Active-Vision-Based Multisensor Surveillance—An Implementation

Ardevan Bakhtari, Michael D. Naish, *Member, IEEE*, Maryam Eskandari, Elizabeth A. Croft, *Member, IEEE*, and Beno Benhabib

***Abstract*—In this paper, a novel reconfigurable surveillance system that incorporates multiple active-vision sensors is presented. The proposed system has been developed for visual-servoing and other similar applications, such as tracking and state estimation, which require accurate and reliable target surveillance data. In the specific implementation case discussed herein, the position and orientation of a single target are surveyed at predetermined time instants along its unknown trajectory. Dispatching is used to select an optimal subset of dynamic sensors, to be used in a data-fusion process, and maneuver them in response to the motion of the object. The goal is to provide information of increased quality for the task at hand, while ensuring adequate response to future object maneuvers. Our experimental system is composed of a static overhead camera to predict the object's gross motion and four mobile cameras to provide surveillance of a feature on the object (i.e., target). Object motion was simulated by placing it on an xy table and preprogramming a path that is unknown to the surveillance system. The selected cameras are independently and optimally positioned to estimate the target's pose (a circular marker in our case) at the desired time instant. The target data obtained from the cameras, together with their own position and bearing, are fed to a fusion algorithm, where the final assessment of the target's pose is determined. Experiments have shown that the use of dynamic sensors, together with a dispatching algorithm, tangibly improves the performance of a surveillance system.**

***Index Terms*—Active vision, dispatching, dynamic sensors, sensor fusion, sensing systems, surveillance.**

I. Introduction

TRADITIONALLY, in automated manufacturing environments, objects have been presented to industrial robots in exact positions and orientations (poses) using complex and expensive fixtures. However, recent advancements in automation technologies, combined with research in machine vision and robot control, should, in the near future, allow industrial robots to adapt to unexpected variations in their environments. Such autonomous systems will rely on real-time sensory feedback to reliably and accurately detect, recognize, and continuously track objects within the robot's workspace, particularly for applications such as *on-the-fly* object interception.

In the vast majority of earlier research in the area of vision-based surveillance, single or multiple static cameras have been utilized (e.g., [1]–[5]). Recent research on multisensor, reconfigurable dynamic systems has shown that significant improvements in the performance of a surveillance system can be realized through the use of mobile cameras (e.g., [6]–[9]). Thus, the focus of this paper is the implementation of such an active-vision-based surveillance system that utilizes multiple reconfigurable cameras for object detection and localization of a single object via sensor fusion.

The underlying methodology, introduced in Section II, is based on the principles of dispatching, typically used in the operation of service vehicles. An optimal subset of dynamic sensors are selected and maneuvered in response to the motion of the object. The data obtained is then combined in a data-fusion process. The modular hardware and software architecture used to implement the system in a flexible and distributed manner is outlined in Section III. The architecture serves to combine the newly developed algorithms for dispatching and motion control with established techniques for object motion prediction, image processing, and sensor fusion. Experimental results are presented in Section IV.

A. Active Vision

Active vision at the conceptual level has been defined as "... intelligent control strategies applied to the data-acquisition process which will depend on the current state of the data interpretation" [10]. In general, active-vision systems have addressed some, but rarely all, of the following issues:

1) control of the internal parameters (e.g., focus, zoom, and aperture) of a reconfigurable camera;
2) control of the external parameters (e.g., position and bearing) of a reconfigurable camera;
3) control of illumination conditions (i.e., intensity, direction, and colour of light sources);
4) determination of a sensing strategy based on the overall goal of the active-vision system.

For example, in [7] and [11], active vision was used only as a way to eliminate the workspace-size/pixel-resolution tradeoffs usually encountered with the use of single static-camera surveillance systems. The control strategy adjusts the external camera parameters (camera orientation) to monitor the entire workspace and acquire high-resolution images of a detected target. Similarly, the active-vision system described in [12] utilizes pan/tilt/yaw cameras to enhance the acquired image quality

Manuscript received November 6, 2003; revised August 24, 2004. This work was supported by the Natural Sciences and Engineering Research Council of Canada.

A. Bakhtari and M. Eskandari are with the Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, ON M5S 3G8, Canada (e-mail: bakhtar@mie.utoronto.ca; meskanda@mie.utoronto.ca).

M. D. Naish and B. Benhabib are with the Department of Mechanical and Materials Engineering, University of Western Ontario, London, ON N6A 5B9, Canada (e-mail: naish@eng.uwo.ca; beno@mie.utoronto.ca).

E. A. Croft is with the Department of Mechanical Engineering, University of British Columbia, Vancouver, BC V6T IZ4, Canada (e-mail: ecroft@mech.ubc.ca).

Digital Object Identifier 10.1109/TSMCC.2005.855525

1094-6977/$20.00 © 2006 IEEE

in order to improve object recognition and tracking for robot guidance and collision avoidance. The system proposed in [13] uses a control strategy to continuously configure both the internal (zoom) and external parameters (position and orientation) of a single camera for added accuracy and detection speed. The robot-mounted camera is positioned and optically configured so that the entire workspace can be monitored—once an object is detected, the camera is maneuvered and optically reconfigured so that high-resolution images of the target can be acquired.

B. Multicamera, Active-Vision Surveillance

The surveillance systems described above are atypical in that they rely solely on a single reconfigurable camera. Most surveillance systems utilize multiple reconfigurable cameras for increased accuracy and reliability [14]. These multicamera surveillance systems use at least one (or combination of more than one) way to combine the sensor data through sensor fusion. Sensors have been commonly classified by the type of fusion method used to process their output [15]: *Cooperative* sensors combine data from each sensor in order to obtain information that would not be available from any single sensor. *Complementary* sensors are those that do not directly depend on each other but whose data can be combined to provide more comprehensive information. *Competitive* sensors refer to those that independently provide information about the same physical phenomenon. Redundant data from competitive sensors is combined through sensor fusion to decrease the uncertainty [16].

The choice of the type of sensor network depends on application-specific requirements, including the sensing environment. For example, in object localization in uncluttered environments, for improved speed and reliability of target detection, a cooperative sensor network can be used to track targets through the use of both static and mobile cameras [17]. The system in [18], for example, uses multiple competitive pan/tilt cameras with remote zoom capabilities in order to increase the tracking accuracy of the system. System performance can be further increased by using a combination of cooperative and competitive sensors. The system proposed in [19], for example, like the system described in 0, uses a high frame rate static camera to roughly estimate the target's position; however, instead of one pair of binocular cameras, it uses six competitive pan/tilt cameras to determine the pose of the target via data fusion.

The multicamera active-vision systems described above use sensing strategies to orient the cameras in order to keep the target in the camera's field of view. However, suitable sensing strategies could be used to maneuver the cameras into optimal poses, not only to keep the target in the camera's field of view but also to maximize system performance [20], [21]. The system proposed in 0 uses high-level operation goals, geometric goals, and uncertainty-reduction goals to generate task plans for an assembly robot. The approach presented in [23] automatically controls the vision system through influence diagrams based on Bayesian tracking methods. The algorithm of [24] encodes the *a priori* workspace knowledge as a discrete probability density function of object locations and, then, generates camera poses using heuristic methods in order to maximize the probability of detecting the target while minimizing cost (time).

In contrast to the systems described above, our reconfigurable active-vision surveillance system comprises a combination of cooperative and competitive sensors, where the competitive cameras are used selectively in order to minimize the amount of data that must be processed. The required sensing strategy is determined through a heuristic optimization technique that dynamically selects a subset of cameras to service a demand point on the object's trajectory. The proposed methodology also determines optimal poses and dynamically adjusts the extrinsic camera parameters (its pose) in order to maximize localization accuracy [6].

C. Dispatching Methods

The principles of dispatching used for the operation of service vehicle [25] can be applied to the effective online reconfiguration of a sensing system—namely, selecting an optimal subset of dynamic sensors to be used in a data-fusion process and maneuvering them in response to the motion of the object. For example, the system proposed in [26] discretizes time and computes new viewing configurations for each time interval, while attempting to minimize changes in sensor position from one interval to the next. The system discussed in [27] determines two-dimensional (2-D) camera layouts using an off-line optimization—deviations from the planned approach are handled online using heuristics to adjust camera actions and temporal camera switching points. The system proposed in [28] seeks viewpoints that satisfy preset constraints over the entire task interval; if none exists, the interval is divided until satisfactory viewpoints are found.

