Dynamic-Scene and Motion Analysis Using Passive Sensors

PART I: A Qualitative Approach

Bir Bhanu, University of California at Riverside
Ramakant Nevatia, University of Southern California
Edward M. Riseman, University of Massachusetts

THE DRIVE SYSTEM USES A QUALITATIVE SCENE MODEL AND A FUZZY FOCUS OF EXPANSION TO ESTIMATE ROBOT MOTION FROM VISUAL CUES, DETECT AND TRACK MOVING OBJECTS, AND CONSTRUCT AND MAINTAIN A GLOBAL DYNAMIC REFERENCE MODEL.

MOTION BECOMES A NATURAL component of visual information processing whenever moving objects are encountered in the environment. To act intelligently in the presence of potential hazards or navigate in a traffic environment, information on actual motion in the scene is indispensable. Autonomous mobile robots must know about the presence and behavior of moving objects to determine appropriate reactions. But because the sensor itself is moving, recognizing independent motion of objects in the image is harder: Stationary objects generally appear to be moving, and moving objects might appear to be stationary. Before the robot can draw any useful conclusions, it must determine the effect of the sensor's motion on the image.

Mobile robots depend on motion analysis to avoid obstacles, recognize landmarks, determine location, acquire models, and detect and track moving objects. For example, depth from motion analysis is an important information source for any navigational system, whether or not that system has domain knowledge to start with. Depth from motion analysis lets robots use sensors with a wide field of view and fast data rates. High-quality optical sensors are much less expensive than active sensors such as laser range finders, and they allow ranging to surfaces that might not be suitable for active ranging. In addition, even the most advanced active sensors do not provide the large field of view and high frame rates that many practical applications need.

Dynamic-scene and motion analysis uses information from a sequence of images obtained from a moving sensor; for example, a forward-looking video camera rigidly mounted on an autonomous mobile robot. The major goals of this work include

- recovering a sensor's motion parameters (decomposing rotation and translation components);
- detecting independently moving objects and recovering their motion parameters;
- performing correspondence of features such as points, lines, and regions between successive frames;
- measuring the depth of three-dimensional environmental objects, including points, lines, and surfaces (accurate passive ranging);
- computing the optical flow of features between successive image frames (the two-dimensional instantaneous velocity of image pixels on discrete frames); and
- integrating motion and depth information with environmental models in vehicular navigation systems.

This article describes some of the dynamic-scene and motion analysis techniques developed at Honeywell to support the DARPA Strategic Computing program's Autonomous Land Vehicle effort. Our approach, called Dynamic Reasoning from Integrated Visual Evidence (DRIVE), addresses the key problems of estimating

0885/9000/92/0200-0045 $3.00 © 1992 IEEE

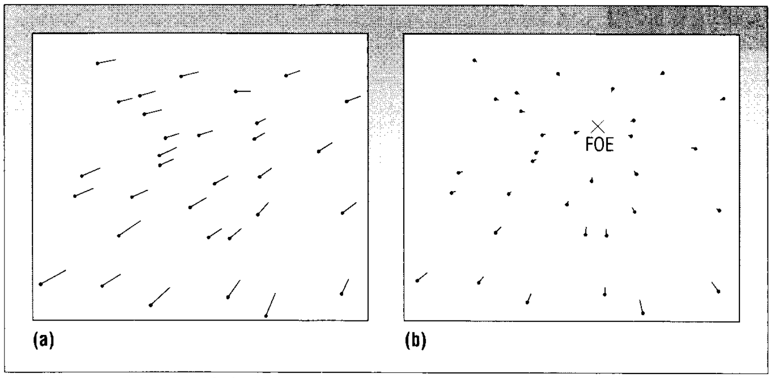


Figure 1. General-motion field: (a) a typical displacement field obtained from a moving camera undergoing translation and rotation; (b) the displacement field after removing the rotational component of motion. The approximate location of the FOE is marked by a cross.

robot motion from visual cues, detecting and tracking moving objects, and constructing and maintaining a global dynamic reference model.[1] A companion article describes other work in dynamic-scene and motion analysis performed at the University of Massachusetts and the University of Southern California (see p. 53), and discusses important general issues and needed technical advances.

Focus of expansion

We can describe a robot's movement as a combination of translation and rotation. When both components are present, far more complex image trajectories are produced. Only translation supplies information about the environment's depth and structure, so the first step in motion analysis is to remove the effects of robot rotation from the images. Suppose we define a camera-centered coordinate system whose origin lies at the center of the camera lens. When a robot rotates about the axis that is perpendicular to the image plane and passes through the center of the camera lens, it makes points in the 3D scene and traces a circle in the 2D image. Rotations about the other two axes make points in the 3D scene that trace hyperbolic paths in the 2D image. Rotations about axes that do not go through the camera focal point are equivalent to a rotation about axes through the focal point plus an additional translation.

