot calculate the individual values of F_1, F_2, and z, but can only estimate certain combined terms, e.g., $(1 - \frac{F_1}{F_2})$, from the measured motion vectors. Hence, the global motion parameters we will deal with are the zoom factor ($\mathcal{Z}$) of the entire image and the combined pan factor $(\mathcal{P}V_{tx} + \Theta, \mathcal{P}V_{ty})$ of each object. In fact, if no additional information is provided, we cannot separate unambiguously the pan factor from the local motion, as will be discussed later. In the next section, we will devise a method to restore these motion components, namely, zoom factor, pan factor, and object displacement.

3. Motion Components Restoration

3.1 Object-Based Motion Components Restoration

Let there be K pixels in each image. Suppose we let each pixel represent an individual object; thus, there are K objects in the entire image. The movement of each object (pixel) can be expressed in the form of Eq. (5); therefore, we have

$$\left\{ \begin{aligned} V_{xi} &= \mathcal{Z}X_i + \mathcal{P}_iV_{tx} + \mathcal{V}_{ox_i} + \Theta \\ V_{yi} &= \mathcal{Z}Y_i + \mathcal{P}_iV_{ty} + \mathcal{V}_{oy_i}, \end{aligned} \right. \tag{6}$$

for $i = 1, \cdots, K$. Assume tentatively that we know the apparent motion vectors (V_{xi}, V_{yi}). Equation (6) may be considered as a set of $2K$ equations with $3K + 4$ unknowns, where 4 of these (i.e., $\mathcal{Z}$, V_{tx}, V_{ty}, and Θ) are common to all pixels and the rest (i.e., $\mathcal{P}_i$, $\mathcal{V}_{ox_i}$, and $\mathcal{V}_{oy_i}$) are particular to each pixel. If the above linear system can be solved, then we can recover the global motion parameters (i.e., $\mathcal{Z}$, $\mathcal{P}_i$, V_{tx}, V_{ty}, and Θ). However, since there are more unknowns than equations, there exists an infinite number of solutions to this system unless other constraints are provided.

Such constraints are facilitated by realizing that, in real-world pictures, objects are larger than a few pixels. Hence, typically a number of pixels share the same object motion vector. For each such object, there are only 7 unknown motion parameters (i.e. $\mathcal{Z}$, $\mathcal{P}_i$, V_{tx}, V_{ty}, $\mathcal{V}_{ox_i}$, $\mathcal{V}_{oy_i}$, and Θ), but there can be many more equations (two per pixel). Hence, we can better estimate these parameters, if their coefficients are linearly independent in these equations.

Now, to begin the process of object identification and motion parameter estimation, we need to measure the apparent motion vectors (V_{xi}, V_{yi}) first. A reliable estimate of (V_{xi}, V_{yi}) necessarily involves several pixels that have the same apparent motion vector [2]. A first and simple way to address the problem is dividing an image into blocks and assuming that pixels inside each block share the same set of motion parameters. However, the methodology presented below for estimating global motion parameters is not restricted to blockwise-equal motion vectors.

Because, in this paper, an “object” is defined as a set of pixels whose elements (pixels or blocks) share the same motion vector and have similar z-coordinate values, it may not correspond to a physical object in common-language sense. When the field depth is large, our “object” often corresponds to a physical object in the ordinary sense. However, a physical object may be partitioned into a few “objects” under our definition if it has a large surface area and the field depth varies significantly over its extent.

The procedure of our proposed object-based motion restoration scheme is shown in Fig. 3. In the motion estimation step, the motion vector of each image block is estimated using an appropriate block motion vector estimation method. In our simulation, we employ the full-search block matching algorithm (BMA). The resulting motion vector field is the basis for the computation in the next two steps, namely, object assignment and motion components estimation.

3.2 Object Assignment

It is well-known that the block matching algorithm does not always provide consistent motion vectors. The deformation induced by camera zoom and the noise contained in images may lead to incorrect motion vectors. To rectify the possible inconsistency, an $n \times n$ 2-D median filter is applied to the estimated motion field. We then conduct the object assignment process below.

