IEEE International Workshop on Intelligent Motion Control
Boğaziçi University, Istanbul, 20-22 August 1990.
Edited by Okyay KAYNAK. IEEE Cat. No: 90TH0272-5

INVITED PAPER

DYNAMIC ANALYSIS OF IMAGE MOTION FOR VEHICLE GUIDANCE

Peter J. Burt, James Bergen, Rajesh Hingorani,
Shmuel Peleg, and P.Anandan

David Sarnoff Research Center
Subsidiary of SRI International
Princeton, NJ USA 08543-5300

Abstract

We outline a selective analysis approach to motion estimation that promises to provide the precision and efficiency required for autonomous vehicle guidance. Efficiency is achieved by implementing computations within a hierarchical (pyramid) structure, and by restricting these computations to selected regions of the scene. These *analysis regions* are moved dynamically over the scene as a sequence of *focal probes*, much as a human driver move his or her eyes and shifts visual attention. Precise motion estimates are obtained by fitting models comprising one or two rigidly moving surfaces to the image data within each focal analysis region. Differential motion within the region separates foreground from background objects, while overall region motion relative to the focus of expansion determines distance from the observer. High level *intelligent control* directs the focal probes.

We show through examples that model-based motion estimation can be used to detect obstacles in the road, and to discriminate such obstacles from road markings. High level intelligent control is described briefly.

I. Introduction

Image motion analysis is potentially an important source of visual information for vehicle guidance. For instance, it can be used to detect possible obstacles in and near the road through motion parallax, and unlike many currently used sources of visual information, it does not rely on objects having known shapes or distinctive features. However, the parallax motion between an obstacle and the road can be small compared to other components of image motion common to both objects. This means motion estimates must be highly precise if they are to serve vehicle guidance. Furthermore, estimates must be obtained over dense arrays of image points in the direction of travel or potential obstacles may be overlooked. For practical applications, these computations must be performed in real time, using hardware of modest size and cost.

Current visual guidance systems analyze motion either by tracking prominent image features over successive image frames or by computing image flow. By tracking painted lane markings in the road, feature-based methods have proven effective for determining the vehicle's own motion. However, because motion estimates are obtained only at scattered points in the field of view, these techniques can not be relied upon for detecting all obstacles. Flow analysis techniques compute dense arrays of estimates, but current approaches often require excessive computation and do not yield precise, robust motion estimates. Furthermore, flow analysis is 'bottom-up,' based on image data, and does not make use of a priori knowledge of motion in the scene, such as that it is due to the road surface or previously detected objects.

Model-based analysis can provide the precise motion estimates required for obstacle detection. But the complete interpretation of real world scenes in terms of surfaces and objects is not feasible: the models are too complex, and computations are prohibitive.

Fortunately, effective vision for driving does not require complete understanding of the scene. While highly precise estimates of motion parallax may be essential to detect an object in the road, much of the scene can be analyzed at a much lower resolution, or can be virtually ignored by the vision system. Model-based analysis can be restricted to local regions of the scene that are most likely to contain information required for the driving task.

In this paper we outline a vision system designed to achieve precision and efficiency through selective analysis. Analysis is split into two distinct but interacting levels, one local and one global. These differ in the types of computations performed and the representations used.

At the local level motion analysis is performed within selected regions of the scene. These analysis regions are moved over the scene to gather information required for the driving task, much as a human driver moves his (or her) eyes and shifts visual attention. Because these regions are limited in size, motion within each region is generally quite simple, often a single surface (possibly tilted in depth), or two surfaces, one foreground the other background. Precise motion estimates are obtained locally by fitting the appropriate surface model to the image data.

Local analysis is implemented within a pyramid structure. This provides a general framework for implementing highly efficient motion analysis algorithms, and for specifying the size and resolution of image regions in which the analysis is performed.

Motion estimates obtained through local analysis are interpreted as objects and distances at the global analysis level. The system maintains a representation of its environment in terms relevant to vehicle guidance, as in the 4D dynamical model introduced by Dickmanns and Graefe [1]. This includes road descriptors and parameters of the observer's own motion, as well as a record of each potential obstacle detected in the road. Based

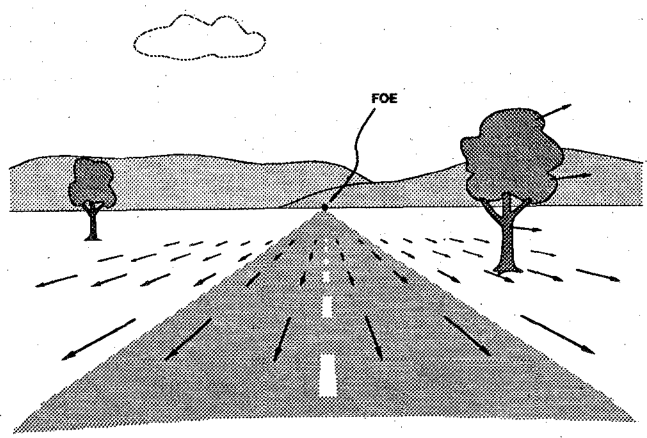


Figure 1: Image motion as observed from a moving vehicle.

on requirements of the driving task and on information in the world model, 'intelligent' control processes at the global level determine where to direct local analysis in the scene, The resulting local motion estimates are then used to update and refine the world model.