Once we remove rotations from the apparent image motion, the remaining image motion is due to translation. When the camera moves forward along a straight line, every point in the image seems to expand from a single point, and each image point's rate of expansion depends on the point's location and the distance between the robot and the point. This point, which is the intersection of the robot's velocity vector with the image plane, is called the focus of expansion. Locating this FOE is also a goal of general-motion processing.

To locate the FOE in an image, we compute the displacement of the projection of a set of 3D points in a pair of images (see Figure 1). The points used are usually distinct (for example, boundaries of high contrast and high curvature). Once we have found the displacement vectors, there are several methods of locating the FOE.[2] The problem is complicated by the fact that points on moving objects must be removed because their displacement motion will be inconsistent with the FOE of the stationary environment and the translating sensor.

Having located the FOE, we can use the points' rate of expansion to determine the relative distance between the camera and points in the scene. For example, as the sensor moves forward, nearby environmental points appear to move rapidly outward, whereas distant environmental points hardly move at all (see Figure 1b).

In addition to determining sensor motion, the vision system must detect and isolate moving objects from the stationary environment, track these objects over time and if possible, estimate their motion parameters and create general expectations about their future behavior. Since the camera itself is moving, we cannot assume that the stationary part of the environment will register in subsequent images. Simple frame-differencing techniques, which subtract successive frames pixel by pixel to detect and isolate moving objects, do not work in this case because all pixels typically change as a result of sensor motion. Spatiotemporal methods,[3] which treat a closely spaced temporal sequence as a volume (x, y, t), have not been demonstrated to be feasible for analyzing general motion or detecting and tracking moving objects in practical sequences. The techniques we describe here avoid these problems by determining sensor motion before trying to recover the 3D scene structure.

Once the moving objects have been found, it is easier to track them from frame to frame. Traditional techniques for object tracking use a multimode approach that synergistically combines several techniques such as centroid tracking, silhouette matching, correlation matching, feature matching, and Kalman filtering.[4] These techniques work in simple situations if the prediction is good; however, they do not use sensor motion or 3D scene structure, and are unlikely to be effective in scenes where there is no simple figure-ground relationship. (In some scenarios where the sensor is stationary, techniques using terrain information and based on frame differencing and Kalman filtering can be useful for tracking.) We expect that estimating and predicting 3D motion will significantly improve the tracking of moving objects in cluttered natural scenes. For example, if we can determine sensor motion, estimating the 3D motion of independently moving objects will let us predict their image projections over time far more accurately. Recently, researchers were able to predict 2D motion based on 3D location estimates.[5]

Qualitative dynamic-scene understanding

The DRIVE system (see Figure 2) emphasizes a qualitative line of reasoning and modeling, in which multiple scene interpretations are pursued simultaneously until ambiguities are resolved.

Fuzzy focus of expansion. Since it is difficult to pinpoint the FOE under arbitrary camera motion and noisy conditions, DRIVE extends the original FOE concept to the so-called fuzzy FOE, a connected image region marking the approximate direction of heading, rather than a singular image point.[2]

First, DRIVE processes sequences of

image pairs and gives unique labels to features; then it obtains the displacement vectors between corresponding feature points in successive images. DRIVE estimates the initial FOE location in the first pair of frames based on the camera's parameters and orientation with respect to the robot. It uses a method called "rotation-from-FOE" to determine the FOE for each frame pair and estimate possible rotations. In this approach, rotations are estimated based on the assumed FOE location. Experiments show that this "rotation-from-FOE" method is more robust against disturbances in the displacement field than other methods such as the traditional "FOE-from-rotation" approach,[1,2] which guesses which rotations will lead to determining the FOE location. Rotation-from-FOE assumes that the FOE for subsequent frame pairs has the same location as the FOE in the last pair of frames, which is a reasonable approximation in most cases. Then the algorithm derotates the second image based on the particular FOE; in other words, it maps the tracked points to the locations they would have if the robot had not rotated between frames. Derotation is guided by the goal of a perfectly radial pattern of displacement vectors. If the second image is perfectly derotated, all displacement vectors will lie on straight lines extending from the FOE location, reflecting the pure translation component of the camera motion.

