Adaptive Panoramic Stereo Vision for Human Tracking and Localization with Cooperative Robots[*]

Zhigang Zhu

Department of Computer Science, City College of New York, New York, NY 10031

zhu@cs.ccny.cuny.edu

Deepak R. Karuppiah, Edward M. Riseman and Allen R. Hanson

Department of Computer Science, University of Massachusetts, Amherst, MA 01003

{deepak|riseman|hanson}@cs.umass.edu

Introduction

An adaptive panoramic stereo vision approach for localizing 3D moving objects has been developed in the Department of Computer Science at the University of Massachusetts at Amherst. This research focuses on cooperative robots involving cameras (residing on different mobile platforms) that can be dynamically composed into a virtual stereo vision system with flexible baseline in order to detect, track, and localize moving human subjects in an unknown indoor environment. This work was carried out under the support of the DARPA ITO Software for Distributed Robotics (SDR) and Autonomous Mobile Robot Software (MARS). This research project is relevant to both civil and military applications with distributed stationary and mobile perceptual agents. Applications are widespread, such as video surveillance and monitoring, security, crowd control, intelligent rooms and office, etc. For example, we are currently looking into applications in metropolitan scenarios such as New York City for solving problems of finding and protecting people in emergency circumstances, for example, during a terrorist attack or a fire in an office building.

[*] This work was done when the first author was with the Computer Science Department at the University of Massachusetts at Amherst. This research was supported by DARPA/ITO programs Mobile Autonomous Robot S/W (MARS) (Contract Number DOD DABT63-99-1-0004) and Software for Distributed Robotics (SDR) (Contract Number DOD DABT63-99-1-0022), and partially by the Army Research Office under grant number DAAD19-99-1-0016.

The idea of distributing sensors to cooperate across different robots stems from the requirements of potentially limited (stereoscopic sensor) resources in a large robot team. However, the advantages of cooperative vision derive from far more than overcoming this resource constraint. Any fixed-baseline stereo vision system has limited depth resolution because of the physical constraints imposed by the separation of cameras, whereas a system that combines multiple views allows a planning module to take advantage of the current context and goals in selecting a subset of better sensor viewpoints. In this article, we focus on cooperative behavior involving cameras that are aware of each other, residing on different mobile platforms, to compose dynamically a virtual stereo sensor with a flexible baseline. In the adaptive stereo model, the sensor geometry can be controlled to manage the precision of the resulting virtual stereo system. This cooperative stereo vision strategy, can be used for any types of cameras, but is particularly effective with a pair of mobile panoramic cameras with a full 360-degree field of view (FOV), and therefore have the potential of almost always seeing each other. Once dynamically calibrated by "looking" at each other, they jointly can view the environment to estimate the 3D structure of visible surfaces and objects in the scene by triangulation, the well-known stereo geometry.

Two new lines of research are implicit under the adaptive panoramic stereo vision approach. First a new way is explored to use distributed sensors themselves as calibration targets – the so called "mutual calibration". Since the distributed mobile robots know each other in both geometry and appearance, each one of them can serve as a calibration target for sensors on other robots. This eliminates the need to use known 3D objects in the scene for camera calibration, and therefore is very useful for 3D vision tasks in unknown environments. Second, this research studies the problem of view planning for a stereo vision system with adaptive viewpoints and baselines, which provide a new ways to improve the measurable range and robustness of stereo vision.

We note that there is a fairly large body of work on detection and tracking of humans, motivated largely by the DARPA VSAM effort [1]. On the other hand, different kinds of omnidirectional (or panoramic) imaging sensors have been designed [2,3,4,5]. Omnidirectional vision has become quite popular with many approaches for robot navigation [4,6], 3D reconstruction [7] and video surveillance [5, 8,10]. The work most related to ours is the real-time human tracking system by Sogo et al [5] using multiple omnidirectional cameras distributed in an indoor environment. Our work differs from theirs in three aspects. First, we explore the potential of using only two panoramic cameras to detect and localize multiple moving humans in a full 360-degree view, even under the degradation of triangulation. Second, we are dealing with panoramic stereo vision on mobile platforms – thus issues of dynamic calibration and view planning should be studied. A novel concept of mutual calibration is proposed, and a systematic error analysis of panoramic stereo with adaptive baselines and viewpoints provides rules for dynamic view planning for multiple panoramic cameras. Third we have developed a more accurate and robust human detection and extraction algorithm. Based on this functionality, the most distinctive feature of our approach is the ability to compose cooperative sensing strategies across the distributed panoramic sensors of a robot team to synthesize optimal panoramic stereo vision for human detection and tracking.