These *dynamic analysis* techniques focus processing resources on objects and events that are critical to vehicle guidance. Together they can reduce data and processing required for visual guidance by factors of of 1,000, 10,000, or more, without a sacrifice in performance.

We begin the present discussion with a statement of the motion estimation problem, then outline a framework for dynamic motion analysis. We describe algorithms used for focal motion estimation and show with several examples that these algorithms can be used to detect obstacles. High level control will be described in general terms, but has not yet been implemented.

II. Problem Statement

Figure 1 illustrates a typical pattern of image motion observed from a moving vehicle. Motion at each point in the scene includes components due to camera rotation and camera translation. In this example rotation is assumed to be zero. Translation is in the direction of the focus of expansion, the FOE. All stationary objects in the scene appear to move radially away from the focus of expansion, with speeds that increase with the angular separation from the FOE, and decrease with distance from the observer. The relative distances of objects can be recovered through a simple geometric relationship given their observed motion and the location of the FOE. Absolute depth is recovered if the speed of the observer's motion towards the FOE is also known.

In practice, motion estimates are difficult to obtain with the precision required for obstacle detection. It is most important for the vision system to analyze motion near the FOE, because this is in the direction of travel. But this is also the direction in which parallax motion becomes vanishingly small. Furthermore, image motion near the FOE is dominated by components of motion due to camera rotation. Such rotations result from camera pan and from vibrations as the vehicle moves over uneven pavement. Small obstacles in the road present the greatest challenge because their parallax motion differs only slightly from that of the road against which they are viewed. These difficulties are exacerbated by the fact that current approaches to motion estimation are prone to error. Errors may be due either to noise in the video signal, or to lack of sufficient pattern detail on which to base motion estimates.

However, several characteristics of the vehicle guidance task allow simplifications of motion analysis. Full precision is needed only in the area of the camera's image that represents the road. Outside this area, analysis can be performed at lower resolution. Furthermore, the highly precise analysis required to detect obstacles can often be performed locally, based on *differential* motion within restricted image regions. Image motion within a local region will generally represent a single physical surface undergoing rigid motion. This motion can be recovered by fitting a planar surface model to image data. Deviations from the model then indicate the presence of points in front of the surface, or of a discontinuity between foreground and background surfaces. In either case the discrepancy indicates that an obstacle may be present. Further analysis can be directed to just those regions where potential obstacles have been detected, to confirm the detection and determine the size and distance to the obstacle.

Before local estimates can be interpreted as depth, estimates obtained within diverse image regions must be combined to determine observer motion. This more global analysis need not be based directly on image data, but on parametric descriptions of motion within local regions.

When rotation is not assumed to be negligible the form of local motion becomes more complex. However not all components of motion are equally important for obstacle detection and vehicle guidance. This is suggested in Figure 2. Here points A and B occur within a local region R of the scene. To sim-

a.

b.

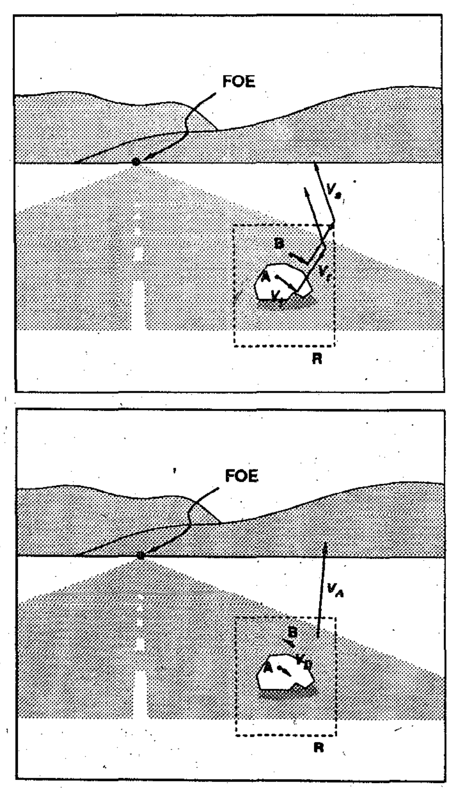


Figure 2. Points A and B within local region R fall on a foreground and background surface, respectively. (a) The components of motion are shown for these points separately. (b) The components of motion are shown as differential motion within the region and average motion of the region itself.

plify, we assume the points are at equal (angular) distances from the FOE. Each point has a velocity in the image that combines three components: V_t, the motion radially away from the FOE that is due to camera translation along the road, V_r, the motion perpendicular to V_t that is due to camera rotation about the direction of the FOE, and V_s, the component of motion due to camera rotation about an axis perpendicular to the direction of the FOE. Of these components, only V_t carries information about depth. The others vary in systematic, predictable ways across the image.