As one can note, the above-mentioned systems (i.e., [25]–[27]) rely heavily on *a priori* knowledge and, therefore, are not sufficiently robust to variations in the target trajectory. The system proposed in [29], however, uses a set of cooperative active-vision sensors for continuous tracking and surveillance of multiple targets. Each camera is controlled individually through an agent-based architecture. In this system, each camera scans the workplace until a target is detected. Information regarding a detected target is shared among all sensors, upon which each sensor decides whether to participate in tracking of the target or to continue scanning for another target. In contrast to [29], the system proposed in this paper dynamically reconfigures the viewpoint (position and orientation) of each camera to increase the tracking accuracy. The dispatching algorithm can quickly and optimally adapt to unexpected trajectory variations without any prior knowledge about the object motion.

II. Sensor Dispatching

In the context of object surveillance, sensor dispatching attempts to maximize the effectiveness of a sensing system used to provide estimates of target parameters at predetermined times or positions along the object trajectory. Herein, it is assumed that the times at which the information is desired are fixed. These predetermined times are referred to as *demand instants* t_j. The position of the object at a particular demand instant is a *demand point* D_j. Without prior knowledge of the object trajectory, the pose of

a demand point corresponding to a demand instant may be predicted from observations of the object motion. This estimation of the demand-point pose changes (and its corresponding uncertainty diminishes) as the prediction accuracy improves over time; however, the demand instant remains constant.

If the sensing system comprises multiple redundant sensors, a subset of these may be sufficient to satisfy the sensing requirements of a demand point. Namely, a sensor-fusion process does not need to combine information from all of the sensors in the system. Instead, a subset of sensors ($k \leq n$, where k is the subset size and n is the total number of sensors) may be selected to survey the object at a particular demand instant. This group of sensors is referred to herein as a *fusion subset*.

In this context, the general sensor dispatching problem is stated as follows. Given a set of sensors and a set of time-varying demand points, determine the subset of sensors (and their corresponding poses) that will sense the object at each demand instant while ensuring that the remaining sensors are adequately distributed throughout the workspace for possible future use. There are a number of methods by which this sensor-dispatching problem may be approached. These include analytical optimization, machine learning, and heuristic approaches. One such heuristic sensor-dispatching approach is detailed in 0, a brief review of which follows.

To facilitate dispatching in real-time, a finite segment of the object trajectory (consisting of m demand instants) may be defined and the corresponding demand points estimated. This set of demands constitutes a *rolling horizon*. The period between two demand instants is the service interval—the amount of time available for planning and positioning of the sensors before data acquisition must occur. An appropriate service interval is usually set *a priori*, based on the task, the needs of the user, and the rate of data acquisition and processing.

A. Sensor Selection

Integral to dispatching is an estimate of the quality of data that each sensor can provide for the demand point at hand, given its pose in the workspace and its motion capabilities. This estimate is used to select sensors for inclusion in a fusion subset (i.e., *assignment*) and assess the desired pose of each sensor during surveillance. Herein, a *visibility measure* that is inversely proportional to the measurement uncertainty is used to assess the fitness of a single sensor or group of fused sensors. The visibility measure for a single sensor is considered to be

$$v_{\mathrm{s}} = \begin{cases} \frac{1}{\|R\|}, & \text{if the demand point is unoccluded} \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

where $\|R\|$ is the Euclidean norm of the covariance matrix associated with the sensor measurement. Note that, since this paper assumes an unobstructed single-object dynamic environment, assessment of occlusion is not necessary and, therefore, outside of the scope of this paper.

The visibility measure for a fusion subset comprising k sensors is defined as

$$v_f = \frac{1}{\|P\|} \tag{2}$$

where P represents the fused covariance matrix

$$P = \left[\sum_{i=1}^{k} R_i'^{-1} \right]^{-1}$$

and

$$R_i' = \begin{cases} R_i, & \text{if the demand point is unoccluded} \\ 0, & \text{otherwise.} \end{cases}$$

For the cameras that are used in our experimental setup, R is a function of six variance parameters: three for the Cartesian position of the target (x, y, z) and three for its orientation (n_x, n_y, n_z). Our variance analysis experiments led to the conclusion that only two controlled sensor parameters significantly affect the measurement variances: the Euclidean camera-to-target distance d and the bearing of the camera with respect to the target θ. Appendix A provides details of these experiments.

The visibility measure allows the merits of one sensor to be compared to others when observing the object at a particular demand point. It also allows the effect of different combinations of sensors (i.e., different fusion subsets) to be assessed. Sensors are selected based on their expected contribution to a high-quality estimate of the parameters of interest. While the best sensors are most often selected, the visibility measure also makes alternate strategies possible. For example, by assessing combinations of sensors over a number of demand points (predicted into the future) tradeoffs may be made, choosing nonoptimal sensors for the current demand to improve performance for other (future) demands. Additional information on the visibility measure may be found in [6] and [30].

B. Assignment and Positioning

The first demand point on the rolling horizon is serviced first. A *coordination strategy* optimally selects a subset of k sensors from the complete set of n sensors to service the demand instant. Furthermore, this search specifies the desired poses for all assigned sensors at the time of data acquisition, thereby implementing the *positioning strategy*.

Sensor assignment is triggered either by an object entering the workspace or the completion of a previous service interval. Once assigned, the subset of sensors cannot be altered until the demand is serviced (i.e., data are acquired), completing the service interval. However, the desired poses of the assigned sensors may be altered in real time. Large uncertainties in the predicted demand-point poses at the time of first assignment may necessitate reevaluation during the service interval. These pose adjustments are handled by a replanning method, discussed in Section II-D.

The assignment and positioning of sensors for a demand point is summarized as follows.

1) Predict the object's pose D_j at the demand instant t_j.
2) For every sensor $S_i, i = 1, \ldots, n$:
   a) Determine its best achievable pose with respect to D_j.
   b) Assess the corresponding (single sensor) visibility metric v_s.

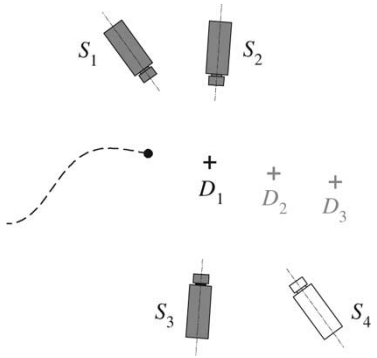

Fig. 1. Assignment and positioning of sensors S_1, S_2, and S_3 to demand point D_1. S_4 is unassigned.

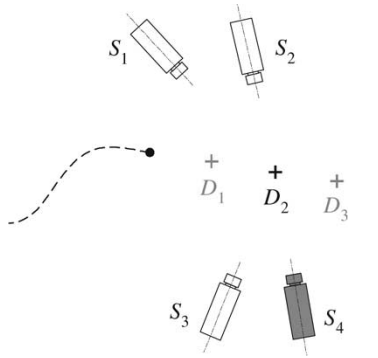

Fig. 2. Preassignment and prepositioning of sensor S_4 to demand point D_2.

3) Rank all sensors according to their achievable visibilities, from highest to lowest.
4) Assign the top k ranked sensors to t_j. [The desired pose of each assigned sensor is the best achievable pose determined in Step 2a).] See Fig. 1 for an example assignment.

Note that the best achievable pose for a demand point D_j is the one that maximizes the visibility of the object at the corresponding demand instant t_j. It is determined using a sensor model that describes the optimal sensing pose (with respect to the object) and may be constrained by the workspace and the dynamic limitations of the sensor. Thus, the best achievable pose specifies the goal of the sensor, given a finite amount of time for sensor positioning.

The size of the ideal fusion subset k is fixed throughout the surveillance task. Assignment means that no more than k sensors will be used for any demand; however, the optimality of the data from these sensors is not guaranteed. As long as at least one sensor can observe the demand point, the system will provide reliable information. Observations from additional sensors serve to increase the accuracy of the system.

C. Preassignment and Prepositioning

Sensors that have not been assigned to the first demand point on the rolling horizon may be preassigned to subsequent demands in anticipation of future sensing requirements. The approach used herein for preassignment and prepositioning is similar to that outlined in Section II-B for the servicing of the first sensing point; however, there are some differences. First, while the suitability of each sensor is considered for each demand instant, only those that have not been previously (pre)assigned may be preassigned to the demand instant under consideration. Second, the preassignment algorithm is an iterative process that considers additional demand instants until either all sensors have been preassigned or the end of the rolling horizon has been reached. See Fig. 2 for an exemplary preassignment.

D. Replanning

As each sensor is assigned to a particular (predicted) demand point, its desired pose with respect to the demand point is also specified. This desired pose is used as an input to the sensor-motion controller. As the service interval elapses, the estimates of the demand-point pose are continually updated as additional observations become available. Note that the estimate of each demand point has an associated region of uncertainty (e.g., a confidence interval). If the newly predicted demand point lies outside of the uncertainty region of the demand point at the time of assignment, then replanning is initiated. The demand-point pose is replaced with the new prediction, and the desired poses of all sensors assigned to this demand point are adjusted to reflect the changes in prediction. In contrast, if the newly predicted demand point lies within the uncertainty region of the original demand point, then no replanning is done since adjustment of the sensor poses provides no guaranteed benefit.