Images are first divided into a number of computational units and in this paper a square image block is used as a computational unit. However, the following approach is also applicable to other computational units such as a single pixel. Let (X_1, Y_1) in Eq. (1) be the center of the computational unit. Given two image units (say, blocks), A and B, according to Eq. (5) we have

$$\left\{ \begin{aligned} V_{xA} &= \mathcal{Z}X_A + \mathcal{P}_AV_{tx} + \mathcal{V}_{oxA} + \Theta \\ V_{yA} &= \mathcal{Z}Y_A + \mathcal{P}_AV_{ty} + \mathcal{V}_{oyA}, \end{aligned} \right. \tag{7}$$

and

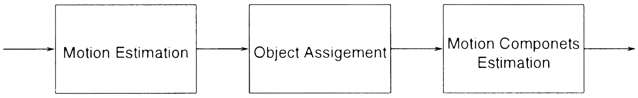


Fig. 3 Motion restoration procedure.

$$\begin{cases} V_{xB} = \mathcal{Z}X_B + \mathcal{P}_B V_{tx} + \mathcal{V}_{oxB} + \Theta \\ V_{yB} = \mathcal{Z}Y_B + \mathcal{P}_B V_{ty} + \mathcal{V}_{oyB}. \end{cases} \tag{8}$$

If these two units belong to the same object (with identical z but different (x, y)), then their motion parameters $\mathcal{P}$, $\mathcal{V}_{ox}$, and $\mathcal{V}_{oy}$ would be equal. And we have, from Eqs. (7) and (8), that

$$\frac{V_{xA} - V_{xB}}{V_{yA} - V_{yB}} = \frac{X_A - X_B}{Y_A - Y_B}. \tag{9}$$

We thus have derived the following *object-assignment rule*.

Object-assignment Rule If zoom exists ($\mathcal{Z}$ is nonzero) and two image units 1 and 2 belong to the same object, then

$$\frac{V_{x1} - V_{x2}}{V_{y1} - V_{y2}} = \frac{X_1 - X_2}{Y_1 - Y_2},$$

where (X_i, Y_i) is the center coordinates of unit i and (V_{xi}, V_{yi}) is the measured motion vector of unit i on the image plane.

There are many ways to group image blocks (units) into objects. We adopt a simple iterative approach as follows. The image blocks are indexed in the raster-scan order and are denoted B_i, $i = 1, \ldots, N$.

Step 0: Set $j = 1$. All blocks are unmarked.

Step 1: Among all the unmarked blocks, choose the one with the smallest index as the reference block and denote it B_{ref}. Mark this block and assign it to object j.

Step 2: For each of the remaining unmarked blocks, test it (as unit B in Eq. (9)) against B_{ref} (as unit A in Eq. (9)) for the equality in Eq. (9). If the equality holds, then mark it and assign it to object j; else, skip it.

Step 3: If all blocks are marked, stop. Otherwise, let $j = j + 1$ and go to Step 1.

3.3 Motion Components Estimation

One way of performing the motion components estimation is to decompose the whole process into three cascaded sub-steps for global motion and local motion estimation. In this process, the apparent motion vector of an object is first processed for zoom estimation. Then, the zoom-removed motion vector is put through zoom-compensated object motion estimation. And finally, camera pan is estimated. We now describe in detail each sub-step.

3.3.1 Zoom Estimation

Based on our global motion model, the zoom-removed motion vector $V_r = V_{obj} + V_t + V_{rot}$ should be a constant for all image blocks belonging to the same object, where V_{rot} represents the motion vector induced by camera rotation. This property is used to estimate the zoom factor, $\mathcal{Z}$. Consider an object consisting of p blocks. Since all the p blocks have the same z-axis coordinate, Eq. (5) is reduced to