The V_t components of motion for points within the local region R can be represented in terms of an average region motion and internal differential motion, Figure 2b. The region motion, V_A, the distance from the region to the FOE, and the speed of the vehicle can be used to determine the mean distance of objects within the region from the observer. The differential component, V_D, then indicates the separation of the points in depth. Differential motion is critical for discriminating foreground from background surfaces, and hence for detecting obstacles in the road.

III. Framework for Dynamic Analysis

A human driving a car continually moves his (or her) eyes to gather information required for the driving task. This is illustrated in Figure 3. At a given moment in time he may look at the road, then at an oncoming car, then at a road sign, then at the road again. The driver fixates an object to see it in detail, and tracks it with his eyes to stabilize its image on his retinae. While his central (foveal) vision provides pattern detail, his peripheral vision monitors a wide field of view at low resolution to detect unexpected events, and to guide foveal vision.

In computational terms, the foveal organization of the eye serves two purposes: by restricting precise analysis to local regions it limits the data that the system must be process, and by isolating and stabilizing image patterns it simplifies the analysis that the system must perform. To gain these benefits without losing sensitivity, eye movements must be guided by high level, intelligent control processes: the human understands the driving task, and moves his eyes to gather information required for that task.

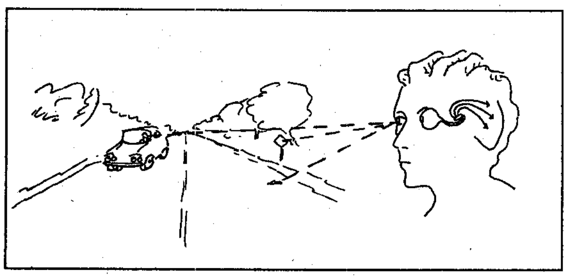


Figure 3. Eye movement strategies used by a driver.

We adopt a framework for motion analysis that can be motivated in part by analogy to human vision. Analysis is organized into distinct, but interacting *local* and *global* stages. Local analysis is further organized as a sequence of *focal probes*. Each probe examines motion within a restricted local *analysis region* of the scene. Successive probes move over the scene to gather and refine motion information. The size and resolution of the local analysis region is changed from probe to probe. Typically, analysis is first performed within a large region, at low resolution, to determine the general motion in the scene. Then analysis is moved to progressively smaller regions, at correspondingly higher resolutions, to examine critical image motions in greater detail. The sequence of probes is controlled dynamically to gather information required for the driving task.

Motion estimates are obtained by fitting a model composed of a small number of moving surfaces, typically one or two, to the image data within the local analysis region. Differential motion is estimated with greatest precision, to allow the system to discriminate foreground from background motion.

Local analysis can be implemented conveniently within a pyramid structure. This is suggested in Figure 4. The analysis regions corresponding to a sequence of focal probes are shown as rectangles superimposed on the scene, on the left. Motion analysis is performed on subarrays of image data within the pyramid representation of the image, as shown on the right. Shaded areas at each pyramid level correspond to the data arrays used in the successive focal probes. In this example, the sequence of probes begins at a low resolution level to pyramid where the array of

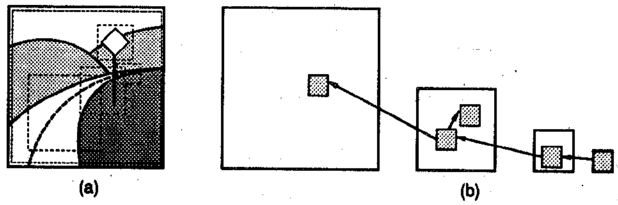


Figure 4. Analysis of a road scene as a sequence of focal probes (a) based on data from a multiresolution image pyramid (b).

data corresponds to the entire scene. Analysis then moves to successively higher resolution levels of the pyramid, where the respective subarrays of data correspond to progressively smaller regions of the scene. In addition to providing a framework for controlling the position, size, and resolution of the analysis regions, the pyramid supports fast analysis algorithms used in motion estimation.

Components of the motion analysis system are shown in Figure 5. A preprocessing stage performs initial image digitization, enhancement, and pyramid construction. At the local analysis stage, estimates of motion are obtained within selected regions through the model fitting process. At the global stage, the system interprets motion estimates as depth, and determines where to 'look' next, where to direct successive focal probes.

