Velocity Determination in Scenes Containing Several Moving Objects *

CLAUDE L. FENNEMA†

Control Data Corporation, Minneapolis, Minnesota 55440

AND

WILLIAM B. THOMPSON

University of Minnesota, Minneapolis, Minnesota 55455

Received February 8, 1978; revised November 8, 1978

A method is described which quantifies the speed and direction of several moving objects in a sequence of digital images. A relationship between the time variation of intensity, the spatial gradient, and velocity has been developed which allows the determination of motion using clustering techniques. This paper describes these relationships, the clustering technique, and provides examples of the technique on real images containing several moving objects.

1. INTRODUCTION

Information about the velocity of regions in an image can be important to the scene analysis process. This information can contribute significantly to scene interpretation: Object trajectories may be calculated, structural ambiguities may be resolved, important events noted, and future situations predicted. At lower levels, information about motion of small structures in an image sequence can be used as an important cue for image segmentation. A discontinuity in the velocity of small image areas strongly implies the existence of an object boundary. Visually dissimilar image regions with the same velocity may either be part of the same object or semantically linked in the overall scene interpretation. If an observer is moving, perceived motion represents an important depth cue.

Because of the value of motion information, increasing attention is being paid to methods analyzing time-varying visual environments. However, many of the methods for motion analysis developed to date are computationally expensive, require already segmented pictures, or are ineffective when object boundaries are

* A contribution of the Central Research Laboratory of 3M Company, St. Paul, Minnesota 55101, and the Department of Computer Science, University of Minnesota, Minneapolis, Minnesota 55455.

† Formerly of 3M Central Research Laboratories.

0146-664X/79/040301-15$02.00/0

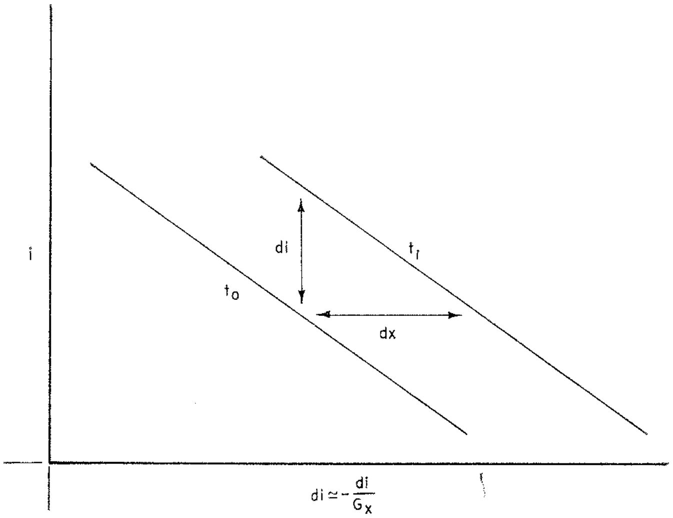

FIG. 1. Intensity wedge represented by the diagonal line at t_0 moves dx along the x axis to t_1 causing a change in intensity of di.

hidden by occlusions or picture borders. In this paper we describe an efficient method for computing the speed and direction of one or more moving objects in a scene. Our method requires neither previously segmented images nor access to complete boundaries within the images. In following sections, we briefly survey the relevant literature, describe our new method, and demonstrate the technique on a number of examples.

2. BACKGROUND

Several approaches have been used to determine velocities from image sequences. Most methods are based on a matching procedure. A pattern in one image frame is searched for in a succeeding image frame. If the pattern is found, the velocity is calculated from the positional shift. An alternate approach is to directly calculate the velocity information from the spatial gradient of the images and the local intensity changes over time due to motion. Matching techniques have been performed at a variety of picture interpretation levels. Badler [1] has described a method which matches objects in a series of images in which the objects have been located and identified. Here the matching task is relatively modest and the major computational effort is in the motion interpretation. Aggarwal and Duda [2] develop a technique for matching vertices in a domain of highly occluded objects. Their approach requires accurate boundary determination in each frame but does not presuppose any static interpretation. Potter [3] suggests matching certain low-level properties of regions. Other researchers have performed the matching on the unprocessed image using cross correlation [4, 5]. A recent approach differences successive frames and then tracks the areas of large differences to determine velocities [6, 7]. A more complete survey of motion analysis is found in [8].

Each of these methods has limitations. Obviously, procedures which require an accurate segmentation cannot be used to assist in the segmentation process itself. Methods dependent on cross correlation are computationally expensive. While some efficiencies are possible [9, 10], much effort is still required to perform the matching at a dense enough sampling of points to be useful for tasks such as segmentation. Techniques using frame-to-frame differences provide efficiency, but require sophisticated higher-level processing to deal effectively with occlusions and situations in which object boundaries are not completely within the image frame [7].

A nonmatching approach to motion analysis of simple scenes has been developed by Limb and Murphy [11]. They relate intensity changes over time at a point to the spatial intensity gradient in the neighborhood of the point. The frame rate is assumed to be sufficiently rapid such that the value of the gradient at a point does not change significantly between frames. Motions along the x and y axes are treated independently. If ΔI is the intensity change at a point due to x axis motion Δx, then it is claimed that

$$\Delta x = \frac{\Delta x}{\Delta I}\,\Delta I \approx -\frac{\Delta I}{G_x},$$

where G_x is the spatial gradient of the intensity along the x axis. (The minus sign is necessary because the surface, rather than the observation point, is moving.) A similar relationship holds for motion only along the y axis. Figure 1 illustrates the effect. Unfortunately, ΔI is affected by motion along both axes and so x-axis motion contributes to the Δy estimate while y-axis motion contributes to the Δx term. Thus the values of the Δx and Δy may be inaccurate for any single image point. However, if the direction of the gradient is randomly distributed, the average values of Δx and Δy give an accurate estimate of the true velocity. The dependency on average value limits the method to the analysis of a single motion.