However, if the predicted FOE location is incorrect, we do not get a perfect radial pattern. In this case, DRIVE computes the error between the guessed location and the true FOE location, updates the FOE location, and calculates the error of the new location. The system continues this modified steepest-descent search until it finds the FOE yielding the minimum error. Finally, the algorithm computes the region around the FOE within which the error is below some threshold. This "fuzzy FOE" embodies the assumption that noise and image distortion will prevent the system from determining the FOE location exactly. The fuzzy FOE prevents a false precision from causing later processing steps to fail, since it finds qualitative relationships rather than precise quantitative range values.

Once the system has located the image's FOE, it determines the robot's approximate velocity from the image information. Given the distance between the camera and the ground, DRIVE calculates the camera's angle of depression from the location of the FOE in the image. Knowing this distance and angle, DRIVE estimates by triangulation the distance between a point on the ground and the robot. From the change in that point's absolute position in consecutive frames, the system determines the robot's velocity.

Up to this point, dataflow is purely from the bottom up. However, DRIVE's reasoning process (described below) provides control information, such as the set of reference points that are believed to be stationary and might be used to compute the FOE.

Qualitative scene model. After computing the fuzzy FOE and derotated displacement vectors, DRIVE reasons about the 3D scene structure and independent object motion using the 2D location and motion of distinct image feature points relative to each other and to the fuzzy FOE. Given only an approximate FOE location, the displacement field's qualitative properties are the main source of reasoning.

This process incrementally builds a 3D camera-centered model of the environment in which the scene is described in qualitative terms, such as the relative distances of environmental features and how these features move in 3D space. The qualitative scene model (QSM) is declarative, describing the status and behavior of its elements and the relationships between them in coarse, qualitative terms. It does not try to derive a precise geometric description of the scene in terms of 3D structure and object motion. Features that are believed to be part of the static environment are labeled and used as references for computing the FOE.

Scene interpretations, the core of the QSM, are hypotheses about the relationships between the facts found in an image and their meaning in 3D space. The reasoning process that forms scene interpretations has access to 2D information in the form of already abstracted image observations. Each hypothesis represents a feasible and distinct interpretation of a scene. The QSM can contain multiple scene interpretations at the same time; individual interpretations are not kept as separate constructs inside the model, but generally share their components (partial interpretations). The interpretation process assembles

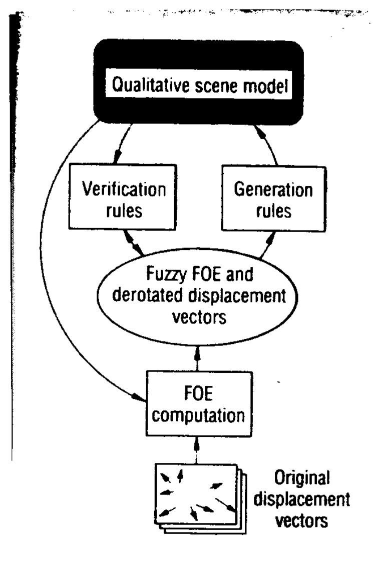


Figure 2. The DRIVE interpretation process.

complete interpretations from partial ones, ranks them, and makes results available to the other reasoning processes.

The QSM's basic elements are called entities, which are the 3D counterparts of the 2D features observed in the image. For example, the point feature A located in the image at x, y at time t — denoted by (Feature A t x y) — has its 3D counterpart in the model as (Member A). We express entity properties and relationships using assertions. For example, (Stationary 1) means that entity 1 is considered stationary in the corresponding scene interpretation. In any scene interpretation, the set of entities is divided into stationary (static) entities and mobile (possibly moving) entities. The QSM supplies DRIVE with a set of environmental entities that are believed to be stationary so that DRIVE can use them to compute the FOE.

The static scene structure is modeled in a way similar to a camera-centered depth map. At time t, the 3D location of any entity K with respect to the camera is completely specified by its image coordinates $x(K,t)$, $y(K,t)$ and its distance from the focal plane $z(K,t)$. However, in contrast to a regular depth map, the distance $z(K,t)$ is not represented by a numeric value, but by a qualitative spatial relationship between entities. In particular, the relation (Closer A B) means that entity A is believed to be closer to the camera than entity B. This relationship can be determined efficiently