A dynamical world model, much like that of Dickmanns and Graefe [1], is maintained at the global analysis level. This includes motion parameters for the vehicle, for the road, and for objects that have been detected on the road. The intelligent control process directs successive probes based on requirements of the driving task, guided by information in the world model, and in response to unexpected events that are detected in the scene.

For each successive probe, the control process specifies the position and size of the analysis region R, within the image domain, and the maximum resolution at which image data should be examined. It also decides what surface model M should be used within the region, and it provides appropriate constraints to be imposed on the estimation process as well as an a priori estimate V_0 of the motion expected in the region. The local stage then performs the specified analysis using a model-based iterative refinement procedure. In effect, this obtains precise motion estimates by aligning the image pattern in successive image frames within the analysis region R. The local stage returns refined estimates of the model parameters, V_k (after the k^{th} iteration), as well as confidence information, C, to the global stage, where these are used to estimate depth, detect obstacles, and update the system's world model.

In the following section we describe algorithms used in local

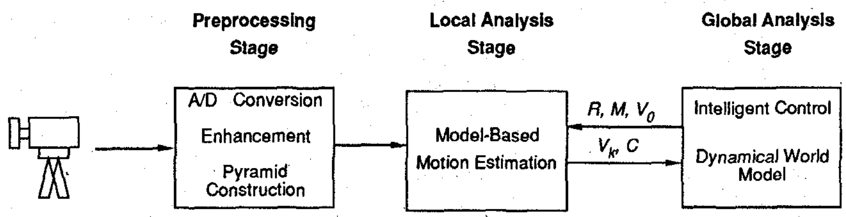


Figure 5. Components of the vision system.

analysis stage of this system. Again, intelligent control and other analysis at the global stage are beyond the scope of the present paper.

IV. Motion Estimation

In formulating analysis within each local analysis region it is assumed that motion corresponds to a single surface in depth, or to two surfaces, one foreground and one background. Motion is then estimated through a model fitting process. This basic assumption may prove incorrect for any given focal probe if motion within the analysis region is too complex. In this case a new smaller region is selected. The process may also fail if the region lacks sufficient image detail on which to base a robust motion estimate. In this case a new larger region is selected.

We have developed several models for describing motion within the local region. These include
(a) a single-component affine motion model [2],
(b) a two-component foreground/background model [2,3], and
(c) a single component smoothly varying model [4].

These suffice for the basic discriminations required for vehicle guidance, to detect small obstacles on the road, or larger objects viewed against a more distant background. In practice, all of these models could be implemented within a single vision system, with different models selected for the analysis of different regions of the scene depending on local motion characteristics.

Tilted Surface Model

Suppose we wish to estimate motion between frames $F(t-1)$ and $F(t)$ of the motion sequence. That is, we wish to find $V(x,y) = (v_x(x,y), v_y(x,y))$ such that

$$F(x,y,t) = F(x-v_x, y-v_y, t-1)$$

within region R.

If V is small then a Taylor series approximation may be used:

$$F(x-v_x, y-v_y, t-1) \approx F(x,y,t) - v_x F_x - v_y F_y - F_t, \quad (1)$$

where

$$F_x = \frac{\partial F(x,y,t)}{\partial x}, \quad F_y = \frac{\partial F(x,y,t)}{\partial y}, \quad \text{and} \quad F_t = \frac{\partial F(x,y,t)}{\partial t}.$$

In the single-component affine model the x and y components of velocity are given by

$$v_x(x,y) = ax + by + c,$$

and

$$v_y(x,y) = dx + ey + f.$$

To fit the model to the data we find the values of the parameters that minimize the error:

$$Err = \sum_{x,y \in R} \left(F(x,y,t) - F(x-v_x, y-v_y, t-1)\right)^2$$
$$\approx \sum_{x,y \in R} \left(v_x F_x + v_y F_y + F_t\right)^2.$$

Six simultaneous equations are obtained by setting the derivatives of the error with respect to each of the model parameters to zero. When solved, this provides expressions for each of the parameters in terms a set of image moments of the general form:

$$K_{mnjk} = \sum_{x,y \in R} x^m y^n F_j F_k. \quad (2)$$

See, for example, [3].

The above computation can give precise results only if the frame-to-frame displacement is small, generally less that one pixel. Otherwise, the Taylor series approximation used in solving for the optimal model parameters are invalid. Since the motions of interest in vehicle guidance will often be much larger than this, a successive refinement procedure is used, Figure 6. For a given pair of image frames and analysis region, this generates a sequence of estimates of motion V_1, V_2, ... V_k. Before each estimate is obtained, the first image is "warped" in accordance with the prior estimate of motion to bring it into rough alignment with the second image. Motion estimation is performed between the warped first image and original second image to obtained an estimate of the residual motion, ΔV. Let $F_m(t-1)$ be frame $F(t-1)$ warped by motion V_m:

$$F_m(x,y,t-1) = F(x-v_{x_m}, y-v_{y_m}, t-1).$$

Analysis begins with an a priori estimate, V_0, of the motion within R. This may be based on previous estimates obtained in the region or in neighboring regions. The model fitting procedure outlined above is performed between $F_0(t-1)$ and $F(t)$ to obtain an estimate of residual motion, ΔV_1. This is added to the prior motion, V_0, to obtained the refined estimate, V_1. The first image is again warped and the estimated residual is obtained. These steps are repeated until a desired level of accuracy is obtained.