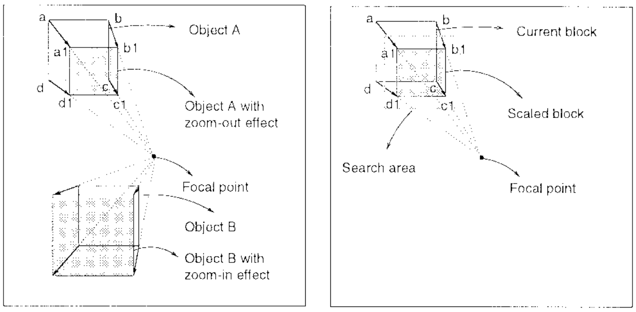
$$\begin{cases} V_{xi} = \mathcal{Z}X_i + V_{rx} \\ V_{yi} = \mathcal{Z}Y_i + V_{ry}, \end{cases} \quad \text{for } i = 1, 2, \cdots, p, \tag{10}$$

where the coordinate (X_i, Y_i) is the center of block i, $V_{rx} = \mathcal{P}V_{tx} + \mathcal{V}_{ox} + \Theta$, and $V_{ry} = \mathcal{P}V_{ty} + \mathcal{V}_{oy}$.

Block-based motion vector estimation fails at object boundaries where pixels inside the same block move in different directions and/or at different speeds, especially for small objects. Thus, the zoom factor estimation should be calibrated according to object sizes. Assume the image is divided into M objects by applying the object assignment rule and let these M objects be denoted $O_1, O_2, \cdots, O_M$. We first estimate the zoom factor of each individual object and call it Z_i for object O_i. Let $N(O_i)$ be the number of blocks belonging to object O_i. The overall zoom factor is a weighted average of all the individual zoom factors,

$$Z' = \frac{1}{\sum_{i=1}^{M} N(O_i)} \sum_{i=1}^{M} N(O_i) Z_i. \tag{11}$$

Figure 4 (a) illustrates the deformation of an object caused by camera zoom. The size of object A becomes smaller due to zoom-out operation. According to Eq. (5), for any given pixel P on the image plane, the displacement of this pixel due to camera zoom is proportional to the distance between P and the focal point. In Fig. 4 (a), since point a is farther from the focal point than any other pixel on object A, its associated global movement $\overline{aa_1}$ is the largest. On the other hand, point c is closest to the focal point; thus, the global movement $\overline{cc_1}$ is the smallest. This non-uniform pixel movement results in shrinkage deformation of object A. In contrast, under zoom-in operation, an object (such as object B in Fig. 4 (a)) becomes larger. Due to such object deformation, the motion vectors obtained using the block-matching method are sometimes incorrect. In the presence of zoom, since the movement of



(a) (b)

Fig. 4 (a) Deformation of objects caused by camera zoom. (b) Block-deformed motion estimation.

an object near the focal center is relatively small, the limited-precision motion vectors (which are integers in our simulation) obtained by the initial BMA may be especially relatively inaccurate in this area. To improve our zoom estimate, a two-step zoom factor estimator is devised.

We first use Eq. (11) to get a rough zoom factor Z'. The raw motion field estimated by the block-matching method consists of both global and local motion. Based on the estimated global zoom factor, we skip the blocks that violate the following two conditions:

1. The angle ϕ between the raw motion vector $\vec{V}_i^r$ and the motion vector $\vec{V}_i^g$ obtained by substituting Z' in place of Z_i is smaller than ϕ_{max}.
2. The magnitude difference mag between $\vec{V}_i^r$ and $\vec{V}_i^g$ is smaller than mag_{max}.

These two conditions are proposed based on the observation that if the fractional part of the motion vector does not have a strong impact on the estimation of the zoom factor, then ϕ and mag must be small. These conditions are used to identify the more accurate motion vectors in the raw motion field. The blocks satisfying those two conditions are deformed by the zoom factor and are compared to the reference picture. The zoom factor is varied between $Z'(1-d)$ to $Z'(1+d)$, where d is some small number, to find a value which minimizes the frame difference. The final value is denoted $\mathcal{Z}$. After the zoom factor $\mathcal{Z}$ is removed, $V_r = (V_{rx}, V_{ry})$ is passed to the zoom-compensating block-deformed motion estimation sub-step.