The uncertainty region for each sensor is specified by its sensor model. Replanning is initiated if $\delta_x > \rho\sqrt{\sigma_x^2} \vee \delta_y > \rho\sqrt{\sigma_y^2} \vee \delta_z > \rho\sqrt{\sigma_z^2}$, where δ_x, δ_y, and δ_z are the position errors between the demand-point estimate at the time of assignment and the current demand-point estimate; ρ is a scaling factor, and σ^2 is the variance of the initial estimate.

E. Initial-Pose Determination

The quality of information obtained during the surveillance of the object normally would be dependent on the initial poses of the sensors. The impact of initial sensor placement is more pronounced if the speeds of the sensors are relatively slower than the speed of the object. Lower sensor speeds would require the sensors to be more widely distributed in the workspace. If a rough estimate of the object trajectory is known *a priori*, the sensing system can be reconfigured in an optimal manner. One such initial sensing-system configuration approach that maximizes the visibility of the surveillance system over the entire target trajectory is outlined in [30].

III. Multicamera Surveillance System Implementation

The multicamera, active-vision surveillance system described in this paper can track a target (i.e., a feature on a moving object) as it maneuvers through the workspace and estimate its position (x, y, z) and orientation (n_x, n_y, n_z) at predefined time instants t_j. In our experimental system, shown in Fig. 3, an object with a circular marker is placed on a two-dimensional xy table so as to simulate the planar motion of an object traveling on a conveyor or on an automated guided vehicle (AGV). The setup

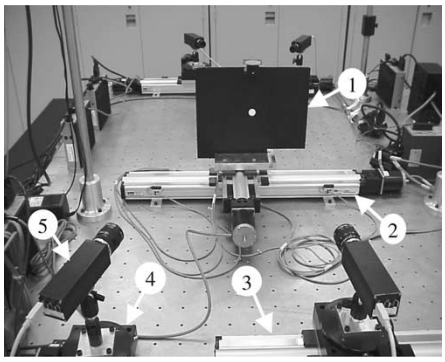

Fig. 3. Surveillance system. See Table I for hardware specifications.

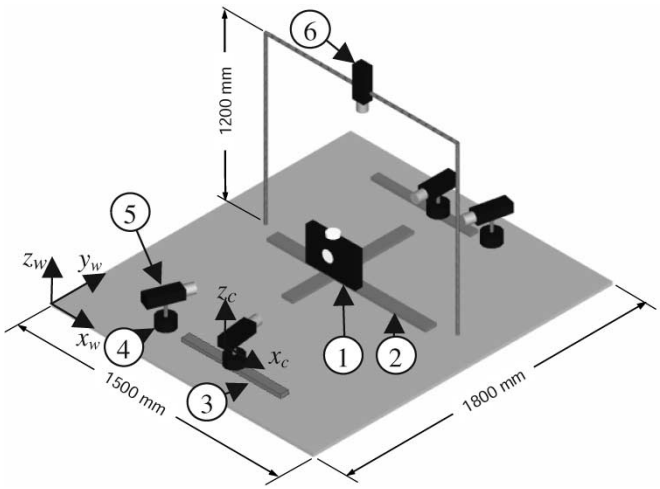

Fig. 4. System layout. See Table I for hardware specifications.

permits experimentation with different target trajectories as well as it facilitates accurate evaluation of system performance by recording the exact pose of the target at each demand instant.

A. Surveillance System Configuration: Hardware and Software

The surveillance system uses information from a static overhead camera to select and position a subset of cameras, from the four active-vision cameras available, for independent target-pose estimation. The active-vision cameras have varying motion capabilities—all of them can rotate about their z axis, while two can also translate along their x axis.[1] These cameras use a 0.25-in CMOS sensor and are equipped with 25-mm lenses. This results in the target circle being about 200–250 pixels in diameter when imaged. Figs. 3 and 4 show the overall layout of the surveillance system; Table I provides the hardware specifications for all of the major system components.

The surveillance system utilizes three main software modules developed in our laboratory: the Prediction, Dispatching, and Imaging Modules, which work cooperatively to predict the target's motion, reconfigure the active-vision system, and image the target at predefined demand instants, respectively (Fig. 5).

[1]Camera x and z axes are in the same directions as the world coordinate frame shown in Fig. 4.

TABLE I
HARDWARE SPECIFICATIONS

Part #	Hardware	Characteristic
1	Target	Matte black aluminium plate marked with white circular marker (Diameter = 25 mm).
2	x-y Table	Range: 500 mm (x)/200 mm(y) Accuracy: 48 μm(x)/24 μm(y) Max Velocity: 0.3 m/sec
3	Two Linear Stages	Range: 300 mm Accuracy: 30 μm Max Velocity: 0.3 m/sec
4	Four Rotary Stages	Accuracy: 10 arc sec Max Velocity: π/6 rad/sec
5	Four Dynamic CMOS Cameras	Resolution: 640×480 pixels Lens Focal Length: 25 mm
6	One Static CCD Camera	Resolution: 640×480 pixels Lens Focal Length: 12mm

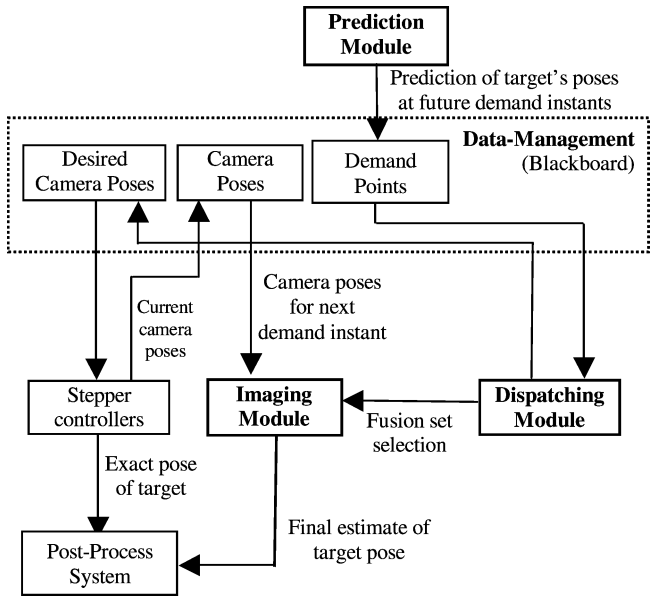

Fig. 5. Software architecture.

Due to the high modularity level of the surveillance system, a common memory space is used to ensure that multiple modules can simultaneously access the current state of common state variables. The common memory space is implemented as a "blackboard" that provides a controlled method for the exchange of data between the individual modules. A Data-Management Module controls the flow of information into and out of the blackboard, preventing inadvertent corruption and, thereby, ensuring data integrity and reliability.

B. Prediction Module

The primary purpose of the Prediction Module is to determine an approximate estimate of the target's (i.e., side marker's) future position by tracking the relative position of a different circular marker mounted on the top surface of the object using a static overhead camera (Fig. 4). Since this marker is restricted to a planar motion, its vertical distance (along the z axis) to the overhead camera remains constant, and the image coordinates are easily transformed to world coordinates after calibration

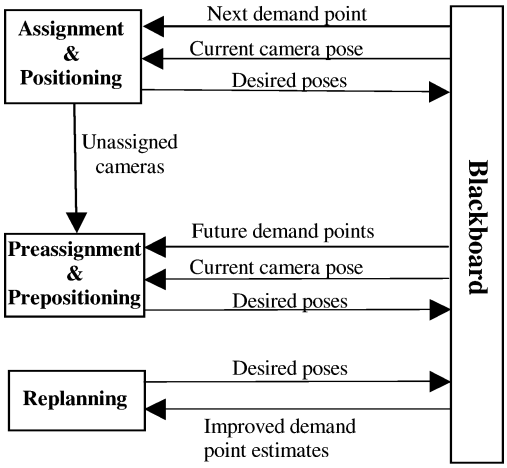

Fig. 6. Dispatching submodules.

of the camera parameters using Tsai's camera-calibration technique [31].

The observed position of the top marker is fed into a recursive Kalman filter (KF) [32] for target-motion prediction. Namely, the module uses the target state model maintained by the KF to predict the future poses of the target corresponding to (service) demand instants as specified by the Dispatching Module. The KF was chosen as a prediction and smoothing operator since it is relatively simple to implement and provides accurate results in tracking a wide range of trajectories.