In practice these analysis steps are best carried out as coarse-fine refinement within a pyramid structure. The initial estimation step is performed at a low resolution level of the pyramid where the sample distance is large compared to the expected motion in the scene. Then, as the alignment process proceeds, and residual velocities become small, the computation is moved to progressively higher resolution pyramid levels, ending with the resolution specified for the analysis region. Convergence is often quite rapid.

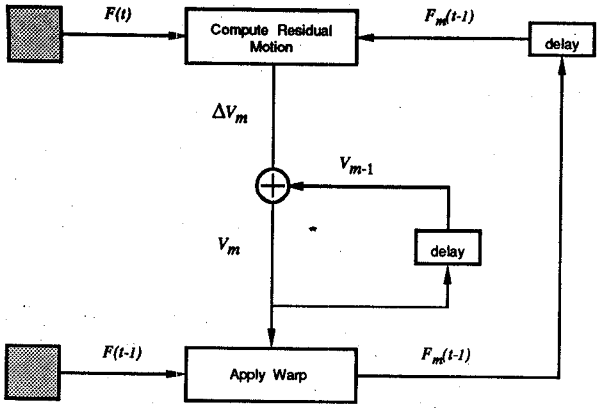


Figure 6. Model-based motion estimation with alignment.

Foreground/Background Model

The above procedure can now be extended to estimate the motions of multiple surfaces within the analysis region. The model-based alignment process is first applied to estimate one of the motions. Image samples that move in accordance with the first motion are then removed from consideration, and the procedure is applied to the remaining samples to determine the second component of motion. The analysis may alternate between components several times to obtain precise estimates.

When estimating motion of foreground and background surfaces, it is expedient to represent each surface motion as simple translation rather than as full affine motion. This reduces the number of parameters to be estimated for each surface, thus improving stability. And it is an adequate model for situations that occur in the scene where the separation of foreground and background surfaces is large compared to the extent that either surface is tilted in depth within the analysis region.

The single-surface motion estimator tends to provide a good estimate of one motion even when multiple motions are actually present. To understand why this is true, consider a case in which two moving patterns are present, but one pattern motion is small, and within the range represented by the Taylor series approximation, Eq. 1, while the other pattern motion is large, beyond this limit. Both surfaces contribute to the integrals, Eq. 2, but only the smaller motion makes a coherent contribution, while the contribution of the larger motion is effectively noise. The smaller motion then tends to dominate the alignment process, and is selected by the analysis.

When analysis is performed within a pyramid structure this selective mechanism can isolate one motion even when both have large and roughly equal velocities. The initial stages of the analysis, at low resolution, tend to estimate an average of the two motions; however, as the successive alignment procedure reduces net velocity, and analysis is shifted to higher resolution, the differences between the velocities becomes progressively more significant. The procedure for estimating residual velocity then tends to select and locks onto just one motion.

The above tendency is accelerated and made more precise by refining the *support region* for the dominant motion component as analysis proceeds. Points within the region R that are least consistent with the current estimate of surface motion are discarded. These 'outliers' tend to be on the pattern moving with a different motion. The remaining points are more accurately aligned to the given moving pattern. Once one motion is obtained, the points that contribute to that motion are set aside, and the algorithm is repeated, starting with the remaining points.

Smoothly Varying Surface Model

The optic flow equation corresponds to the translation component of surface motion and can be used to estimate a separate motion vector at each point in an analysis region. Let $V(x,y) = (v_x(x,y), v_y(x,y))$ be the estimated velocity for images $F(x,y,t-1)$ and $F(x,y,t)$. Then at each point (x,y) the flow is obtained by solving these two equations:

$$\begin{aligned} v_x \sum F_x^2 + v_y \sum F_x F_y + \sum F_x F_y = 0, \\ v_x \sum F_x F_y + v_y \sum F_y^2 + \sum F_y F_t = 0. \end{aligned} \quad (3)$$

This basic form of the optic flow computation has several shortcomings. Because a separate estimate of motion is obtained for each image point, and each such estimate is based on very limited image data in the neighborhood of that point, estimates tend to be unreliable. The derivation is based on a Taylor approximation to image intensities, and thus assumes that frame-to-frame displacements are small, generally well under a pixel distance. Perhaps the most important shortcoming is that optic flow does not model discontinuities in motion, as at the boundaries between foreground and background surfaces.