3.3.2 Block-Deformed Motion Estimation

With the $\mathcal{Z}$ component in Eq. (6) is removed by the zoom estimator, the remaining terms signify translational motion and they are given by

$$\begin{cases} V_{rxi} = \mathcal{P}V_{tx} + \mathcal{V}_{ox} + \Theta \\ V_{ryi} = \mathcal{P}V_{ty} + \mathcal{V}_{oy}, \end{cases} \quad \text{for } i = 1, \cdots, p. \tag{12}$$

Although we may use these values for subsequent computation, we conduct a zoom-compensating block-deformed motion estimation for added accuracy in motion vector values. The estimation method can be motivated by considering Eq. (4). We see from the equation that, if $\mathcal{Z}$ is known, then the zoom-removed motion can be estimated by searching around the zoomed object $\frac{F_1}{F_2}(X_i, Y_i)$ for the translational motion vector yielding the least motion-compensated prediction error. The resulting estimate could be more accurate than that obtained in the initial BMA (which corresponds to "zoom-compensated" motion estimation with $\mathcal{Z} = 0$).

Figure 4 (b) illustrates the proposed method. According to Eq. (5), the original block $\square abcd$ is deformed to $\square a_1b_1c_1d_1$ by the zoom factor $\mathcal{Z}$. During the deformation process, the pixels of a block may be moved to non-integral positions. We use simple 2-D bilinear interpolation to regenerate the integer location pixels. Within the given search area, we choose the displacement (V^*_{rxi}, V^*_{ryi}) that yields the minimum absolute difference between the block $\square a_1b_1c_1d_1$ and the reference block. Thus, (V^*_{rxi}, V^*_{ryi}) represents the zoom-removed motion of $\square abcd$. If, due to noise, (V^*_{rxi}, V^*_{ryi}) are different for different i, then a simple averaging over i (equivalent to MMSE estimation [11]) yields a single zoom-removed motion vector for the object.

3.3.3 Pan Estimation

Now return to Eq. (12) with (V^*_{rxi}, V^*_{ryi}) in place of (V_{rxi}, V_{ryi}). Note that because we assume all the p blocks belong to the same co-planar object, they share the same focal ratio $\mathcal{P}$, the same object motion vector $(\mathcal{V}_{ox}, \mathcal{V}_{oy})$, and the same pan factors (V_{tx}, V_{ty}) and Θ. If we examine the equation carefully, we find that the rank of the unknowns equals to 2 which is smaller than 6, the number of unknowns. In fact, the above observation reveals one fact; that is, the three quantities, $\mathcal{P}V_{tx}$, $\mathcal{V}_{ox}$, and Θ always stick together. Hence, we can not estimate their values separately. Similarly, $\mathcal{P}V_{ty}$ and $\mathcal{V}_{oy}$ can not be separated. If we know *a priori* that the object does not have local motion, then the combined pan vector $(\mathcal{P}V_{tx} + \Theta, \mathcal{P}V_{ty})$ of the object is equal to (V_{rxi}, V_{ryi}). In addition, the values of $\mathcal{P}V_{tx}$, $\mathcal{P}V_{ty}$, and Θ can be estimated via MMSE estimation over the whole image [11].

3.4 Overall Scheme

In summary, our complete global and local motion estimation/segmentation is shown in Fig. 5. We start with the raw motion field estimated using full-search BMA followed by 2-D median filtering. Then, image blocks are grouped into objects according to the object-assignment rule. Next, the zoom factor is estimated. After extracting the camera zoom, block-deformed motion estimation is conducted. Finally, the camera pan may be extracted if we know the object motion or the