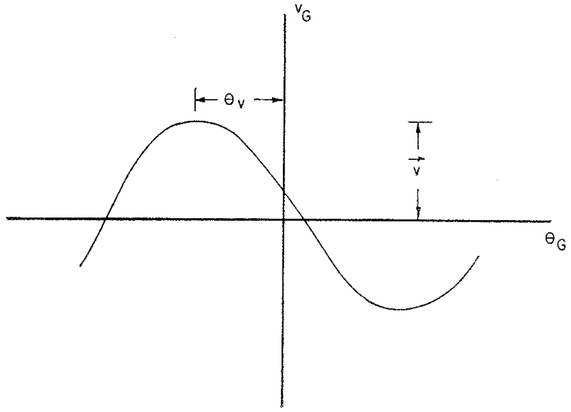

FIG. 2. Values of $V_G = (di/dt)/|\mathbf{G}|$ plotted against θ_G form a cosine curve with amplitude $|\mathbf{v}|$ and phase $-\theta_v$.

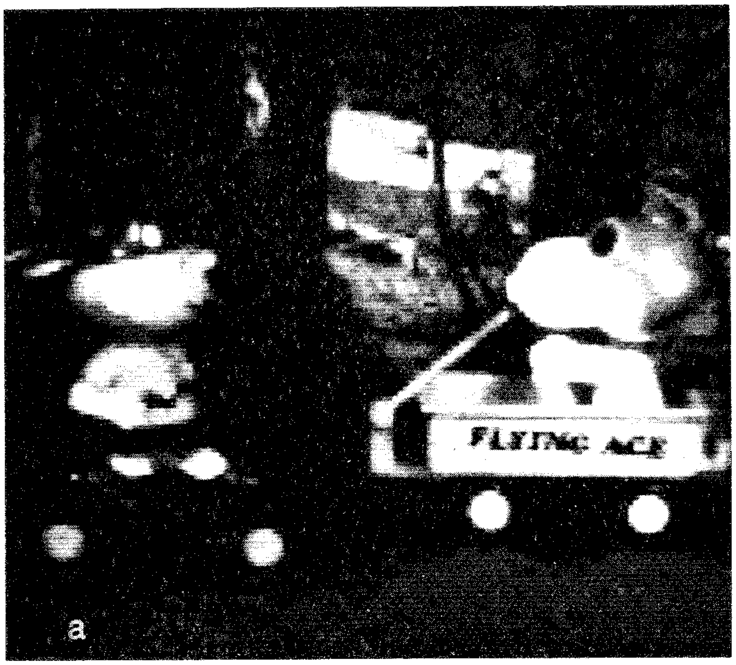

FIG. 3a. Toy problem.

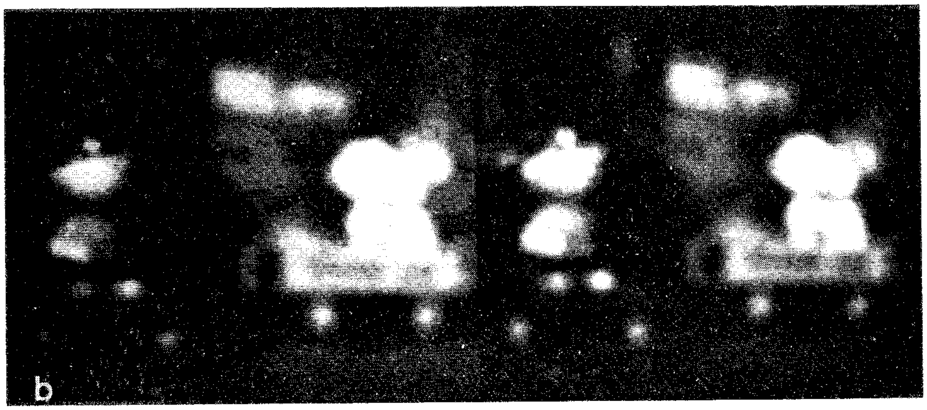

FIG. 3b. Blurred pair.

In the next section, we describe a significant extension to the Limb and Murphy approach: Our method allows the analysis of several moving objects, does not require randomly distributed gradients, and can identify individual pixels belonging to a moving object with a particular velocity. Moreover, there is no need for any image segmentation prior to application of the method.

3. THE GRADIENT-INTENSITY TRANSFORM METHOD

Our technique, which we call the Gradient Intensity Transform Method (GITM), is a nonmatching approach which develops velocity information from