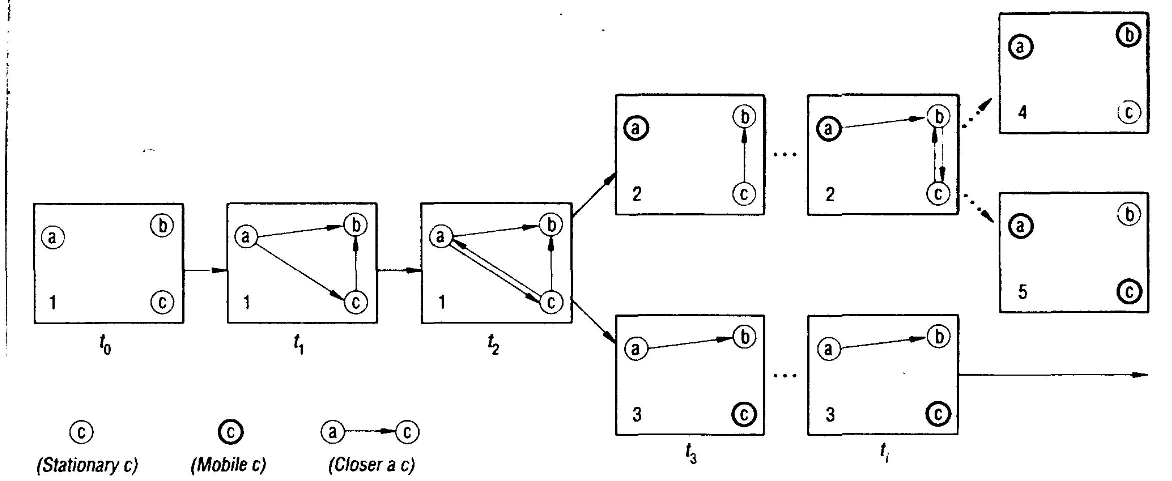


Figure 3. Developing a qualitative scene model over time.

and reliably from the divergence of displacement vectors. While a regular depth map must be updated after every frame, this semitopological map does not need to be modified as the camera moves through its environment. During this time, however, the model is continually refined as additional Closer relationships become evident.

Object motion is described at progressive levels of detail. The least that can be said about a moving entity *C* is (Mobile *C*), which simply means that this entity is not part of the static environment. Once an entity has been identified as being in motion, it is considered mobile in all subsequent frames, even when its 3D motion can no longer be verified. Relative motion between two entities in 3D may be detectable before the individual motion of a single entity becomes apparent. The fact (Movement-Between *C D t*) states that relative motion between *C* and *D* at time *t* has been deduced, but it tells nothing about which of the two entities are actually moving. This would be expressed by the more specific fact (Moves *C t*) or (Moves *D t*). Details about how an entity moves within the camera-centered coordinate frame are expressed by additional facts; for example, (Moves Left *C t*), (Moves Down *C t*), (Approaching *C t*), or (Receding *C t*).

The QSM network of hypotheses has the property of inheritance. Hypotheses common to all interpretations of a scene are found near the network root and are inherited by all interpretations below them. When the model receives a fact that is consistent with more than one hypothesis about the scene, the model branches. It develops all intermediate interpretations simultaneously as new information is received, eliminating any interpretation whose internal hypotheses conflict. Building the QSM thus involves four different activities: deriving 3D facts from the 2D image sequence, creating hypotheses about the scene, detecting conflicting hypotheses, and resolving those conflicts. To avoid a combinatorial explosion of possible scene interpretations, the search for the most plausible scene interpretation is guided by metarules:

- Always tend toward the most stationary (that is, the most conservative) solution. By default, all new entities (features entering the field of view) are considered stationary.
- Assume that an interpretation is feasible unless it can be proved false.
- If a new conclusion causes a conflict in a current interpretation, remove the conflicting interpretation.
- If current interpretations cannot accommodate a new conclusion, create a new, feasible interpretation and remove the conflicting ones.

The reasoning engine. The overall structure of the DRIVE interpretation process is shown in Figure 2. The QSM serves as the blackboard in a rule-based inference system and is maintained by a generate-and-test process. The interpretation network is accessible to two key sets of rules, each of which can modify it. The rules are based on perspective transformation and changing relationships between scene entities and the fuzzy FOE.[6]

There are two major groups of rules. Forward-chaining generation rules examine newly created derotated images and determine their consequences with respect to the model's current state. Then they place immediate conclusions (hypotheses) in the model. For example, if two image points *A* and *B* lie on opposite sides of the fuzzy FOE and are getting closer to each other, then one point must be in motion relative to the other. If an interpretation includes the hypothesis that one of these points is stationary, then the other point can be asserted to be mobile. The second group of rules, backward-chaining verification rules, check existing interpretations and try to prove them false. They are typically rules that would produce too many conclusions if they were used to generate hypotheses. A violation of a verification rule often indicates that an interpretation is implausible. For instance, one verification rule states that if an object is lower than another in the image, it is closer to the camera. Suppose point *A* is lower than point *B*, but an interpretation already includes the hypothesis that point *B* is closer to the camera than point *A*. If this verification rule fires, the interpretation will be marked as conflicting. Whether it is ultimately removed from the model will

Figure 4. An image sequence containing two moving cars.