In addition to applying the adaptive panoramic stereo vision approach to two physically moving platforms, this technique could also be very useful in a sensor network with many distributed and stationary sensors for video surveillance or human augmentation. In video surveillance, it is very important to decide which sensor subsets could form the best views for optimal 3D localization of a subject. In a human augmentation application with a sensor network of stationary panoramic cameras, a human with a wearable camera system could have extended perceptual capabilities if he is a node inside the network. This would allow the human to both receive augmented 3D perceptual information, but also play a role as an element of the virtual stereo pair. In such a network with mobile sensors, adaptive panoramic stereo vision technology can determine the location of the human by reconfiguring a stereo vision system from a set of stationary cameras in the environment. Then the mobile wearable sensor can form a virtual stereo system with any of the monitoring cameras to determine the 3D location of a third subject of interest, sometimes at greatly increased accuracy beyond perceptual ability of the human.

Panoramic Imaging Geometry

An panoramic imaging sensor appropriate for our goal of detecting and localizing human subjects should have a vertical viewing angle that spans the horizon, enabling a robot to see both the heads of moving humans and the ground plane across a reasonable range of distance. Different kinds of omni-directional (or panoramic) imaging sensors have been designed [2,3,4,5]. In these experiments we utilized the panoramic annular lens (PAL) camera system [3], with a field of view (FOV) of 360-degrees horizontally the desired viewing geometry around the horizon, -15 to +20 degrees vertically (Fig. 1).

Using the imaging geometry for the PAL lens, we have developed the mathematical model [9] for calibrating this panoramic camera. This turns out to be a rather difficult problem, involving the solution of nonlinear equations of 13 independent parameters. Alternatively we can use an empirical method to transform the panoramic image, which consists of the following three simple steps:

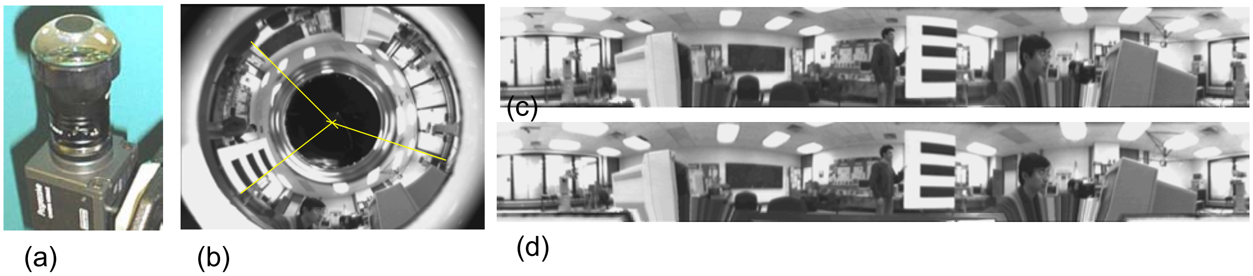

Fig. 1. Panoramic camera and images. (a) PAL lens and camera; (b) an orignal panoramic image (768*576); (c) cylindrical panoramic image, with radial distortion; and (d) cylindrical panoramic image, after eliminating radial distortion.

Step 1. Center determination - First, we adjust the camera to point vertically upward so that projections of vertical lines in the world remain straight in the panoramic image and they intersect at a single point in the center of the image (Fig. 1b). If more than two such lines are detected in an original panoramic image, the center point can be determined by their intersection.

Step 2. Vertical distortion rectification - Distortion exists in the vertical direction of the unwarped cylindrical image (or the radial direction in the original panoramic image) due to the non-linear reflection and refraction of the 2nd-order mirror surfaces of the panoramic lens. Note the unequal widths of the black-white bars on the whiteboard in Fig. 1c caused by the vertical distortion (the widths are equal in the real world). We use a 2nd-order polynomial to approximate the distortion along the vertical direction. Fig. 1d shows the rectified result using the 2nd-order polynomial approximation where only 3 point pairs are needed in the original and the rectified cylindrical images. In practice we use the least mean square method with more than 3 pairs of points.

Step 3. Calibration of the virtual cylindrical imaging sensor - After image unwarping and distortion rectification, we have a cylindrical image generated by a panoramic "virtual" camera with a single “virtual” viewpoint. Before 3D processing is applied, we must find the viewpoint and the effective focal length of the panoramic virtual camera. We assume that the robots with panoramic sensors rest in a common planar surface, in our case the floor. The viewpoint of the virtual panoramic camera corresponds to a horizontal circle in the cylindrical image, which is the intersection of the cylindrical image and the horizontal plane passing through the viewpoint. The projection in the horizontal direction is a circular projection, and that in the vertical direction is a linear perspective projection. There are four basic camera parameters for 3D calculation: the effective focal length in the horizontal direction (i.e. the radius of the cylindrical image), the effective "focal length" in the vertical direction, the vertical coordinate of the horizon circle, and the height of the virtual camera. These parameters are estimated by a simple calibration algorithm using more than three known points in the scene [10].