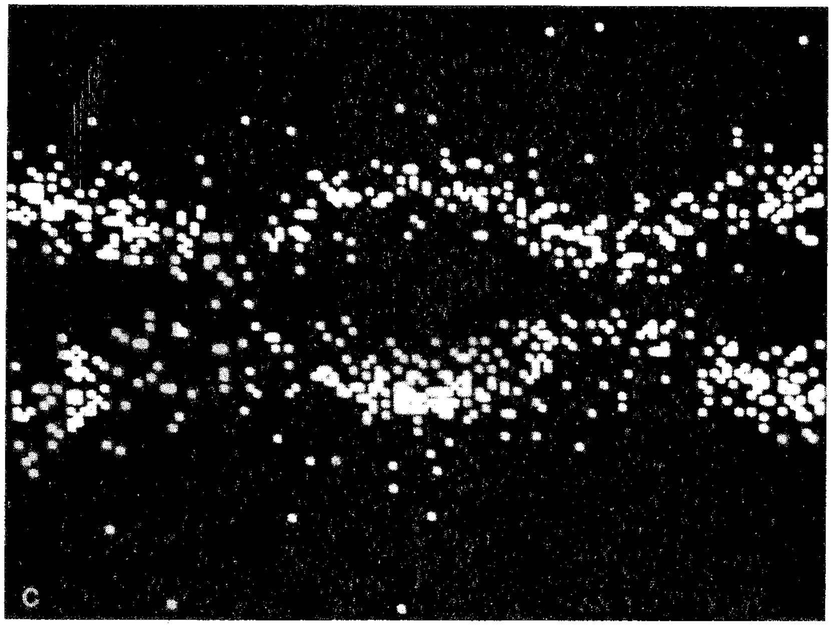

FIG. 3c. Points in (v_G, θ_G) space.

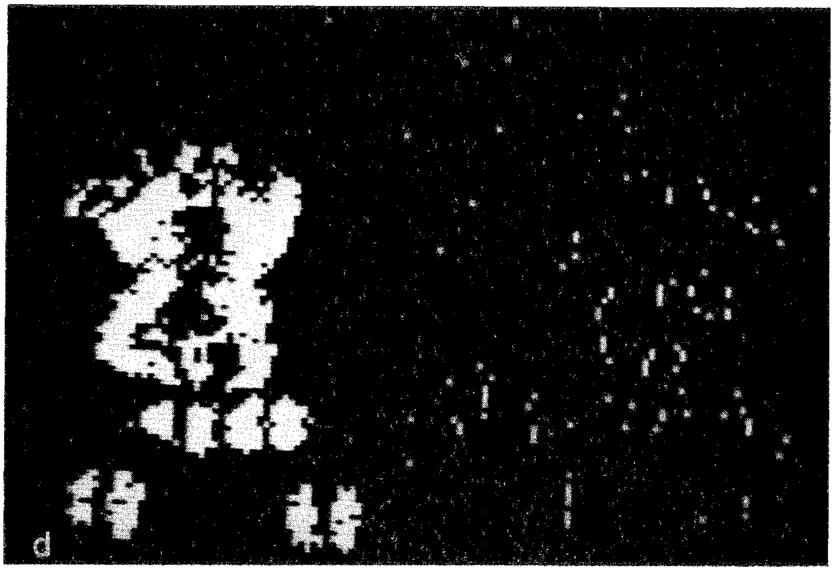

FIG. 3d. Pixel classification for Object 1.

the time variation of intensity at a pixel due to the motion and from the spatial variation of the intensity function. It is applicable for rigid body translation not involving rotation. In the following paragraphs, we will show that the time variation of intensity and the spatial gradient at each point place constraints on the possible speed and direction of the surface or object at that point. These

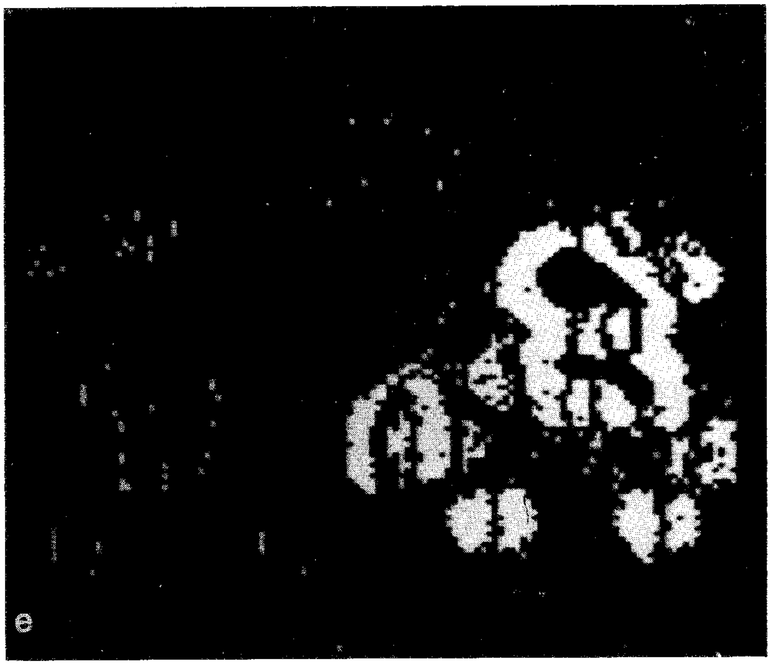

FIG. 3e. Pixel classification for Object 2.