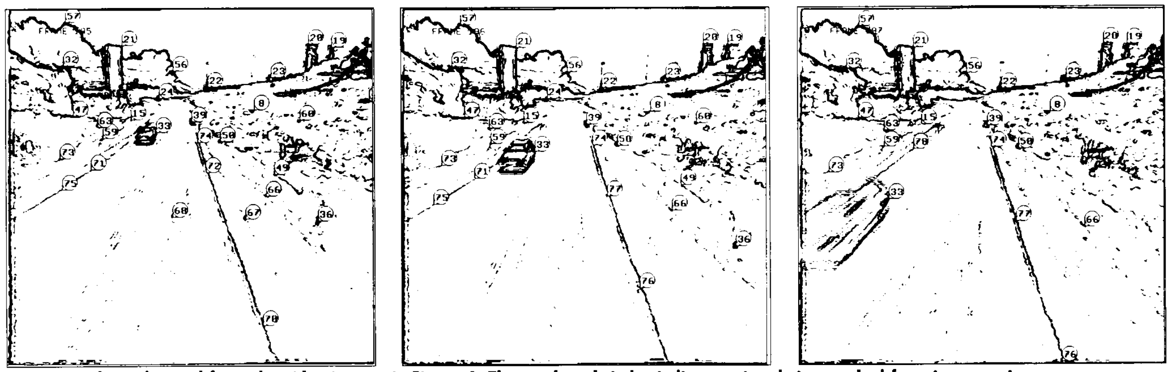


Figure 5. Edges obtained from the video images in Figure 4. The numbered circles indicate points being tracked from image to image.

depend on the global state of the QSM. Naturally, verification relies heavily on the backward-chained part of the reasoning process. Goal-driven, backward-chaining rules also deliver image information "on demand," that is, when the model needs information to complete a reasoning step at some level of the reasoning process.

Figure 3 shows how we develop a QSM by deducing information from the expansion rates of tracked image points. At time t_0, there are three features a, b, c in the model, all of which are assumed to be stationary. At time t_1, the subsystem has established three Closer relationships: Point A is closer than points B and C, and point C is closer than point B. At time t_2, a conflict arises in this interpretation. A new piece of evidence indicates that C is closer than A. As a result, the system creates two new interpretations, each of which considers one point as mobile (heavy circles). At time t_i, a new conflict arises in interpretation 2 from the additional fact that point B is closer than C. Since there is an active interpretation that can explain this conflict, interpretation 2 is collapsed. Only one valid interpretation remains, interpretation 3.

Detecting and tracking moving objects

In detecting moving objects, DRIVE accounts for the 3D structure of the observed environment along with the robot's motion. The system detects motion in two ways.[7] First, it can directly deduce some forms of motion from 2D displacement vectors without knowing anything about the underlying 3D structure. For example, if a forward-looking camera finds that a point is moving toward the fuzzy FOE rather than away from it, then that point must belong to a moving object. No other interpretation is possible.

Other kinds of motion are more subtle and require a second, interpretive detection method. Suppose the robot is approaching a "T" intersection. There is a building on the far side of the intersection's right branch, a van is approaching the intersection along the right branch, and DRIVE is tracking one point on the building and one on the truck.

As the robot moves forward, the point on the building will appear to move outward toward the edge of the image. However, if the truck is approaching the intersection at an appropriate speed, it will remain at the same position in the image. From the expansion pattern alone, the robot might conclude that the truck is a stationary object at infinite distance and so collide with it at the intersection.

The QSM, however, allows a more sophisticated kind of reasoning. Because the truck is occluding the view of the building, it must be closer to the robot than the building. Since the image point on the building is moving outward, the building cannot be at infinite distance. Therefore, the truck cannot be at infinite distance, and there must be a different interpretation of its expansion pattern. The most probable interpretation is that the truck is moving toward the robot.