The first two of these limitations can be largely overcome imposing 'smoothness' constraints, by refining estimates through a successive alignment procedure, and by performing computations within a pyramid structure. Then the flow approach is quite effective for estimating complex patterns of motion, provided velocity varies smoothly over the analysis region, and there are no discontinuities.

Computations again take the form shown in Figure 6. Let $V_0(x,y)$ be an a priori estimate of image motion. (Lacking better information this is taken to be zero.) Analysis begins at a level of the pyramid at which errors in motion with respect to V_0 are expected to be less than a sample distance. The first image of the pair, $F(t-1)$, is warped towards the second in accordance with this estimate of motion, to form $F_0(t-1)$. The optic flow equation, Eq. 3, is used to estimate residual motion ΔV_1. A low-pass filter, w, is applied to the residual flow field to enforce a smoothness constraint, then the prior estimate and the smoothed residual are summed to form an updated estimate of flow. These steps are repeated until the desired precision is obtained. At iteration k the flow equation is used to compute residual motion ΔV_k between $F_{k-1}(t-1)$ and $F(t)$. Then

$$V_k = V_{k-1} + w * \Delta V_k.$$

Successive refinement steps move to higher resolution levels of the pyramid.

V. Examples

We now provide several examples to illustrate the ability of local model-based motion estimation to detect obstacles.

1. Block in Road

The first example is shown in Figure 7a. Here a stone block, about 10 cm in height, is observed at the side of the road. This must be detected as a three dimensional obstacle based on its parallax motion with respect to the road. The scene also contains dirt and shadow marks on the road that should not be confused with obstacles.

The detection task is challenging because the block is relatively small and is viewed at a large angle with respect to the ground plane, so that its parallax motion with respect to the road is small. The block is situated in the scene at roughly 30 degrees from the focus of expansion, so that the common motion

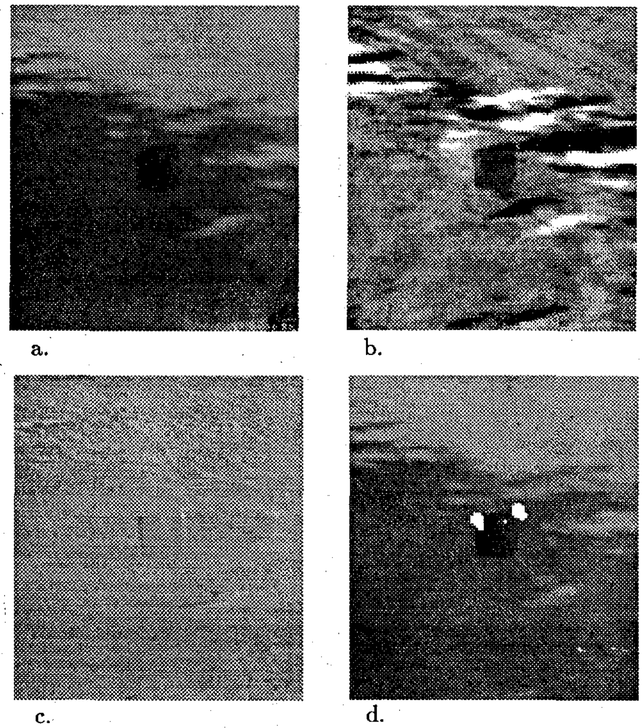


Figure 7. (a) A stone block by the side of the road. (b) Difference between successive images showing rapid camera motion. (c) Difference after estimating and compensating for motion of the ground plane. (d) Detection of highest points in the scene.

of the road and block are large compared to the differential motion between these objects. In addition the camera was panning rapidly at the time this image was obtained.

Figure 7b shows the difference between two frames (four frames apart in the image sequence). Note that displacements are uniformly large. A single surface affine model was applied to the region shown in this example (128 by 128 pixels). Because the road constitutes the major portion of the analysis region, the surface fit corresponds to the road surface. Figure 7c shows a difference image after warping the first image towards the second in accordance with this motion estimate.

Several points are of interest in these results. Most of the patterns on the road surface are cancelled, indicting that the estimate of road motion is quite good. However, discrepancies remain, particularly towards the top of the image. These may reflect limitation of the affine model to represent motion over the analysis region in this case. Residual difference values are most apparent along the sharp boundary between the shadowed and sunny parts of the road. Note that the left and right vertical edges of the block are apparent in the difference image and that the width of these edge difference patterns forms a wedge that grows wider towards the top. This reflects the fact that the parallax motion of the block with respect to the ground plane increased with height. Finally, Figure 7d shows regions of the scene at which motion is most different from the ground plane, indicating the presence of a possible obstacle in the road.