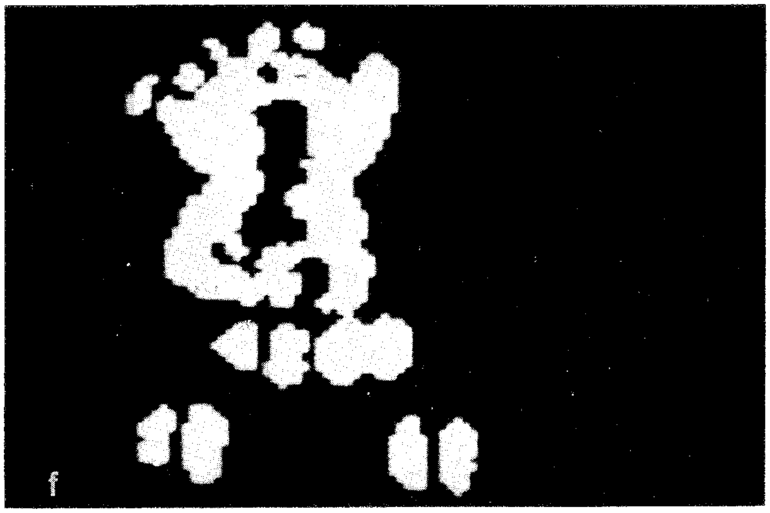

FIG. 3f. Filtered classification for Object 1.

constraints do not, however, uniquely determine the velocity vector; but, if a sufficient number of image points correspond to a surface with a common velocity, the velocity vector can be determined using clustering techniques. We describe a modified Hough transform clustering approach which is used to determine this velocity.

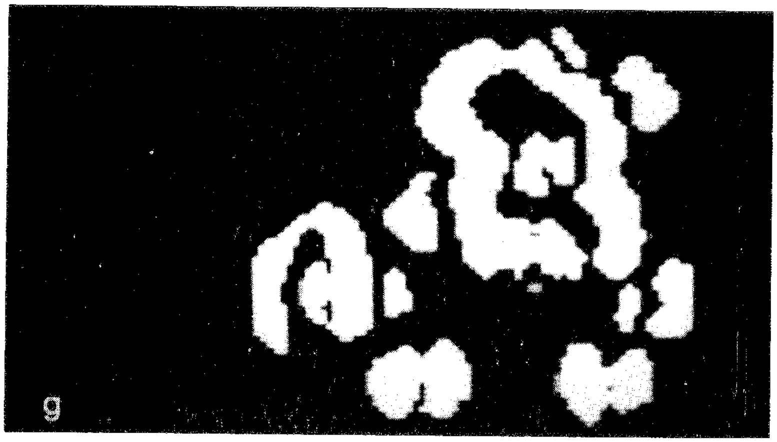

FIG. 3g. Filtered classification for Object 2.

Motion can be represented by a velocity vector

$$\mathbf{v} = \frac{d\mathbf{s}}{dt}.$$

From the definition of the gradient, an incremental change in intensity due to a spatial shift of the evaluation point, $d\mathbf{p}$, satisfies the relationship

$$di = \mathbf{G} \cdot d\mathbf{p}.$$

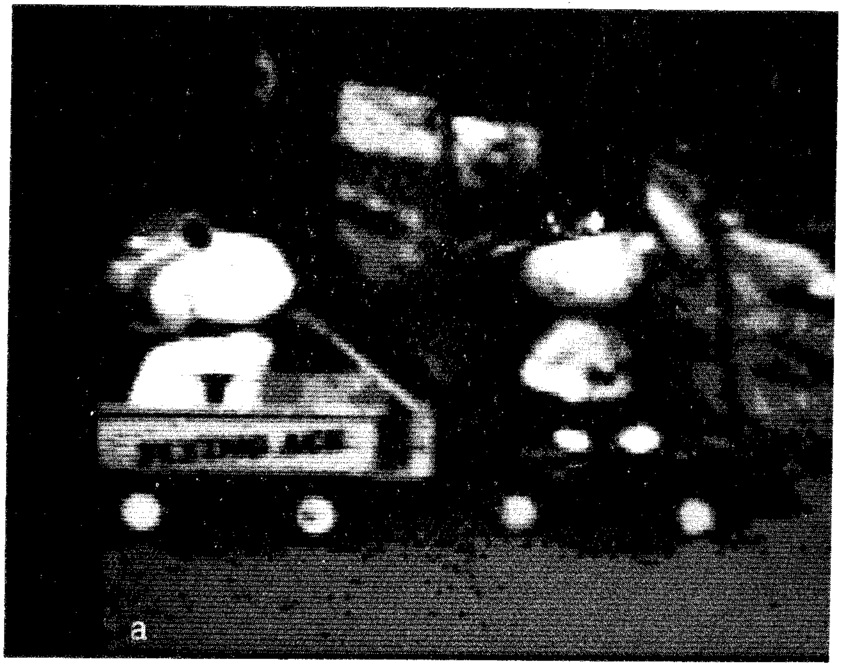

FIG. 4a. The race.

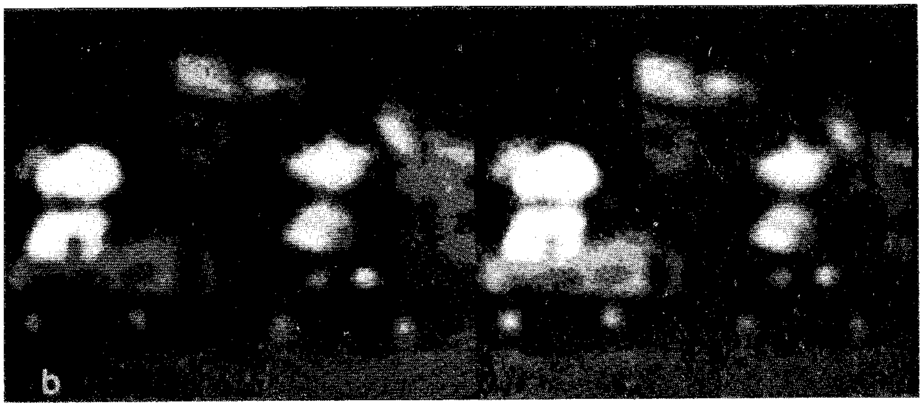

FIG. 4b. Blurred pair.