We have tested this technique on a variety of images taken by the Autonomous Land Vehicle.[8] Figures 4 through 7 show a sample motion analysis and a computer-generated scene interpretation using a Symbolics 3670. Figures 4 and 5 show an image sequence containing two moving cars: one car has passed the robot and is barely visible in the distance, and the other is

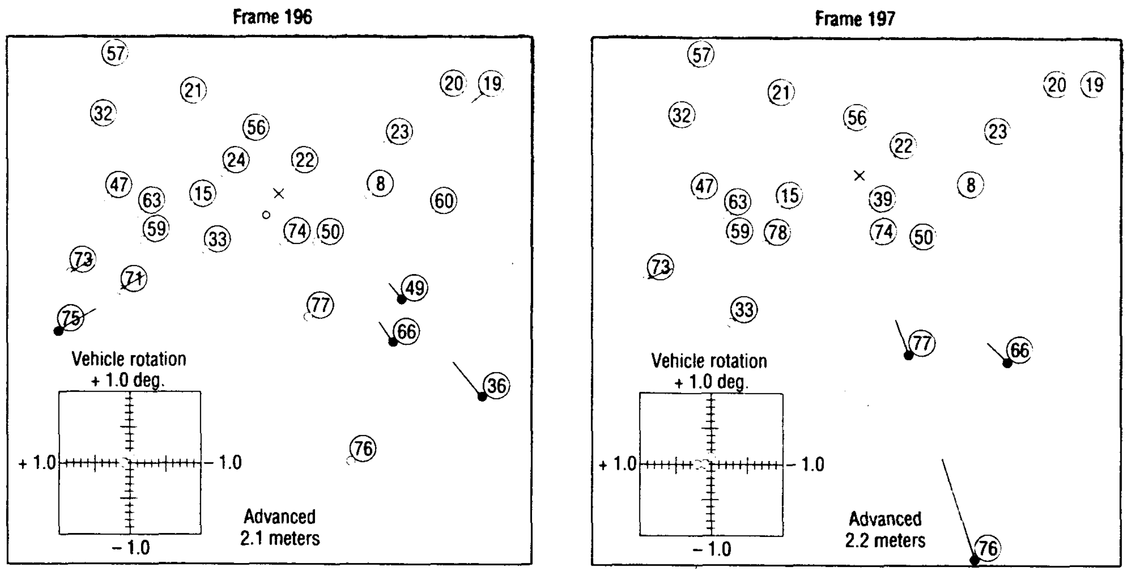


Figure 6. The displacement vectors, the resulting fuzzy FOE (shaded area), camera rotations about two axes, and estimated advancement.

approaching in the opposite direction and is about to pass. The numbered circles in Figure 5 represent points that are being tracked from image to image. Points 24 and 33 are on the moving cars.

Figure 6 shows the resulting fuzzy FOE (the shaded area). The circle inside the shaded area is the estimated FOE location with the lowest error value. Displacement vectors indicate the apparent motion of points from frame 195 to 196 (Figure 6a) and from frame 196 to frame 197 (Figure 6b). Dots mark the endpoints of selected vectors used to compute the vehicle's velocity and the distance it has advanced (estimated in meters). Its rotation about two axes (by amounts less than one degree) is plotted. Rotations about the third axis are small enough to be neglected.

Figure 7 shows the generated qualitative scene interpretations. Numbers at dots and in circles indicate stationary points, numbers in squares indicate potentially moving points, and numbers in diamonds indicate moving points. Lines between two points indicate that a point in a bigger circle is closer to the vehicle than one in a smaller circle or at a dot. A point near the bottom of the interpretation is near the bottom of the image. After frame 195, the QSM creates two interpretations (the upper boxes). Based on earlier conclusions, both interpretations show that entity 24 (indicated by a square) may be moving, but its direction of motion is undetermined. Interpretation 1 (frame 196), in which points 24 and 33 are labeled as mobile, is ranked higher than interpretation 2 (frame 196), in which point 33 is stationary, because interpretation 1 contains more stationary points. The QSM cannot rule out either interpretation, so both are carried over to the next frame pair (the lower boxes). If point 33 were stationary (interpretation 2, frame 197), its rapid expansion would indicate it must be at least as close to the vehicle as point 76. However, since point 76 is much lower in the image than point 33, and hence assumed closer to the vehicle, this contradicts the heuristic that entities lower in the image are generally closer in 3D space, which makes the entire interpretation implausible. As a result of this conflict, interpretation 2 (frame 197) is eliminated, leaving only the correct interpretation for frame 197 (interpretation 1), in which point 33 is definitely moving.

With reliable computation of displacement fields, this qualitative technique for understanding dynamic scenes reasons accurately on hundreds of image frames. The technique can also be extended to cases where the features are lines, regions, or contours. A forthcoming book will describe the qualitative approach in more detail.[9]

Theoretically, vision systems could use the information deduced from image motion to construct 3D models of local environments. This, of course, is one of the most fundamental goals of computer vision and would be of immense importance if autonomous mobile robots could build such models reliably. The entire static environment might be recovered up to the limits of image digitization and the accuracy of determining correct correspondences between successive frames. In fact, this goal still has not been achieved due to a variety of practical vision problems, such as the absence of robust algorithms and the need for high computational throughput, which we discuss in our accompanying paper.