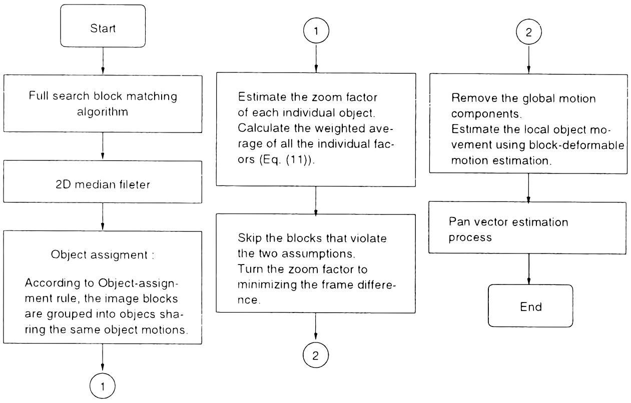

Start
Full search block matching algorithm
2D median fileter
Object assigment :
According to Object-assignment rule, the image blocks are grouped into objecs sharing the same object motions.
1
1
Estimate the zoom factor of each individual object. Calculate the weighted average of all the individual facors (Eq. (11)).
Skip the blocks that violate the two assumptions. Turn the zoom factor to minimizing the frame difference.
2
2
Remove the global motion components. Estimate the local object movement using block-deformable motion estimation.
Pan vector estimation process
End

Fig. 5 Flowchart of the overall scheme.

relative field depths of the objects.

4. Experimental Results

The proposed algorithm was tested on a variety of video sequences. *Table Tennis* and *Flowergarden* are 80-frame sequences with 240 lines × 352 pixels resolution and the *MITz* contains 22 frames with resolution 256 lines × 256 pixels. *Table Tennis* is a photographed sequence containing both individual object movement and zoom-out action. *Flowergarden* contains only translational pan motion. And *MITz* sequence (whose 20th frame is shown in Fig. 6) is created artificially from the first frame of the *MIT* sequence (a well-known HDTV test sequence originally produced by Massachusetts Institute of Technology) with a zoom-out factor $= -0.03$.

4.1 Zoom Estimation

The full-search BMA is used to estimate the compound raw motion vectors. The estimated motion vectors are at full pixel accuracy but the zoom-induced motion vectors are in general non-integral. Therefore, the raw motion vectors contain estimation errors. Given an estimated zoom factor, a motion field can be rebuilt based on Eq. (1). Denote it by $(\hat{V}_x, \hat{V}_y)$. The rebuilt vector $(\hat{V}_x, \hat{V}_y)$ can be decomposed into an integral part and a fractional part. If we ignore the fractional part and use only the integral part to derive the zoom factor, then the error in zoom factor estimation can be written as

$$\begin{cases} dZ_x = \dfrac{\hat{V}_x}{X} - \dfrac{\hat{V}_{xi}}{X} = \dfrac{dV_x}{X} \\ dZ_y = \dfrac{\hat{V}_y}{Y} - \dfrac{\hat{V}_{yi}}{Y} = \dfrac{dV_y}{Y} \end{cases}, \tag{13}$$

where dZ_x and dZ_y are the zoom factor errors in the x and y directions, respectively, (X, Y) is the block coordinate relative to the focal point, and $(\hat{V}_{xi}, \hat{V}_{yi})$ denotes the integral part and (dV_x, dV_y) denotes the fractional part. According to Eq. (13), for a block located near the focal point; that is, when either X or Y is close to 0, the estimation error dZ_x or dZ_y can be large. Figure 7 shows the zoom factor estimation error caused by integer truncation with a zoom factor $= -0.03$ for the *MITz* sequence with block size 16 × 16. Note that the estimation errors dZ_x and dZ_y are relatively large along the x- and y- axes (small X or Y), verifying the above analysis. In Sect. 3.3, we suggested two conditions to fine-tune the estimated zoom factor. In that refining process, two threshold values ϕ_{max} and mag_{max} were introduced. Figure 7 shows that the estimation errors are less than 0.2 % in regions away from the x- and the y-axes. Therefore, the tuning range d for Z' as introduced in Sect. 3.3 is set equal to 0.2 %. Simulation results for zoom factors ranging from -0.02 to -0.04 show that the values of ϕ are typically smaller than 0.1 and the values of mag are smaller than 0.2. Therefore, we let $\phi_{max} = 0.1$ and $mag_{max} = 0.2$ in our simulation.

Figure 8 (a) depicts the estimated zoom factor using the proposed method performed on the blocks satisfying the two zoom factor conditions. As described previously, the *MITz* sequence has a zoom factor $= -0.03$. The maximum estimation error in Fig. 8 (a) is about 2.8 %.