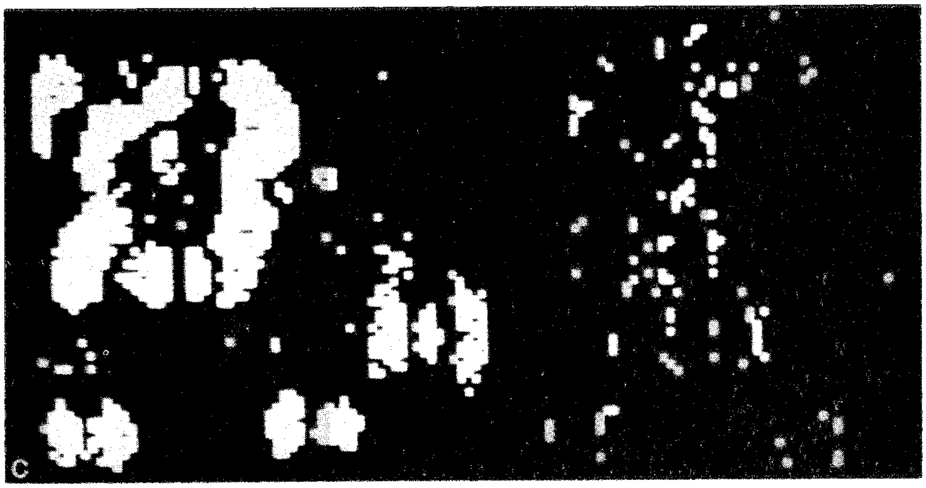

FIG. 4c. Unfiltered pixel classification for Object 1.

In our case, however, it is the underlying surface that is moving, not the sampling point. Thus, if $d\mathbf{s}$ is the incremental translation of the surface

$$di = -\mathbf{G}\cdot d\mathbf{s}. \tag{1}$$

It will be useful to represent both velocities and gradients in a magnitude-phase (polar coordinate) notation

$$\mathbf{G} = (|\mathbf{G}|, \theta_G), \qquad \mathbf{v} = (|\mathbf{v}|, \theta_v).$$

We can relate translational differences to velocity and time by

$$d\mathbf{s} = \mathbf{v}dt. \tag{2}$$

Combining (1) and (2)

$$di = -\mathbf{G}\cdot\mathbf{v}dt = -|\mathbf{G}|\,|\mathbf{v}|\cos(\theta_G - \theta_v)dt. \tag{3}$$

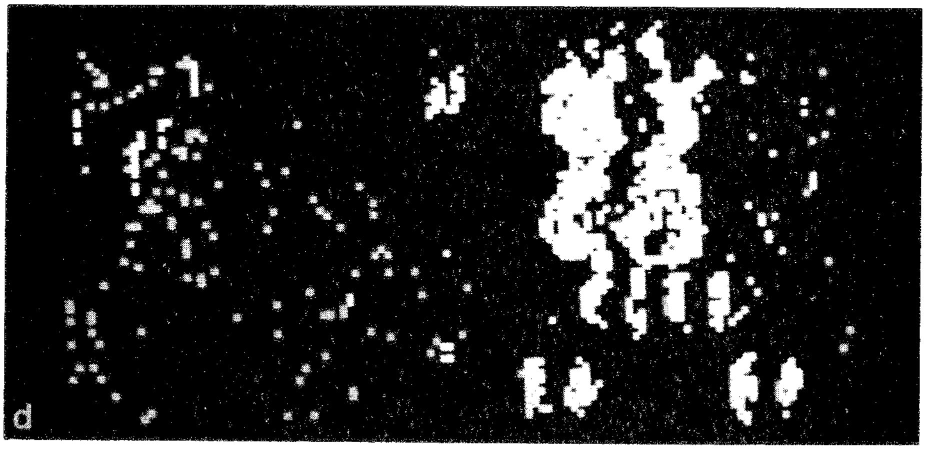

FIG. 4d. Unfiltered pixel classification for Object 2.

Let v_G be the magnitude of the velocity vector projected onto the gradient vector. Then from (3) we have

$$v_G = -|\mathbf{v}|\cos(\theta_G - \theta_v) = \frac{di/dt}{|\mathbf{G}|}. \tag{4}$$

If velocity is constant over the image, then values of v_G plotted against θ_G will lie along a cosine curve with amplitude $|\mathbf{v}|$ and phase $-\theta_v$. If sufficient variability in θ_G exists, the relationship between v_G and θ_G will uniquely define $\mathbf{v}$ (see Fig. 2).

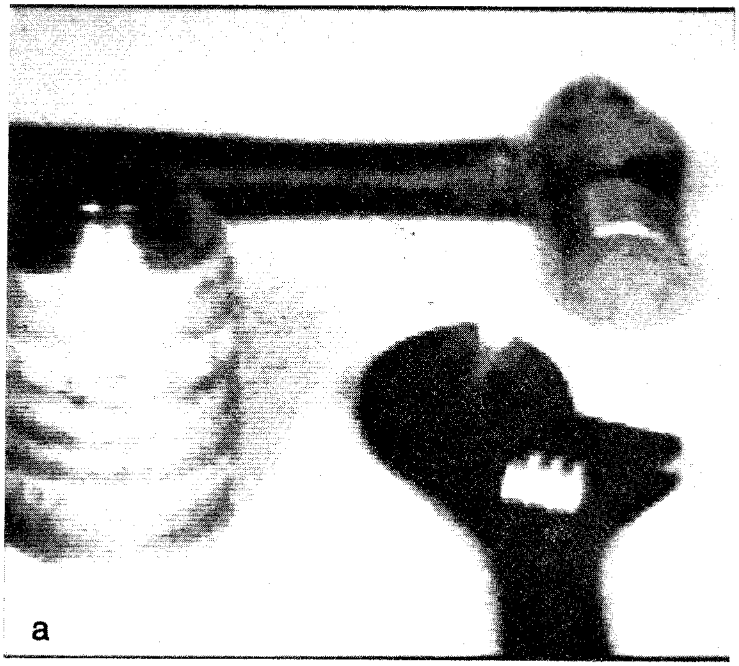

FIG. 5a. Moving tools.