Acknowledgments

This article describes the work of several of our colleagues at our respective research laboratories. We especially wish to thank Salit Gazit, Gerard Medioni, Keith Price, Wilhelm Burger, Barry Roberts, and Harpreet Sawhney for their help in preparing and reading this article. We also thank DARPA and Bob Simpson for their support, enthusiasm, and encouragement.

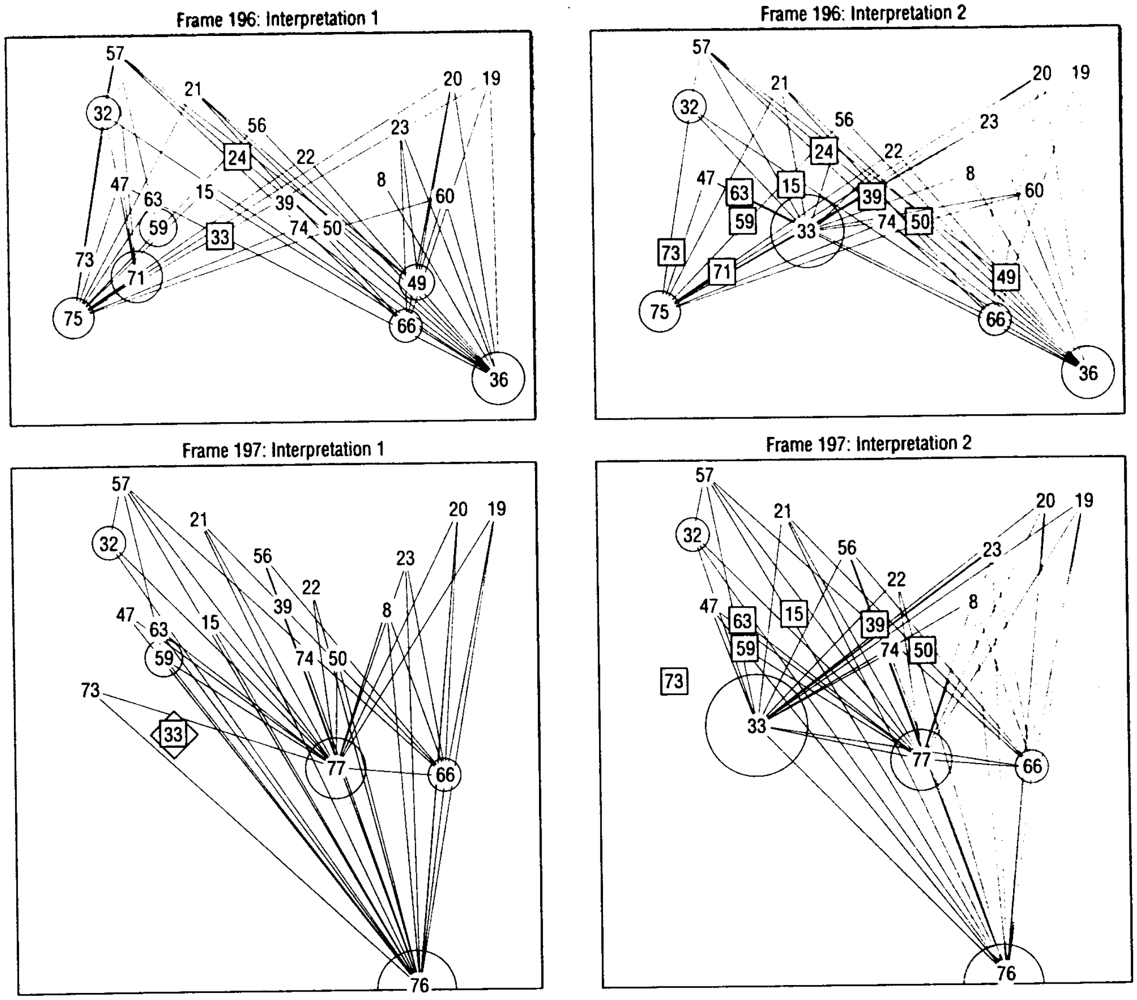


Figure 7. Scene interpretations based on the apparent motion of points in Figure 6.