We next test the *Table Tennis* sequence. At the beginning of the *Table Tennis* sequence, the arm and

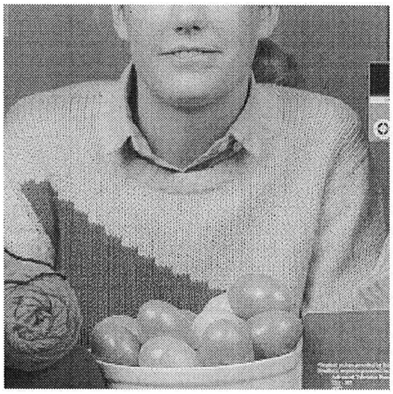


Fig. 6 The 20th frame of the *MITz* sequence.

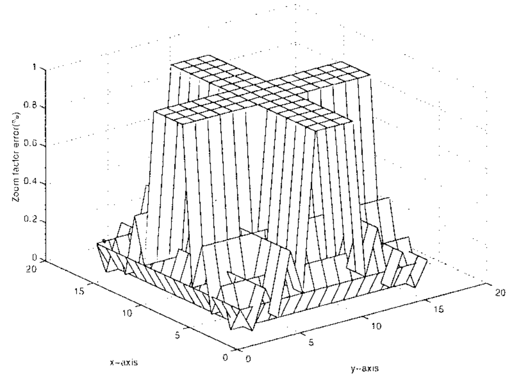
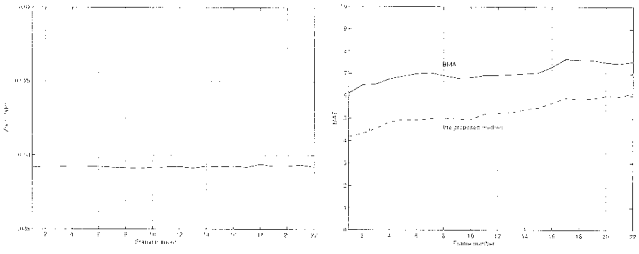


Fig. 7 Zoom factor error caused by integer truncation with zoom factor $= -0.03$.



(a)

(b)

Fig. 8 (a) The estimated zoom factor for the *MITz* sequence. (b) Comparison of mean absolute prediction errors for the *MITz* sequence.

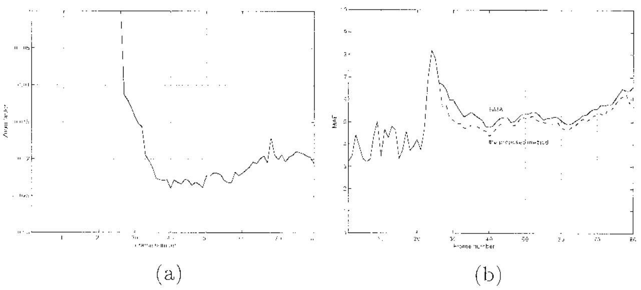


Fig. 9 (a) The estimated zoom factor for the *Table Tennis* sequence. (b) Comparison of mean absolute prediction errors for the *Table Tennis* sequence.

the ping-pong ball are the only moving items. The zoom-out operation takes place at the 27th frame. Figure 9 (a) shows the estimated zoom factor, which equals to zero during the first 26 frames and is around -0.02 for the rest of the sequence. This result is consistent with visual inspection. Since we do not know the exact zoom factor in this case, we cannot measure the errors in its estimation.

4.2 Prediction Performance of the Overall Algorithm

Figure 8 (b) compares the mean absolute prediction error (MAE) of the conventional BMA and the proposed scheme for the *MITz* sequence. The proposed scheme yields a smaller MAE which is about 20 % less. Similarly, for the *Table Tennis* sequence, we also obtain a lower MAE using the proposed scheme, as shown in Fig. 9 (b). Since there is no zoom action until the 27th frame, the proposed method has the same MAE performance as the conventional BMA for the first 26 frames. In the rest of the sequence, the proposed scheme outperforms the conventional BMA scheme.

4.3 Motion Fields and Object Segmentation

If global motion exists, it is difficult to recognize the moving objects by merely looking at the motion vectors. In our proposed scheme, the global motion caused by zoom can be separated from the object movement. As a result, moving objects at different speeds can be identified.