2: Postman

In this example a postman is seen walking across a country road, Figure 8a. The postman's truck is parked on the right, and he is approaching a mailbox on the left. The road is narrow,

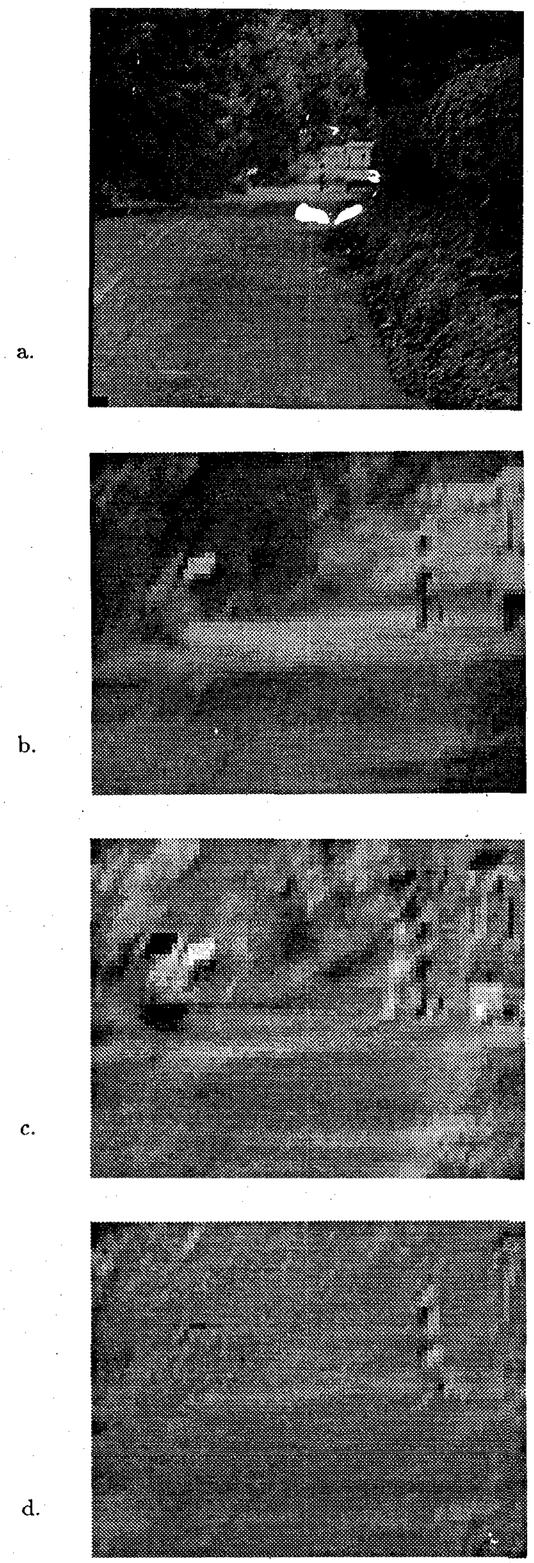
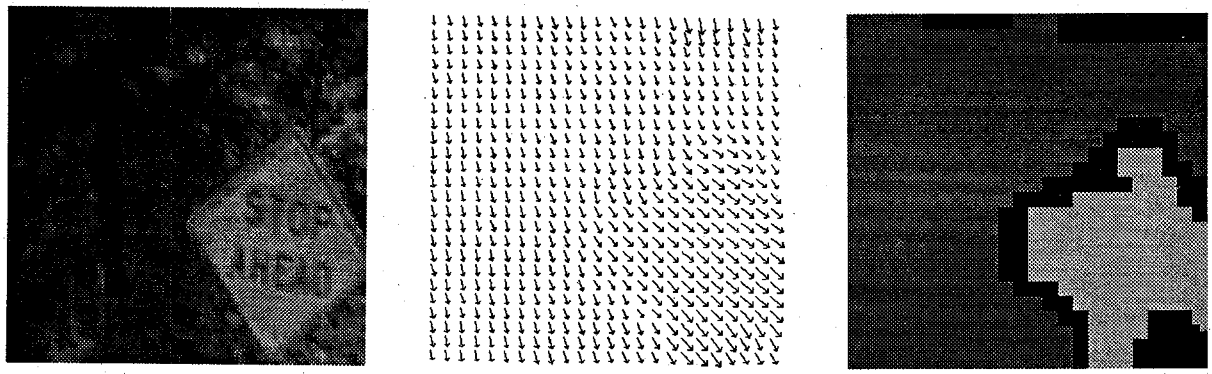


Figure 8. (a) Postman crossing road. (b) Region selected for motion analysis. (c) Difference between successive image frames showing motion due to the camera and postman. (d) Difference after detecting and compensating for surface motion within the analysis regions.



a. b. c.