References

1. B. Bhanu and W. Burger, "Qualitative Motion Detection and Tracking of Targets from a Mobile Platform," *Proc. DARPA Image-Understanding Workshop*, Morgan Kaufmann, San Mateo, Calif., 1988, pp. 289-318.

2. W. Burger and B. Bhanu, "Estimating 3D Egomotion from Perspective Image Sequences," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. PAMI-12, No. 11, Nov. 1990, pp. 1,040-1,058.

3. R.C. Bolles, H.H. Baker, and D.H. Marimont, "Epipolar-Plane Image Analysis: An Approach to Determining Structure from Motion," *Int'l J. Computer Vision*, Vol. 1, No. 1, June 1987, pp. 7-55.

4. B. Bhanu, "Automatic Target Recognition: State-of-the-Art Survey," *IEEE Trans. Aerospace and Electronic Systems*, Vol. AES-22, No. 4, July 1986, pp. 364-379.

5. H.S. Sawhney and A.R. Hanson, "Identification and 3D Description of 'Shallow' Environmental Structure in a Sequence of Images," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition*, CS Press, Los Alamitos, Calif., 1991, pp. 179-185.

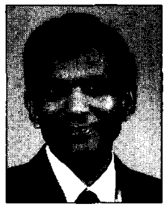
6. B. Bhanu and W. Burger, "Qualitative Approach for Dynamic Scene Understanding," *Computer Vision, Graphics, and Image Processing–Image Understanding*, Vol. 54, No. 2, Sept. 1991, pp. 184-205.

7. W. Burger and B. Bhanu, "Qualitative Understanding of Scene Dynamics for Autonomous Mobile Robots," *Int'l J. Robotics Research*, Vol. 9, No. 6, Dec. 1990, pp. 74-90.

8. B. Bhanu et al., "Qualitative Target Motion Detection and Tracking," *Proc. DARPA Image-Understanding Workshop*, Morgan Kaufmann, San Mateo, Calif., May 1989, pp. 370-398.

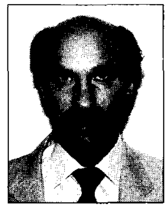
9. W. Burger and B. Bhanu, *Qualitative Motion Understanding*, Kluwer Academic Publishers, Norwell, Mass., to be published in Spring 1992.

Bir Bhanu is professor of electrical engineering and computer science and director of the Visualization Research Laboratory at the University of California at Riverside. He began researching qualitative motion understanding while a senior fellow at Honeywell Systems and Research Center. His research interests include machine learning, robot navigation, scientific visualization, target recognition and tracking, object modeling, multisensor integration, parallel algorithms, image-understanding algorithms and systems, and photo interpretation. Bhanu received his SM and his EE in electrical engineering and computer science from MIT, his PhD in electrical engineering from the University of Southern California Image Processing Institute, and an MBA from the University of California at Irvine. He also received an ME from the Birla Institute of Technology and Science and a BS from Banaras Hindu University. Bhanu is an associate editor of *Pattern Recognition* and *The Journal of Mathematical Imaging and Vision* and a member of the IEEE Computer Society, IEEE, ACM, AAAI, and the Pattern Recognition Society.



Ramakant Nevatia is professor of computer science and electrical engineering at the University of Southern California. He also directs the Institute for Robotics and Intelligent Systems there. His research interests include computer vision, artificial intelligence, and robotics. He earned a BS and MA from the University of Bombay and a MS and PhD from Stanford University, all in electrical engineering.

Nevatia is an associate editor of the journals *Pattern Recognition* and *Computer Vision, Graphics, and Image Processing*. He is the author of two books, *Machine Perception* and *Computer Analysis of 3D Curved Objects*, and a member of the IEEE Computer Society, IEEE, AAAI and ACM.



Edward M. Riseman is professor of computer and information science at the University of Massachusetts. He also directs the Computer Vision Laboratory, with projects in knowledge-based scene interpretation, motion analysis, mobile robot navigation, and parallel architectures for real-time vision. He and his colleagues are applying the analysis of static and dynamic images to such domains as natural outdoor scenes, biomedical images, industrial robotic environments, aerial images, and satellite images. His research interests have also included computer vision, artificial intelligence, learning, and pattern recognition.

Riseman received his BS from Clarkson College of Technology and his MS and PhD in electrical engineering from Cornell University. He is a member of the IEEE Computer Society, IEEE, ACM, AAAI, and the Pattern Recognition Society.

Readers can reach the authors in care of Bir Bhanu, College of Engineering, University of California at Riverside, Riverside, CA 92521, or by e-mail to bhanu@shivish.ucr.edu