Consider the *Table Tennis* sequence which contains not only zoom-out action but also object movement. Figures 10 (a) and (b) show the initial motion field and the final object movement obtained from our scheme, respectively, for frame 30. The motion of the ping-pong ball and the arm are as clearly indicated by the final object motion field. However, some apparently incorrect block motion is also obvious. This can be attributed to incorrect initial motion estimation.

Figure 11 shows the segmented moving objects after their global motion was extracted. The incorrect identification around the edge of the table may due to a kind of aperture problem—the simple image texture around these edges leads to incorrect initial motion vector estimates. To further improve the performance of our scheme, one essential step is to improve the initial estimates of motion vectors.

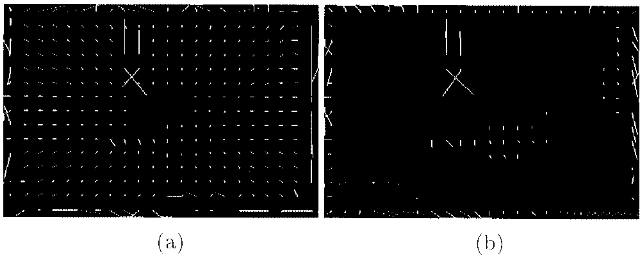


Fig. 10 (a) Initial motion field at the 30th frame of the *Table Tennis* sequence obtained by full-search BMA. (b) Final object motion field obtained by the proposed scheme for the same frame.

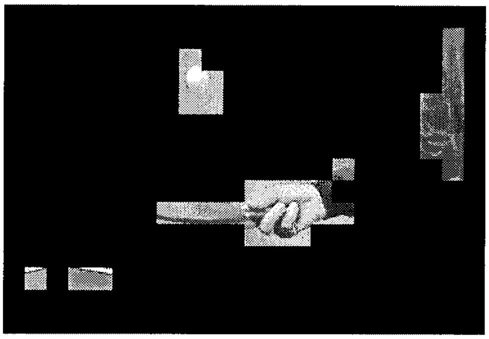


Fig. 11 Moving objects identified in the 30th frame of the *Table Tennis* sequence.

In the last section, we explained that, without certain information such as the z-coordinates of objects, we cannot separate the pan vector from object motion unambiguously. However, if the object motion is known (e.g., if the objects are still) and there exists a translational pan, then, according to Eqs. (12) and (4), the relative depth between any two objects can be obtained. Figure 12 shows a translational pan example. Objects are grouped into 5 image planes, each having a different depth. The sky and the house constitute the farthest image plane. Because the field depth value z of the tree trunk varies from the top to the bottom, its top part, middle part and bottom part appear to move at different speeds due to camera pan. By our "object" definition (as discussed in Sect. 3.1), the tree is split into three objects.

5. Conclusion

We described a mathematical model for apparent interframe image motion which may contain camera zoom, camera translational and rotational pans, and object movement. Based on this geometrical model, we proposed an object assignment rule that the pixels on the

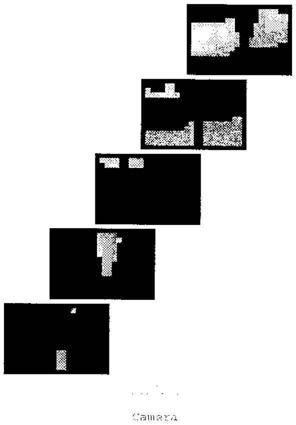


Fig. 12 Objects at different field depths in the 6th frame in the *Flowergarden* sequence.

same object should comply with. By examining the image pixels against this rule, an image can be partitioned into objects. Then, a motion component restoration procedure was developed to estimate the zoom factor and other motion parameters. Depending on one's knowledge about the contents of the scene, the pan factor and individual object motion may also be able to be separated and estimated. Simulations were conducted and the results show that the proposed method yields better motion-compensated interframe prediction than the simple conventional BMA.