C. Dispatching Module

The Dispatching Module has two primary functions: 1) selecting a subset of dynamic cameras for target imaging, from the set of all available cameras; and 2) positioning all cameras with respect to the predicted target poses at specific demand instants (demand points), as estimated by the Prediction Module. The Dispatching Module directs the Imaging Module to perform multicamera image acquisition of the target at each demand instant. The heuristic dispatching approach detailed in Section II is implemented using an online dispatching architecture.

The Dispatching Module consists of four submodules, shown in Fig. 6: 1) an Assignment and Positioning Submodule; 2) a Preassignment and Prepositioning submodule; 3) a Replanning Submodule; and 4) a Motion-Control Submodule.

1) Assignment and Positioning Submodule: This submodule uses an iterative-ranking approach to select a subset of cameras to be assigned to the next demand instant. An "increasingly complex" ranking procedure is used. The cameras are first ranked based solely on *a priori* information regarding the demand point. This is followed by an assessment of the normalized distance of each sensor to the demand point (i.e., actual distance/[max velocity $\times$ time]). Sensors are ranked by their achievable proximity to the demand point—the closest sensor ranks highest. Finally, the sensors are ranked according to the estimated visibility of the demand point from their best achievable pose. The best achievable pose is determined through a heuristic optimization of the sensor uncertainty as given by the sensor model, as discussed in Section II. The top-ranked cameras are assigned to the next demand instant with a specified desired pose (based on the best achievable pose), while unassigned cameras are assigned to future demand instants using the preassignment and prepositioning module.

2) Preassignment and Prepositioning Submodule: This submodule uses a similar method to the Assignment and Positioning Module—the algorithm considers as many future demand instants as needed until all available sensors are preassigned or the end of the rolling horizon has been reached. The desired pose of each assigned (or preassigned) sensor is set to its best achievable pose with respect to the demand point that it is assigned to.

3) Replanning Submodule: This submodule continually monitors the sensor assignments as the predictions of the demand-instant locations improve. If the newly predicted pose of the target differs from the original target-pose prediction used at the time of camera assignment by more than the uncertainty associated with the estimate, the best achievable camera poses are reassessed.

4) Motion Submodule: The main objective of this submodule is to maneuver the cameras into their desired poses as specified by the above three submodules. It serves as an interface between our in-house-developed software modules and the commercial motion controllers, which control the stepper motor drives of the linear and rotary stages on which the dynamic cameras are mounted. As such, this submodule determines the time-varying trajectories of the cameras online. The Motion Submodule also independently controls the motion of the user-selected object trajectory and records it for later system-performance assessment.

This submodule also polls the motion controllers for real-time camera-pose information to be utilized by both the Dispatching and Imaging Modules. The Dispatching Module uses the current camera poses in order to assess camera assignments and determine their future poses. The Imaging Module uses the current camera-pose information to transform the target pose, obtained in camera coordinates, to world coordinates. This transformation serves to align the data from individual camera frames into a common reference frame as required for fusion.

D. Imaging Module

Upon receiving a command from the Dispatching Module, the Imaging Module uses the assigned dynamic cameras to acquire and process images of the target for independent localization.

1) Marker-Pose Determination: An analytical technique developed by Safaee-Rad *et al.* [33] is used to estimate the pose of the target's circular marker in camera coordinates based on the perceived ellipse, defined by the marker's boundary edges. The five elliptical parameters—namely, the center point (x and y), the major axis, the minor axis, and the orientation—are required by the algorithm. The elliptical parameters are found through a least squares type of fitting technique, applied to the edge-detected image. Model fitting is accomplished by minimizing

the following error function J:

$$J = \sum_{i=1}^{N} [w_i Q(X_i, Y_i)]^2 \tag{3}$$

where w_i is the weighting factors that account for the nonuniformity of the data points along the boundary of the ellipse and $Q(X_i, Y_i)$ is the general equation of an ellipse given by

$$Q(X_i, Y_i) = aX_i^2 + bX_iY_i + cY_i^2 + dX_i + eY_i + f \tag{4}$$

where (X_i, Y_i) is the position of the ith edge pixel marking the boundary of the perceived ellipse. The optimized vector (a, b, c, d, e, f) is then used to calculate the five parameters of the ellipse. From these elliptical parameters, the three-dimensional (3-D) pose of the circular marker is computed [33]. The orientation of the marker is determined first. Then, using the intrinsic camera parameters and the previously known marker size (in mm), the marker position is calculated (see Appendix B for details).

2) Data Fusion: Individual estimates of the target's pose are first transformed from their camera coordinate frames into a common world coordinate frame using the known camera poses. Since the center of the camera frame and its rotation axis may not coincide, there may exist an offset that must be accounted for when transforming the target position from the camera frame to the common world coordinate frame. This offset is determined through a moving camera-calibration method outlined in Appendix C.

The aligned estimates are then fused to determine a single estimate of the target's pose in world coordinates. The specific fusion algorithm, *Optimal Region* [15], was chosen herein for several reasons: 1) it requires a minimum amount of *a priori* knowledge about the target's trajectory, providing robustness to unexpected trajectory variations; 2) uncertainties in all cameras are estimated and considered to allow optimal fusion; 3) in the event of sensor malfunction, invalid data may be identified and discarded, affording some degree of fault tolerance; and 4) it is computationally efficient, enabling an online implementation.

A model developed in [34], which represents each sensor reading as a range containing the correct value of the variable, serves as the basis of the Optimal Region fusion algorithm. In order to distinguish between the model and the physical sensor, two terms were defined in [34]: *concrete sensor* and *abstract sensor*. A concrete sensor is the physical sensor with a single value reading X. An abstract sensor is a range of values r, which includes the correct value of the physical variable being measured by the sensor

$$r \in \Re, \quad X - \delta < r < X + \delta \tag{5}$$

where δ represents the accuracy of the sensor, calculated through *a priori* knowledge of the sensor's uncertainty.

The Optimal Region fusion algorithm requires an abstract sensor reading from each of the physical sensors (i.e., given k sensor readings, the algorithm will acquire k ranges, $r_1 \ldots r_k$). If a no-fault system is assumed (i.e., all abstract sensors return a range that includes the correct value of the physical variable), then the range of values common to all abstract sensors contains the correct value of the physical variable. This common range r_c is referred to as the optimal region

$$r_c = r_1 \cap r_2 \cap \cdots \cap r_k. \tag{6}$$

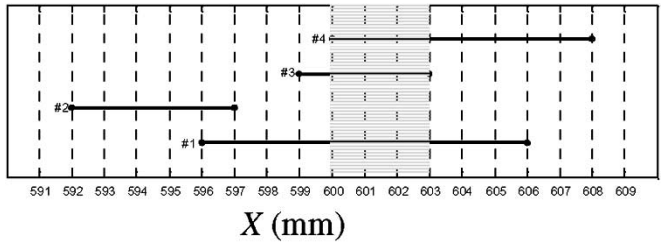

Fig. 7. Optimal region of four abstract sensor ranges.

In real systems, however, faults may occur and, therefore, one or more sensors may not return a range that includes the correct value of the physical variable. The optimal region algorithm would still return a range that includes the correct value if

$$k_j < k/2 \tag{7}$$

where k_f is the total number of faulty sensors and k is the total number of sensors used (see [34] for proof and further details). To accomplish this, the algorithm finds regions where $(k - k_f)$ of the k abstract sensors intersect, referred to as *probable regions*. The algorithm uses range trees recursively to return the smallest region that contains all probable regions as the optimal region. The final fused estimate L is the weighted average of the center of each probable region q_j

$$L = \frac{\sum_j q_j s_j}{\sum_j s_j} \tag{8}$$

where s_j is the number of sensors intersecting in the jth probable region.

Let us consider the following example. Four sensors are used to estimate the linear position of a target X. Sensor 1 returns an X-value of 601 mm, where through off-line sensor modeling it is known with 99.74% confidence that the true target position is within ± 5 mm of this reading. Therefore, the abstract sensor range is defined to be 596 to 606 mm, as shown in Fig. 7. Abstract sensor ranges are found for the other three concrete sensors in a similar manner. The maximum number of allowable faulty sensors for fusion with $k = 4$ sensors is one. The optimal region is defined as a range of values for which $k - k_f = 3$ abstract sensors agree. Therefore, in this example, the optimal region is $600\,\mathrm{mm} < X < 603\,\mathrm{mm}$, where the ranges for Sensors 1, 3, and 4 intersect. Sensor 2 is not included in the probable region and, therefore, is considered to be faulty. The final fused estimate is the center of the single probable region, $L = 601.5$ mm.