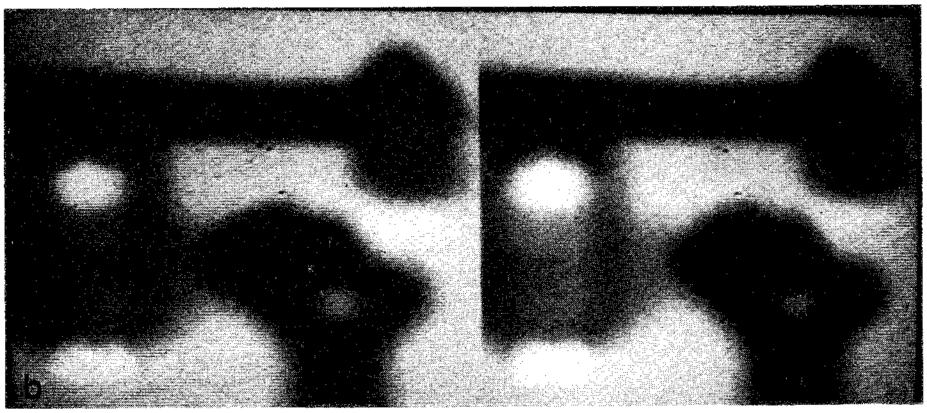

FIG. 5b. Blurred pair.

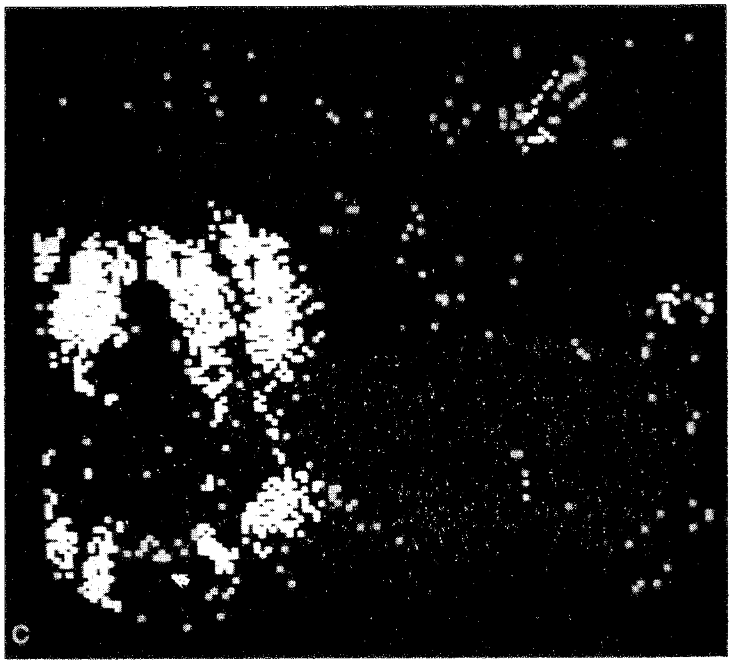

FIG. 5c. Unfiltered pixel classification for Object 1.

For digital imagery, we must adapt the above result to a situation in which the image functions are known only at discrete sample points. Gradients are estimated using a Sobel operator [12] extended to provide gradient direction as well as magnitude. Sampling over time is also discrete. Since the above analysis is only valid if the gradient at a point remains constant over the time between samples, we only consider points where the gradients at time t_1 and t_2 satisfy the restrictions

$$\left| |\mathbf{G}(t_1)| - |\mathbf{G}(t_2)| \right| < a_1$$

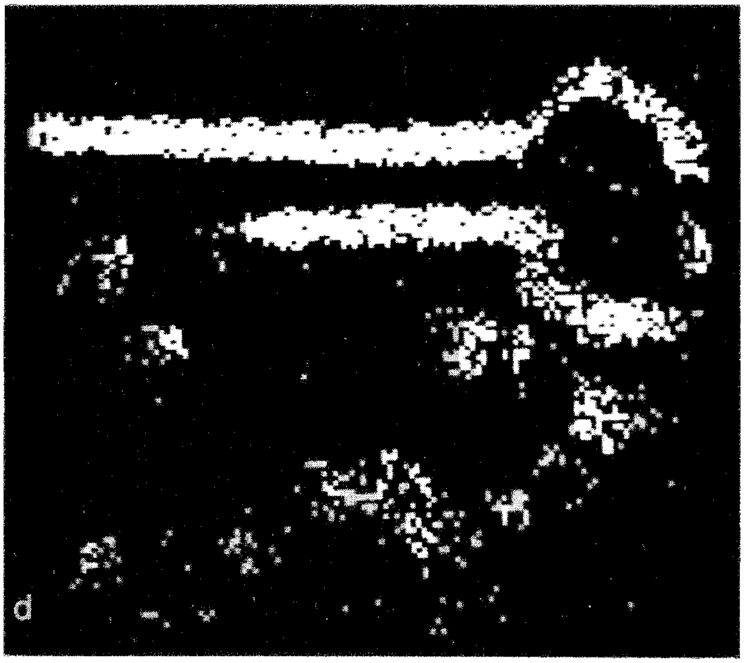

FIG. 5d Unfiltered pixel classification for Object 2.