There are several possible ways to further enhance our algorithm. First, since our motion component restoration procedure is based on the measured motion field, a better motion vector estimation algorithm should improve the global motion restoration performance. Secondly, iterations over the segmentation step and the motion vector estimation step may produce better results. And thirdly, it was for simplicity that we assumed each object to be composed of square blocks. Allowing more natural object shapes should result in more accurate motion estimation as well as better object assignment.

Acknowledgment

This work was supported by the National Council of ROC under grant NSC 82-0404-E-009-112.

References

[1] K.R. Rao and J.J. Hwang, Techniques and Standards for Image, Video, and Audio Coding, Prentice Hall, 1996.

[2] H.-M. Hang and Y.-M. Chou, "Motion estimation for image sequence compression," in Handbook of Visual Communications, eds. H.-M. Hang and J.W. Woods, Academic Press, San Diego, CA, 1995.

[3] G. Keesman, "Motion estimation based on a motion model incorporating translation, rotation and zoom," Signal Processing, vol.4, pp.31–34, 1988.

[4] Y.T. Tse and R.L. Baker, "Global zoom/pan estimation and compensation for video compression," Proc. IEEE Int. Conf. Acoust. Speech Signal Processing, pp.2725–2728, April 1991.

[5] J. Konrad and E. Dubois, "Estimation of image motion field: Bayesian formulation and stochastic solution," Proc. IEEE Int. Conf. Acoust. Speech Signal Processing, pp.1072–1075, April 1988.

[6] S.F. Wu and J. Kittler, "A differential method for simultaneous estimation of rotation, change of scale and translation," Signal Processing: Image Commun., vol.2, pp.69–80, 1990.

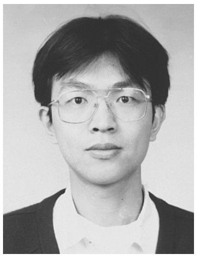
[7] M. Hoetter, "Differential estimation of the global motion parameters zoom and pan," Signal Processing, vol.16, pp.249–265, 1989.

[8] M. Irani and P. Anandan, "A unified approach to moving object detection in 2D and 3D scenes," IEEE Trans. Pattern Anal. & Mach. Intell., vol.20, no.6, pp.577–589, June 1998.

[9] A. Zakhor and F. Lari, "Edge-based 3-D camera motion estimation with application to video coding," IEEE Trans. Image Processing, vol.2, no.4, pp.481–498, Oct. 1993.

[10] J.S. Su, H.M. Hang, and D.W. Lin, "Motion restoration—A method for object and global motion estimation," SPIE Visual Communications and Image Processing '94, pp.1870–1881, Sept. 1994.

[11] G.H. Golub and C.F. van Loan, Matrix Computations, The Johns Hopkins University Press, Baltimore and London, 1989.



Chi-Hsi Su received the BS degree in electrical engineering form the Tatung Institute of Technology, Taipei, Taiwan, R.O.C. in 1989, and the M.S. degree in electronics engineering from National Chiao Tung University (NCTU), Hsinchu, Taiwan, R.O.C. in 1991. He is currently a Ph.D. student in the Department of Electronics Engineering, NCTU. His current research interests include video coding and combined source/channel coding.

Hsueh-Ming Hang received Ph.D. in Electrical Engineering from Rensselaer Polytechnic Institute, Troy, NY, in 1984. From 1984 to 1991, he was with AT&T Bell Laboratories, Holmdel, NJ. He joined the Electronics Engineering Department of National Chiao Tung University, Hsinchu, Taiwan, in December 1991. He was an associate editor of *IEEE Transactions on Image Processing* and currently an associate editor of *IEEE Transactions on Circuits and Systems for Video Technology* and an editor of *Journal of Visual Communication and Image Representation*, Academic Press.

David W. Lin received the Ph.D. degree in electrical engineering from the University of Southern California, Los Angeles, in 1981. He was with Bell Laboratories during 1981–1983, and with Bellcore during 1984–1994. Since 1990, he has been a Professor in the Department of Electronics Engineering and the Center for Telecommunications Research, National Chiao Tung University, Hsinchu, Taiwan, R.O.C. His research interests include various topics in signal processing and communication engineering.