IV. Implementation Experiment: Robustness to Trajectory Variation

A large number of experiments conducted in our laboratory verified that the performance of a surveillance system can be tangibly improved by using active sensors in conjunction with a dispatching algorithm. The observed improvements are primarily

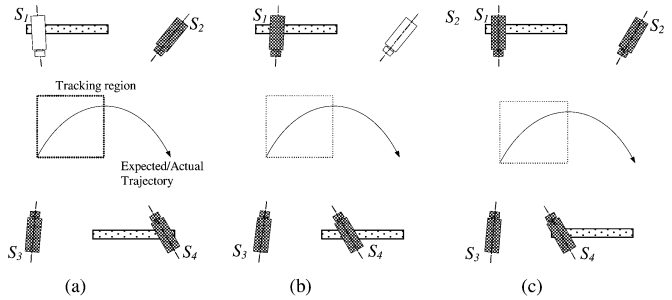

Fig. 8. Experiment Set A. Optimal initial sensing-system configurations for (a) a static system, (b) a medium-speed system, and (c) a fast-speed system.

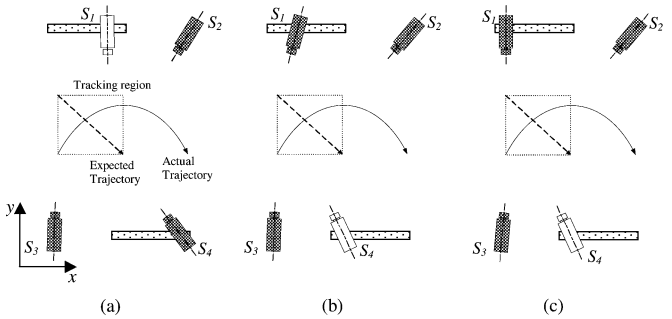

Fig. 9. Experiment Set B. Optimal initial sensing-system configurations for (a) a static system, (b) a medium-speed system, and (c) a fast-speed system.

due to: 1) increased robustness of the system (i.e., improved ability to cope with *a priori* unknown object trajectories); 2) decreased uncertainty associated with target-pose estimation, achieved through better sensor positioning and the use of redundant sensor data; and 3) increased reliability gained through sensory fault tolerance.

A. Experimental Setup

In the specific examples discussed herein, the performances of three surveillance-system configurations are considered. Each configuration is distinguished by the motion capabilities of its sensors, namely: 1) static; 2) medium speed; and 3) fast speed. The static system has no camera-motion capabilities and, therefore, the cameras are restricted to their initial poses, optimally set at the start of the experiments. In the medium-speed system, the cameras have both translational and rotational motion capabilities, with maximum velocities of $x = 10$ mm/s and $\alpha = 0.1$ rad/s. In the fast-speed system, the cameras also have both translational and rotational capabilities; however, the maximum velocities are set to $x = 25$ mm/s and $\alpha = 0.2$ rad/s. The goal of each system is to track a moving object with a trajectory that is *a priori* unknown and at each demand instant to use the best three out of four cameras to acquire target data. Prior to each experiment, the cameras are placed at optimal initial poses, specific to the motion characterisitics of the sensors and the expected object trajectory.

For the first set of experiments (Experiment Set A), the cameras of each surveillance system were placed at an optimal initial configuration based on the assumption that the object will follow a parabolic trajectory (Fig. 8). For these experiments, this trajectory assumption held true. For the second set of experiments (Experiment Set B) the cameras were initially configured based on the assumption that the object would enter from the top-left corner of the workspace and follow a straight-line trajectory to the opposite diagonal corner (Fig. 9). For these experiments, this trajectory assumption did not hold true. The object actually entered the workspace from the bottom-left corner and followed a parabolic trajectory.

The maximum speed of the target along its trajectory was set to 20 mm/s for all of the experiments. Consequently, one notes that the cameras of the medium-speed system had a lower translational speed than the target along the x axis for most of the target's trajectory, while the cameras of the fast-speed system had a translational speed along the x axis that exceeded the target's speed throughout its entire trajectory. This superiority is reflected in the optimal initial configuration of the system in both sets of experiments (Figs. 8 and 9). The initial configuration for the medium-speed system has the two cameras with translational capabilities (i.e., Cameras 1 and 4) placed ahead of the target, to compensate for their slower translational velocity. This strategy forced the systems to utilize the rotational capabilities of each camera to a greater extent than the faster system, in an effort to keep the target within the field of view of the cameras. The initial configuration of the cameras for the static system was selected such that the target can be imaged by at least three cameras at any demand point along the object trajectory, albeit at a lower quality than may be achieved with the other two mobile sensing systems. It should be noted that the size of the tracking region was limited by the viewing area of the static system.

B. Performance Evaluation

System evaluation was carried out using the visibility metric introduced in Section II. The visibility of the target for a particular camera is calculated using the expected variance of the measurements. This variance is a function of the camera's Euclidean distance to the target and its bearing (i.e., the angle between the camera's optical axis and the normal of the target's surface). In addition, the system performance was also evaluated by determining the extent of the errors in the real-time target-pose estimation. The absolute error in position estimation E_{position} is the Euclidean distance between the true target position (x_t, y_t, z_t) as measured by the motion controller and the estimate of the target's position (x_e, y_e, z_e) as determined by the surveillance system

$$E_{\text{position}} = \sqrt{(x_e - x_t)^2 + (y_e - y_t)^2 + (z_e - z_t)^2}. \quad (9)$$

Similarly, the absolute error in surface normal estimation $E_{\text{orientation}}$ is the angle between the true surface normal $\boldsymbol{n}_t$ and the estimated surface normal $\boldsymbol{n}_e$

$$E_{\text{orientation}} = \cos^{-1}(\boldsymbol{n}_t \cdot \boldsymbol{n}_e). \quad (10)$$

C. Results

The pose of the moving target was estimated at ten distinct demand instants during its motion. The demand-instant

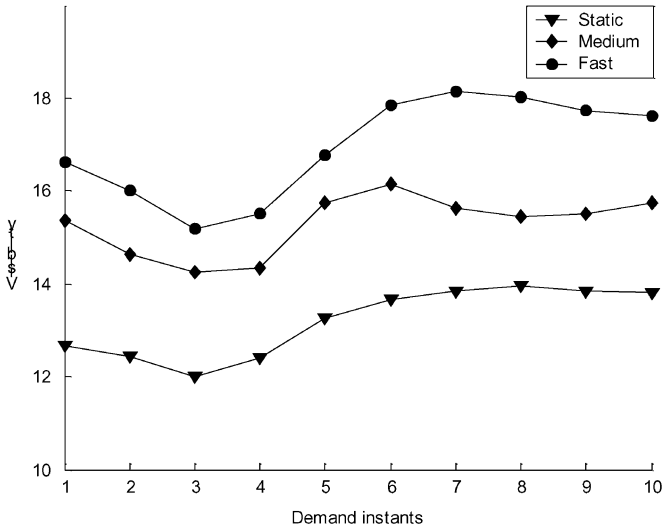

Fig. 10. Experiment Set A. Observed visibility of the target by a sensing-system configuration that is expecting and observing a parabolic trajectory.

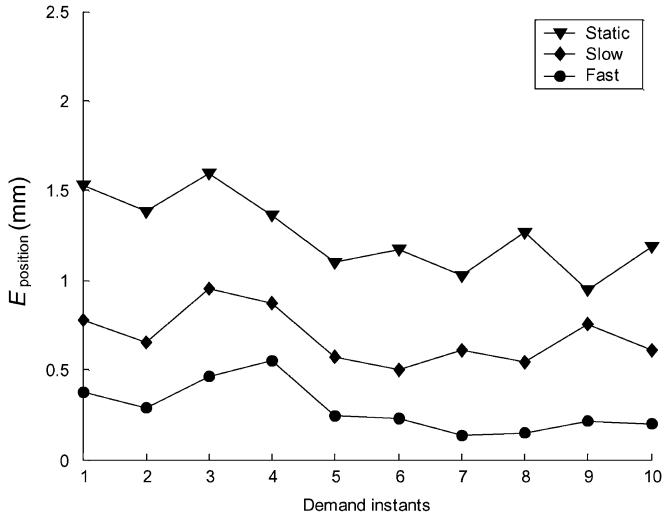

Fig. 12. Experiment Set A. Absolute errors in the target's position estimates by a sensing-system configuration that is expecting and observing a parabolic trajectory.