and

$$|\theta_G(t_1) - \theta_G(t_2)| < a_2$$

or]

$$[2\pi - |\theta_G(t_1) - \theta_G(t_2)| < a_2.$$

Furthermore, we only process points for which the intensity change between sampled frames, ΔI, is sufficiently large:

$$|\Delta I| > a_3$$

This serves to eliminate points where the intensity change is due primarily to scanner noise.

The number of image points satisfying these constraints is greatly increased if the image function is first low-pass filtered. A smoothing operation reduces noise and increases the extent over which gradients remain approximately constant while suppressing those areas where textural variations have a short enough period to cause confusion. In addition, smoothing allows use of the technique on edges which would otherwise be discontinuous points in the image function. The desirable amount of blurring relates to the velocities expected in the imagery. The extent of the blurring function (i.e., the effective width of the point spread function of the blur) should be at least as great as the maximum expected velocity. The value is not critical. Blurring can be easily implemented by defocusing, as we have done here, and thus involves no extra computation.

To determine actual velocity $(|\mathbf{v}|, \theta_v)$, we use a modification of the Hough transform [13, 14]. At each point, ΔI is used as an estimate of *di/dt*. From (4),

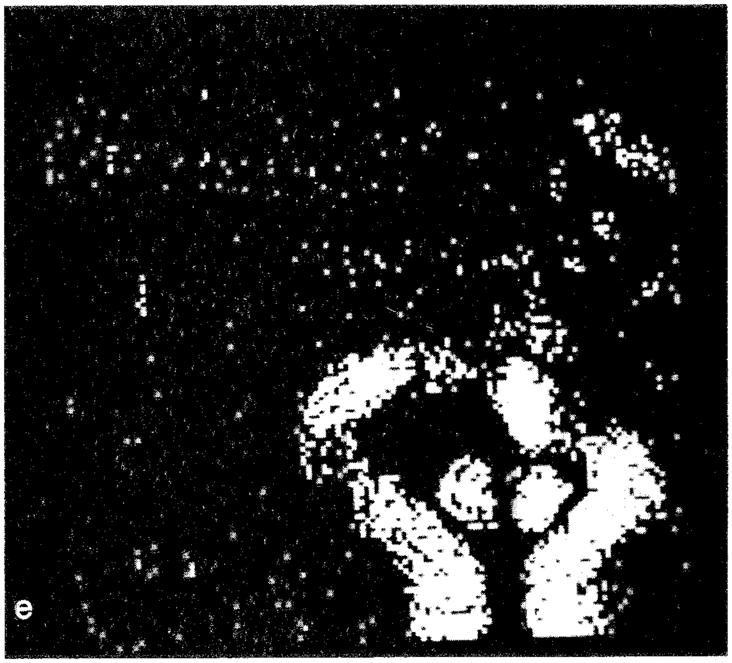

FIG. 5e. Unfiltered pixel classification for Object 3.

the possible values of v that could give rise to ΔI at that point are constrained to lie along the curve

$$|\mathbf{v}| = \frac{v_G}{\cos(\theta_G - \theta_v)} = -\frac{\Delta I}{|G|\cos(\theta_G - \theta_v)}. \tag{5}$$

If the direction of the gradient varies at different points of the moving object, then values at those points will map into different curves in $(|\mathbf{v}|, \theta_v)$ space. The intersection of these curves will uniquely specify **v**.

Solving for this intersection analytically is a complex task, but the Hough transform approach allows a significant computational simplification. $(|\mathbf{v}|, \theta_v)$ space is treated as a discrete array of accumulators. For each image point, ΔI and **G** are used in Eq. (5) to trace a curve through $(|\mathbf{v}|, \theta_v)$ space corresponding to possible velocities. This curve is digitized onto an accumulator grid and then

TABLE 1

Toy Problem

Object	Computed direction	Computed speed	Measured direction	Measured speed
Charlie	0	2	0	2
Snoopy	3.141	2	3.141	2

TABLE 2

The Race

Object	Computed direction	Computed speed	Measured direction	Measured speed
Snoopy	0	1.33	0	1–2
Charlie	0	3.33	0	4

each accumulator corresponding to a point on the curve is incremented. An arbitrary upper bound for $|\mathbf{v}|$ is chosen to deal with situations where $\cos(\theta_G - \theta_v) \approx 0$. When the curves corresponding to all the points in the picture have been entered, a peak in the accumulator array corresponds to a velocity.

In situations involving more than one moving object, different moving surfaces will usually be manifested as distinct peaks in the $(|\mathbf{v}|, \theta_v)$ accumulators. The single most prominent peak in the accumulator array is found. The corresponding velocity is labeled $\mathbf{v}_1$, and all points satisfying the relationship

$$\left| |\mathbf{v}_1| \cos(\theta_G - \theta_{v1}) + \frac{\Delta I}{|\mathbf{G}|} \right| < t$$

for some appropriate threshold t are assigned velocity v_1. The analysis is repeated using the original image data minus those points already assigned a velocity. This procedure continues until no sufficiently well-defined peaks are found in the accumulator array. The assignment of velocities by this technique is subject to error dependent on the order in which peaks are found and the magnitude of the threshold t. This error may be reduced by a final pass in which each image point is assigned to the closest of the candidate velocity vectors.