Figure 9. (a) Road sign viewed against a line of trees by the side of a road. (b) Estimate of smooth surface motion obtained through a coarse-fine optic flow computation. (c) Segments obtained from the flow field.

twisted, and hilly. Trees and bushes line either side and overhang the road. The camera is on a vehicle that is approaching the scene quite rapidly. These circumstances mean that motion flow is complex.

In order to detect potential obstacles in the road, the system selects (here selection is done manually) an analysis region centered on the road in the distance. The analysis region (80 by 64 pixels) is shown in Figure 8b. Figure 8c shows a difference image between successive frames, without motion compensation. Note that the motion is generally significant over the region. Analysis was performed on this region using the single surface affine model. Figure 8d shows a difference image formed after warping the first image towards the second in accordance with the resulting motion estimate. Much of the background and the road are compensated by this process. The postman is visible, as are objects near the side of the road.

3: Road Sign

The final example is a sign observed at the side of the road (96 by 96 pixels), Figure 9a. This sign was quite close to the focus of expansion, which was roughly at the left hand edge of the region shown. The trees that form the background recede into the distance on the left, but are at the same distance as the sign on the right. The objective of analysis in this case is to use motion parallax to separate the sign from the tree line.

The scene was analyzed by first estimating a smooth surface motion through the application of the coarse-fine optic flow algorithm. The resulting optic flow field was then segmented into two surfaces using a split and merge procedure [5]. Figure 9b shows the optic flow. Figure 9c shows the two surface segmentation. Note that the resulting regions correspond to the sign and the trees. The black areas of the segmented image show portions of the flow field that did not fit either surface motion (due to inevitable errors along the motion boundary) and hence were disregarded.

VI. Summary and Discussion

There are several key features in the system we have described for vehicle guidance. Analysis is organized into two distinct but interacting levels, one local and one global. We believe this will simplify the motion analysis problem by restricting requirements for highly precise analysis to the local level. Locally computed differential motion serves to detect motion boundaries between foreground and background surfaces, and hence to detect potential obstacles in the road. Global analysis provides information regarding observer motion that is required to interpret local differential motion.

Local analysis is model-based: image motion within each analysis region is interpreted as a single surface in coherent motion, or as two surfaces in a foreground/background relationship. The model is fit to image intensity data over the entire analysis region. It is not limited to selected image features. This allows for the computation to be robust and the estimates precise. Precision is further improved through a successive refinement procedure.

The local analysis regions are moved dynamically over the scene, as a sequence of focal probes. In this way the system focuses its computing resources on regions most likely to contain critical information. Computations are implemented within a pyramid structure, so that the size and resolution of the analysis region can be controlled from probe to probe.

Effective use of selective analysis techniques depends on the availability of intelligent control procedures to decide where to direct probes and how to interpret results. We have not developed this control aspect of the system.

There are many aspects of a motion analysis system for vehicle guidance that we have not discussed here. For example, a system must estimate its own motion, or equivalently, the position of the FOE and rotation parameters of the camera. The determination of the FOE can be based at least to first approximation on inertial sensors on the vehicle. The estimate of this direction can be further refined by analyzing the pattern of motion observed within the scene. The components of motion due to camera rotation are common to all points in the image, so can be estimated directly from observed motions. The differential motions in each local analysis region point to or away from the FOE, so estimates of differential motion at scattered points in the scene can be used to locate the FOE itself.

A somewhat different strategy may be required near the FOE. Since differential motion is small, it should be measured over an extended time base, over multiple frame times. The direction of the FOE and other parameters of observer motion are represented in the dynamical world model, and are refined over time. This provides stability to estimates obtained over the longer time intervals, averaging out effects of variations that may be due to vibrations.

The focal probes and alignment techniques we describe are implemented within digital analysis algorithms. They are an electronic processing mechanism analogous to eye movement strategies in human vision. An additional component of mechanical camera tracking is also appropriate in real world applications. Camera motion is required to extend the effective field of view, and to reduce camera blur.

References

1 E. D. Dickmanns and V. Graefe, "Dynamic monocular machine vision," *Machine Vision and Applications*, pp. 223-240, 1988.

2 P. J. Burt, J. R. Bergen, R. Hingorani, R. Kolczynski, W. A. Lee, A. Leung, J. Lubin, and H. Shvaytser, "Object tracking with a moving camera, an application of dynamic motion analysis," In *IEEE Workshop on Visual Motion*, pp. 2-12, 1989.

3 J. R. Bergen, P. J. Burt, R. Hingorani, and Shmuel Peleg, "Transparent-motion analysis," In *Proceedings of the First European Conference on Computer Vision*, pp. 566-569, 1990.

4 J. R. Bergen and E. H. Adelson, " Hierarchical, computationally efficient motion estimation algorithm," *J. Opt. Soc. Am. A., Vol. 4*, pp. 35, 1987.

5 T. Pavlidis, *Structural Pattern Recognition*. Springer, New York, 1977.