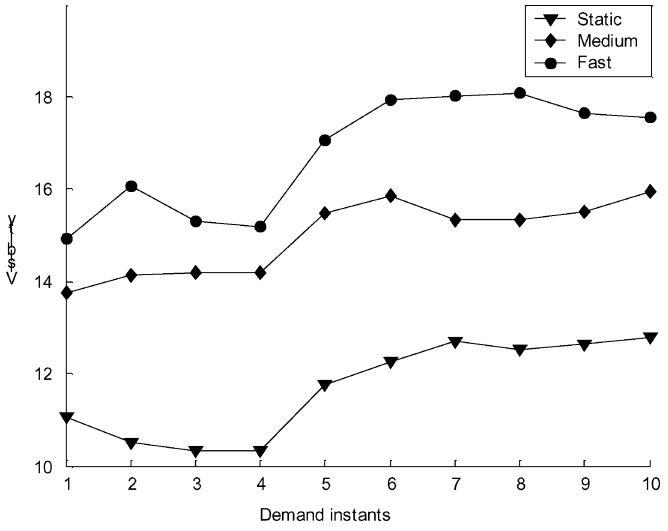

Fig. 11. Experiment Set B. Observed visibility of the target for a sensing-system configuration that is expecting a straight-line trajectory but actually observing a parabolic trajectory.

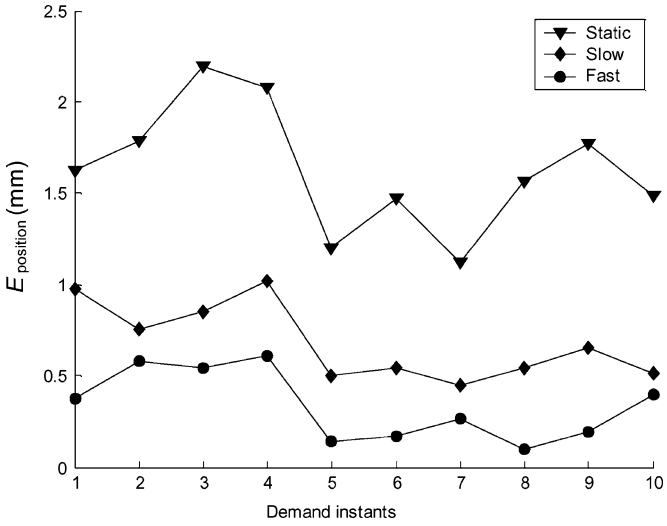

Fig. 13. Experiment Set B. Absolute errors in the target's position estimates by a sensing-system configuration that is expecting a straight-line trajectory but actually observing a parabolic trajectory.

visibilities corresponding to sensing systems initially configured for different object trajectories, Experiment Sets A and B, are given in Figs. 10 and 11, respectively. When comparing the three systems in both experiments, it can be noted that the use of mobile cameras tangibly increases the target's visibility (and, hence, reduces estimation uncertainty). As the dynamic capabilities of the cameras improve, the performance of the system is enhanced.

Comparing the visibilities of the three distinct systems in Experiment Set A with those of Experiment Set B, several observations are made. First, as expected, the static system of Experiment Set A has a noticeably higher visibility than the static system of Experiment Set B, since the former system is better initially configured for observing the parabolic object trajectory. Second, in Experiment Set B, although both systems (medium and fast speed) suffer in performance during the initial demand instants due to a poor initial system configuration, they both recover within at most three demand instants (the fast-speed system recovers faster). Thus, the experiments confirm that the dispatching algorithm can deal with object trajectories that deviate significantly from the expected trajectory, whereas the static system suffers significant performance degradation. The absolute errors in the estimation of the target pose for both experiments are presented in Figs. 12–15.

V. Conclusion

The implementation of a novel multicamera surveillance system, suitable for target localization in online applications, has been presented in this paper. The surveillance system uses

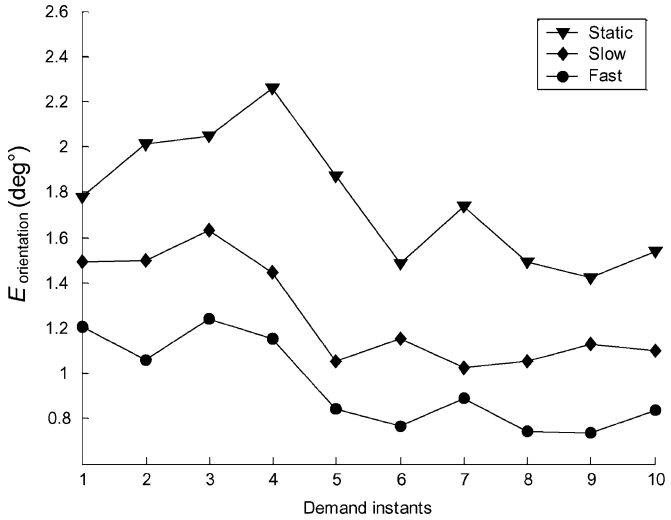

Fig. 14. Experiment Set A. Absolute errors in the target's surface-normal estimates by a sensing-system configuration that is expecting and observing a parabolic trajectory.

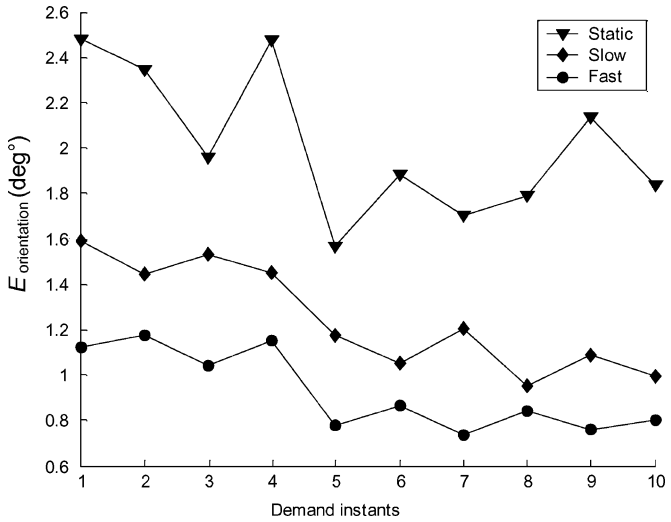

Fig. 15. Experiment Set B. Absolute errors in the target's surface-normal estimates by a sensing-system configuration that is expecting a straight-line trajectory but actually observing a parabolic trajectory.

both active-vision and multisensor fusion in order to achieve higher object-localization accuracies than are typically possible through the utilization of single and/or static sensors. Dispatching is utilized to optimally select and position groups of cameras to determine the target's pose at the next demand instant. Independent target-pose estimates, as determined by each camera, are fused to provide a final estimate of a target's pose.

Extensive experiments, some of which are presented herein, have shown that tangible improvements in accuracy and reliability can be realized through the use of multiple, active-vision cameras controlled by the proposed dispatching algorithm. Additional experiments have shown that the effectiveness of the proposed surveillance system for object localization is greatly affected by the number of cameras used, their motion capabilities, and their initial configurations.

APPENDIX A
SENSOR MODELING

Sensor modeling is an important part of optimal dispatching. The objective is to estimate a sensor's performance given a set of environmental conditions. As mentioned in Section II-A, the performance of each camera, assuming that the view of the target is unoccluded, is described in this work by a visibility performance metric

$$v_s = \frac{1}{\|R\|} \tag{A1}$$

where $\|R\|$ is the Euclidean norm of the covariance matrix associated with the parameter measurements. For the cameras used in our experiments, there are six variance measurements, three for the target position (x, y, z) and three for orientation (n_x, n_y, n_z). Through variation analysis it was determined that only two controlled parameters significantly affect the measurement variances: the Euclidean camera-to-target distance d and the camera's bearing θ.

Two-factorial experiments were performed to determine the relationship between each measurement variance and the two controlled parameters, d and θ. As an example, the results for two variances (estimation along the y axis and one surface normal) are shown in Fig. 16. The improvement in variance is mainly due to the shape of the elliptical projection of the circular marker (the target is better viewed at an angle between 20° and 40° from the surface normal), and noise in images has lower noise-to-signal ratios at close distances (i.e., when the marker occupies a larger area in the acquired images). Other parameters, such as illumination, also affect the measurement variance, but since they are uncontrollable they are not included in the visibility measure.

APPENDIX B
THREE-DIMENSIONAL LOCATION ESTIMATION

The process of 3-D location estimation [33] of a circular feature, using the estimated general parameters of the ellipse, is as follows.

1) Estimation of the coefficients of the general equation of the cone

$$ax^2 + by^2 + cz^2 + 2fyz + 2gzx + 2hxy + 2ux + 2vy + 3wz + d = 0. \tag{B1}$$

2) Reduction of the equation of the cone

$$\lambda_1 X^2 + \lambda_2 Y^2 + \lambda_3 Z^2 = 0 \tag{B2}$$

where the XYZ-frame is the canonical frame of the conicoids.