4. RESULTS

The GITM approach has been applied to a variety of scene sequences with good results. Figures 3 to 5 are typical of the kind of scenes which have been successfully analyzed. All images were digitized from real 3D scenes to a 128 × 128 array of 8-bit monochrome pixels. From this data GITM was able to determine the velocity magnitude to within 1 pixel/frame (Tables 1, 2, and 3) and was able to classify pixels belonging to each velocity fairly accurately. Moreover, the technique works well with scenes containing several moving objects. The scene in Fig. 3

TABLE 3

Moving Tools

Object	Computed direction	Computed speed	Measured direction	Measured speed
Wrench	0	2.67	0	3
Hammer	1.178	1.33	1.571	2
Lens	3.141	2	3.141	3

contains two objects moving in opposite directions: Snoopy to the left and Charlie to the right. Table 1 shows how the computer output of GITM compares with the speed and angle measurements obtained from the blurred images on a CRT using a track ball (to nearest pixel/frame). These two results correspond to the two clearly evident cosine curves in (v_G, θ_G) space shown in Fig. 3c. Using the velocity measurements from GITM the original picture points were then classified according to velocity in the manner explained in the last paragraph of the previous section. Figure 3d shows those points associated with velocity (2, 0) and Fig. 3e those which correspond to velocity $(2, \pi)$. Figures 3f and g result from Figs. 3d and e through the use of a filter which assigns to each pixel the value which most often occurs among the neighboring pixels. This classification of pixels based on their velocity is good enough to suggest a segmentation procedure based on motion.

Figure 4 demonstrates that GITM can distinguish between two velocities which have the same direction. As shown in Table 2, Snoopy and Charlie move in the same direction, but at about 1 pixel/frame and 4 pixels/frame, respectively. The GITM computed results are in close agreement. Figures 4c and d show how well the pixels are classified.

The performance of GITM on three moving objects is shown in Fig. 5 and Table 3. Note that in the figure the lens partially occludes the hammer. The local nature of the method makes it relatively insensitive to changes along the occlusion boundary. GITM classifies the pixels fairly well, and the velocity calculations are reasonably close to those measured with the exception of the hammer velocity angle. This error is due to blurring of peaks in $(\mathbf{v}, \theta_v)$ space and suggests a possible weakness of the procedure as a measurement method for a large number of moving objects. However, for scene analysis purposes the accuracy is sufficient, and the classification capability will prove quite useful.

5. CONCLUSIONS

This paper has described a nonmatching procedure (GITM) which can detect and quantify the velocities present in a sequence of digital images containing several moving objects. Results have shown that this procedure can handle occlusion, works on blurred images, can determine the velocity angle within $\pi/8$, and can determine the velocity magnitude within about 1 pixel/frame. Furthermore, a method was described which uses the velocities obtained from GITM to classify the pixels in the image according to velocity. This classification is good enough to suggest a possible motion-based segmentation of the scene into objects.

REFERENCES

1. N. Badler, "Temporal Scene Analysis: Conceptual descriptions of object movements," Tech. rep. 80, Dept. of Computer Science, Univ. of Toronto, February 1975.
2. J. K. Aggarwal and R. O. Duda, Computer analysis of moving polygonal images, *IEEE Trans. Computers* **C-24**, 1975, 966–976.
3. J. L. Potter, Velocity as a cue to segmentation, *IEEE Trans. Syst. Man. Cybern.* **SMC-5** 1975, 390–394.

4. J. A. Leese, C. S. Novak, and V. R. Taylor, The detection of cloud pattern motions from geosychronous satellite image data, *Pattern Recognition* **2**, 1970, 279–292.
5. E. A. Smith and D. R. Phillips, Automated cloud tracking using precisely aligned digital ATS pictures, *IEEE Trans. Computers* **C-21**, 1972, 715–729.
6. R. Jain, D. Militzer, and H. H. Nagel, Separating non-stationary from stationary scene components in a sequence of real world TV-images, In *Proceedings of the 5th Int. Joint Conf. on Art. Intel.*, pp. 612–618, August 1977.
7. D. C. Hogg, A methodology for real time scene analysis, In *Proceedings of the 5th Int. Joint Conf. on Art. Intel.*, p. 627, August 1977.
8. W. N. Martin and J. K. Aggarwal, Dynamic scene analysis, *Computer Graphics Image Processing* **7**, 1978, 356–374.
9. H. P. Moravec, Towards automatic visual obstacle avoidance, In *Proceedings of the 5th Int. Joint Conf. on Art. Intel.*, p. 584, August 1977.
10. D. I. Barnea and H. F. Silverman, A class of algorithms for fast digital image registration, *IEEE Trans. Computers* **C-21**, 1972, 179–186.
11. J. O. Limb and J. A. Murphy, Estimating the velocity of moving images in television signals, *Computer Graphics Image Processing* **4**, 1975, 311–327.
12. R. O. Duda and P. E. Hart, "Pattern Classification and Scene Analysis," pp. 271–272, Wiley, New York, 1973.
13. A. Rosenfeld, "Picture Processing by Computer," pp. 151–152, Academic Press, New York, 1969.
14. R. O. Duda and P. E. Hart, Use of the Hough transformation to detect lines and curves in pictures, *Commun. ACM* **15**, 1972, 11–15.