3) Estimation of the coefficients of the equation of the circular-feature plane (plane intersecting the cone that would result in a perfect circle)

$$lX + mY + nZ = p \tag{B3}$$

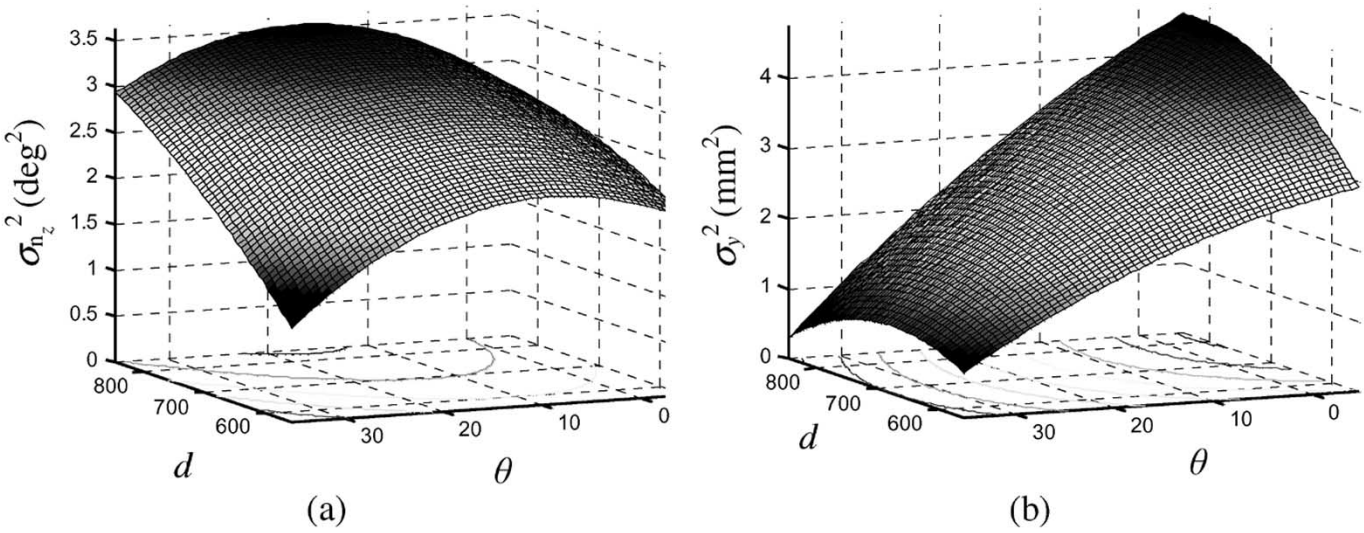

Fig. 16. Response surfaces of variances in (a) orientation estimation about z axis, $\sigma_{n_z}^2$ and (b) y axis position, σ_y^2.

using the following transformation:

$$\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{-m}{\sqrt{l^2+m^2}} & \frac{-nl}{\sqrt{l^2+m^2}} & l & 0 \\ \frac{-l}{\sqrt{l^2+m^2}} & \frac{-mn}{\sqrt{l^2+m^2}} & m & 0 \\ 0 & \sqrt{l^2+m^2} & n & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X' \\ Y' \\ Z' \\ 1' \end{bmatrix}. \quad \text{(B4)}$$

The transformation is defined such that Z' is normal to the plane axis defined by (B3). The solution to determining $l, m,$ and n depends on which of three possible cases occurs: a) $\lambda_1 < \lambda_2$; b) $\lambda_1 > \lambda_2$; or c) $\lambda_1 = \lambda_2$.

4) Estimation of the direction cosines of the surface normal with respect to the camera frame: l, m, n.
5) Determination of the radius of the circular feature (r) and its center $(X_o'Y_o'Z_o')$ in camera coordinates $(X'Y'Z')$ by solving the following system of equations:

$$X_o' = -\frac{B}{A}Z_o'$$
$$Y_o' = -\frac{C}{A}Z_o'$$
$$Z_o' = \pm\frac{Ar}{B^2 + C^2 - AD} \quad \text{(B5)}$$

where

$$A \equiv \left(\lambda_1 l_1^2 + \lambda_2 l_2^2 + \lambda_3 l_3^2\right)$$
$$B \equiv \left(\lambda_1 l_1 n_1 + \lambda_2 l_2 n_2 + \lambda_3 l_3 n_3\right)$$
$$C \equiv \left(\lambda_1 m_1 n_1 + \lambda_2 m_2 n_2 + \lambda_3 m_3 n_3\right)$$
$$D \equiv \left(\lambda_1 n_1^2 + \lambda_2 n_2^2 + \lambda_3 n_3^2\right). \quad \text{(B6)}$$

APPENDIX C
MOVING-CAMERA CALIBRATION

In order to determine the relationship between the camera coordinates and the world coordinates, the extrinsic camera parameters must be determined. The extrinsic parameters were found in our work through the application of Tsai's calibration method [31]. In order to account for changes in extrinsic camera parameters, the algorithm first applies a transformation matrix to return the rotated camera frame to the frame identical to its original reference frame before rotation and then applies a second transformation matrix to account for any translational movement of the camera since the initial calibration.

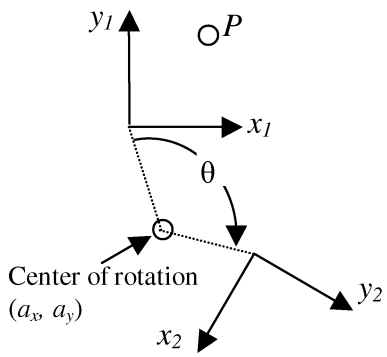

Fig. 17. Camera coordinate frame transformation due to camera rotation.

In order to counter the effects of rotation, one must know the rotation axis, the rotation angle, and the center of rotation. We first make the assumption that the rotation axis is in the z direction. A secondary calibration method is used to determine the center of rotation in camera coordinates. First, the target is placed in the camera's field of view, and readings are taken on its position in camera coordinates. Multiple readings are taken and the results fused to minimize the effect of random noise. Without changing the target position, the camera is then rotated as much as possible while still keeping the target in its field of view. A second set of readings is taken on the target's position in the new camera coordinates. The following equations are then solved for the required offsets:

$$\begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \sin(\theta) \end{bmatrix} \begin{bmatrix} P_{x_1} - a_x \\ P_{y_1} - a_y \end{bmatrix} + \begin{bmatrix} a_x \\ a_y \end{bmatrix} = \begin{bmatrix} P_{x_2} \\ P_{y_2} \end{bmatrix} \quad \text{(C1)}$$

where (P_{x_1}, P_{y_1}) and (P_{x_2}, P_{y_2}) are the target locations in camera coordinates before and after rotation, respectively; θ is the rotation angle; and (a_x, a_y) is the center of rotation in camera coordinates (Fig. 17). The inverse of (C1) transforms any camera-frame rotation to the original camera-frame location.

ACKNOWLEDGMENT

The authors would like to thank S. Zhu for his technical assistance.

REFERENCES

[1] M. Beynon, D. Van Hook, M. Seibert, and A. Peacock, "Detecting abandoned packages in a multi-camera video surveillance system," in *Proc. IEEE Int. Conf. Advanced Video and Signal Based Surveillance*, Miami, FL, 2003, pp. 221–228.

[2] V. Kettnaker and R. Zabih, "Bayesian multi-camera surveillance," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Fort Collins, TX, 1999, pp. 253–259.

[3] T. Ellis, "Multi-camera video surveillance," in *Proc. IEEE Conf. Security Technology*, Atlantic City, NJ, 2002, pp. 228–233.

[4] D. Hogg, "Model-based vision: A program to see a walking person," *J. Image Vis. Comput.*, vol. 1, no. 1, pp. 5–20, 1983.

[5] B. Rao and H. Durannt-Whyte, "A decentralized Bayesian algorithm for identification of tracked targets," *IEEE Trans. Syst., Man, Cybern.*, vol. 23, no. 6, pp. 1683–1698, Nov. 1993.

[6] M. D. Naish, E. A. Croft, and B. Benhabib, "Coordinated dispatching of proximity sensors for the surveillance of manoeuvring targets," *J. Robot. Comput. Integr. Manufact.*, vol. 19, no. 3, pp. 283–299, Jun. 2003.

[7] E. J. Gonzalez-Galvan, F. Pazos-Flores, S. Skarr, and A. Cardebas-Galindo, "Camera pan/tilt to eliminate the workspace-size/pixel-resolution tradeoffs with camera-space manipulation," *J. Robot. Comput. Integr. Manufact.*, vol. 18, no. 2, pp. 95–104, 2002.

[8] K. Kurihara, S. Hoshino, K. Yamane, and Y. Nakamura, "Optical motion capture system with pan-tilt camera tracking and real-time data processing," in *Proc. IEEE Int. Conf. Robotics and Automation*, Washington, DC, 2002, pp. 1241–1248.

[9] R. Collins, O. Amidi, and T. Kanade, "An active camera system for acquiring multi-view video," in *Proc. IEEE Int. Conf. Image Processing*, Rochester, NY, 2002, pp. 517–520.

[10] R. Bajcsy, "Active perception vs. passive perception," in *Proc. 3rd IEEE Workshop on Computer Vision: Representation and Control*, Detroit, MI, 1985, pp. 55–59.

[11] L. Marcenaro, F. Oberti, and C. Regazzoni, "Change detection methods for automatic scene analysis using mobile surveillance cameras," in *Proc. IEEE Conf. Image Processing*, Vancouver, BC, Canada, 2000, pp. 244–247.

[12] J. A. Strickrott and S. Negahdaripour, "On the development of an active vision system for 3-D scene reconstruction and analysis from underwater images," in *Proc. IEEE Oceans Conf.*, Halifax, NS, Canada, 1997, pp. 626–633.

[13] K. Hosoda, H. Moriyama, and M. Asada, "Visual servoing utilizing zoom mechanism," in *Proc. IEEE Int. Conf. Robotics and Automation*, Nagoya, Japan, 1995, pp. 178–183.

[14] M. Dietl, J. Gutmann, and B. Nebel, "Cooperative sensing in dynamic environments," in *Proc. IEEE Int. Conf. Intelligent Robots and Systems*, Maui, HI, 2001, pp. 1706–1713.

[15] R. R. Brooks and S. S. Iyengar, *Multi-Sensor Fusion: Fundamentals and Applications with Software*. Englewood Cliffs, NJ: Prentice-Hall, 1998.

[16] R. C. Luo and M. G. Kay, "Multisensor integration and fusion for intelligent machines and systems," in *Data Fusion in Robotics and Machine Intelligence*, M. A. Abidi and R. C. Gonzalez, Eds. San Diego, CA: Academic, 1992, pp. 7–136.

[17] P. Piexoto, J. Batista, and H. Araujo, "Integration of information from several vision systems for a common task of surveillance," *J. Robot. Autonomous Syst.*, vol. 31, no. 1, pp. 99–108, Apr. 2000.

[18] Y. Kim, K. Lee, C. Lim, and K. Park, "A study of the implementation of moving object tracking," in *Proc. SPIE Conf. Visual Communications and Image Processing*, Taipei, Taiwan, 1995, pp. 1183–1193.

[19] D. Woo and D. Capson, "3D visual tracking using a network of low-cost pan/tilt cameras," in *Proc. Can. Conf. Electrical and Computer Engineering*, Halifax, NS, Canada, 2000, pp. 884–889.

[20] C. Cowan and P. Kovesi, "Automatic sensor placement from vision task requirements," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 10, no. 3, pp. 407–416, Mar. 1988.

[21] K. Tarabanis, R. Tsai, and P. Allen, "Analytical characterization of the feature detectability constraints of resolution, focus, and field of view for vision planning," *Comput. Vis. Graph. Image Process., Image Understand.*, vol. 59, pp. 340–358, May 1994.

[22] S. A. Hutchinson and A. C. Kak, "Spar: A planner that satisfies operational and geometric goals in uncertain environments," *AI Mag.*, vol. 11, no. 1, pp. 20–61, 1990.

[23] T. Levit, J. Agosta, and T. Binford, "Bayesian methods for interpretation and control in multi-agent vision systems," in *Proc. SPIE Conf. Applications of AI X: Machine Vision and Robotics*, Orlando, FL, 1992, pp. 536–548.

[24] Y. Ye and K. Tsotsos, "Sensor planning for 3D object search," *Comput. Vis. Image Understand.*, vol. 73, no. 2, pp. 145–168, Feb. 1999.

[25] H. Psaraftis, "Dynamic vehicle routing problems," in *Vehicle Routing: Methods and Studies*, B. L. Golden and A. A. Assad, Eds. Amsterdam, The Netherlands: North-Holland, 1988, pp. 223–248.

[26] S. Sakane, T. Sato, and M. Kakikura, "Model-based planning of visual sensors using a hand-eye action simulator: HEAVEN," in *Proc. Int. Conf. Advanced Robotics*, Versailles, France, 1987, pp. 163–174.

[27] T. Matsuyama, T. Wada, and S. Tokai, "Active image capturing and dynamic scene visualization by cooperative distributed vision," in *Advanced Multimedia Content Processing*. Lecture Notes in Computer Science, vol. 1554, Berlin, Germany: Springer-Verlag, 1999, pp. 252–288.

[28] S. Abrams, P. Allen, and K. Tarabanis, "Computing camera viewpoints in an active robot workcell," *Int. J. Robot. Res.*, vol. 18, no. 3, pp. 267–285, Mar. 1999.

[29] T. Matsuyama and N. Ukita, "Real-time multi-target tracking by a cooperative distributed vision system," *Proc. IEEE*, vol. 90, no. 7, pp. 1136–1150, Jul. 2002.

[30] M. D. Naish, E. A. Croft, and B. Benhabib, "Simulation-based sensing-system configuration for dynamic dispatching," in *Proc. IEEE Int. Conf. Systems, Man, and Cybernetics*, vol. 5, Tucson, AZ, 2001, pp. 2964–2969.

[31] R. Tsai, "A versatile camera calibration technique for high accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses," *IEEE J. Robot. Autom.*, vol. 3, no. 4, pp. 323–344, Aug. 1987.

[32] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Trans. ASME, J. Basic Eng.*, vol. 82, no. D, pp. 35–45, 1961.

[33] R. Safaee-Rad, I. Tchoukanov, K. C. Smith, and B. Benhabib, "Three-dimensional location estimation of circular features for machine vision," *IEEE Trans. Robot. Autom.*, vol. 8, no. 5, pp. 624–640, Oct. 1992.

[34] K. Marzullo, "Tolerating failures of continuous-valued sensors," *ACM Trans. Comput. Syst.*, vol. 8, no. 4, pp. 284–304, Nov. 1990.

Ardevan Bakhtari received the B.A.Sc. degree in mechanical engineering, mechatronics option, from the University of Toronto, Toronto, ON, Canada, in 2002. He is currently pursuing the Ph.D. degree in sensor planning for surveillance of dynamic environments at the University of Toronto.

His current research interests are in the area of mechatronics, including active-vision, multisensor surveillance, and agent-based control.

Michael D. Naish (S'96–M'03) received the B.Sc. degree in computer science and the B.E.Sc. degree in mechanical engineering from the University of Western Ontario, London, ON, Canada, in 1996, the M.A.Sc. degree in mechanical engineering from the University of British Columbia, Vancouver, BC, Canada, in 1999, and the Ph.D. degree in mechanical and industrial engineering from the University of Toronto, Toronto, ON, Canada, in 2004.

Since 2003, he has been an Assistant Professor in the Department of Mechanical and Materials Engineering, University of Western Ontario. His research interests include active-vision systems, sensor planning, multisensor systems, and mechatronics.

Maryam Eskandari received the B.A.Sc. degree in mechanical engineering from the University of Tabriz, Tabriz, Iran, in 1993 and the M.A.Sc. degree in mechanical engineering from the University of Toronto, Toronto, ON, Canada, in 2002.

Her research focused on design and implementation of multicamera sensing systems for the surveillance of dynamic targets. She is currently working as a Design Engineer in the Department of Mechanical and Industrial Engineering, University of Toronto.

Elizabeth A. Croft (M'95) received the B.A.Sc. degree in mechanical engineering from the University of British Columbia, Vancouver, BC, Canada, in 1988, the M.A.Sc. degree from the University of Waterloo, Waterloo, ON, Canada, in 1992, and the Ph.D. degree from the University of Toronto, Toronto, ON, Canada, in 1995.

She is currently an Associate Professor in mechanical engineering at the University of British Columbia. Her research interests include human–robot interaction, sensor and device fusion, and mechatronics.

Beno Benhabib is a Professor in the Departments of Electrical and Computer Engineering and Mechanical and Industrial Engineering at the University of Toronto, Toronto, ON, Canada. His main research interests are in the area of robotics and automation, primarily for manufacturing. He is the author of the book *Manufacturing: Design, Production, Automation and Integration* (Marcel Dekker, 2003) and more than 240 articles in international conferences and journals. In the past two decades, he has supervised the research work of over 90 postdoctoral fellows and graduate students.