Adaptive Panoramic Stereo

Adaptive panoramic stereo vision is formed by two panoramic cameras residing on two separate (mobile) platforms, and has three nice properties: omnidirectional (360 degree) 3D estimation, adaptive baselines and viewpoints, and dynamic “mutual calibration”. Using the cylindrical panoramic camera model described above, we are going to deal with two virtual cylindrical imaging sensors in the following. For simplicity let us assume that both of them are subject to planar motion on the floor and are at the same height above the floor. However the idea of adaptive panoramic stereo vision and mutual calibration described below can be extended to the case of 6-DOF robot motion, for example, by employing more than three robots as in [6].

Suppose that O_1 and O_2 are the viewpoints of the two cameras and they can be localized by recognizing each other in the panoramic images as P_{21} and P_{12} (Fig. 2a). Baseline B is the distance O_1O_2 between the two sensor viewpoints. The projections of a target point T in the two panoramic images are represented by T_1 and T_2. Then a triangle O_1O_2T can be formed. By defining an arbitrary starting orientation for each cylindrical image, the three inner angles ϕ_1, ϕ_2 and ϕ_0 of the triangle can be calculated from the following four bearing angles: the bearings of the target in image 1 and image 2 respectively; and the bearing angles of camera 1 in image 2, and camera 2 in image 1 respectively. Then the distances D_1 and D_2 from the two cameras to the target can be calculated by the standard *triangulation method* as

$$D_1 = B\frac{\sin\phi_2}{\sin\phi_0} = B\frac{\sin\phi_2}{\sin(\phi_1+\phi_2)}, \quad D_2 = B\frac{\sin\phi_1}{\sin\phi_0} = B\frac{\sin\phi_1}{\sin(\phi_1+\phi_2)} \qquad (1)$$

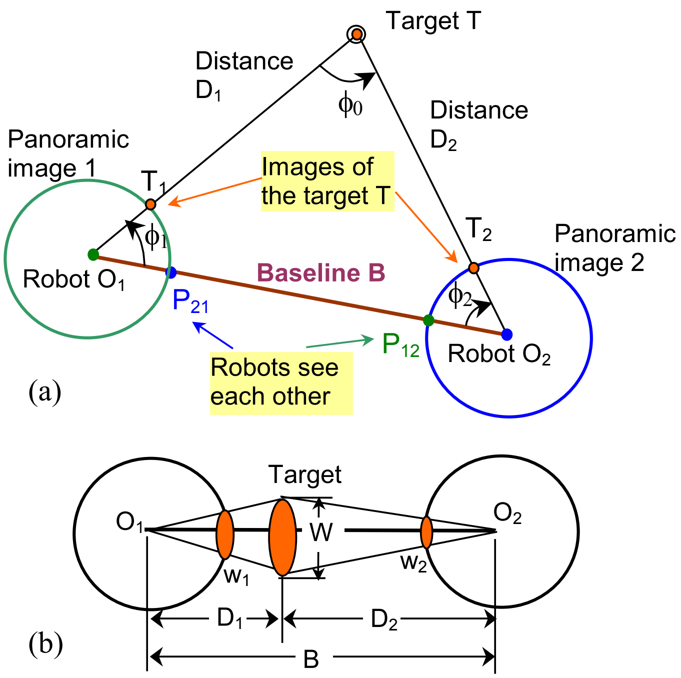

Fig. 2. Panoramic stereo geometry – top views of robots O_1 and O_2, baseline B, and target T: (a) triangulation method; (b) size-ratio method

Errors of panoramic stereo vary when the target is in different locations if the locations of the two cameras are fixed – larger errors occur when the target is close to the baseline and smaller errors when better triangulation is formed. In our case of adaptive panoramic stereo where the viewpoint and baseline relation can change, it is interesting to find the best configuration for estimating the distance of a target. First we will show that panoramic stereo can estimate *almost always* the distance of the target in the full 360° view. It is commonly known that the triangulation relation is invalid if the target is aligned with the baseline O_1O_2 (i.e. a singularity in the triangulation process). Fortunately, colinearity of the sensors and the target can be easily verified by checking the bearing angles of the images of the two robots seen by each other. In this case, the 3D location of the target can still be estimated by using the *size-ratio* method as

$$D_1 = B \frac{w_2 \cos\phi_2}{w_1 \cos\phi_1 + w_2 \cos\phi_2} \cos\phi_1 , \; D_2 = B \frac{w_1 \cos\phi_1}{w_1 \cos\phi_1 + w_2 \cos\phi_2} \cos\phi_2 \qquad (2)$$

where w_1 and w_2 are the widths of the target in the panoramic image pair. Note that the cosines in the above equations only give signs since the angles are either 0° or 180°, depending on if the target is between the two robots, or to left or right side of both of them. In Fig. 2b, the target is between the two robots, and both ϕ_1 and ϕ_2 are zeros, therefore we have $\cos\phi_1 = \cos\phi_1 = 1$. In the *size-ratio method*, since the two cameras view the human target from exactly the opposite direction, the widths of the symmetric approximately convex human target in the two images corresponds to a scale factor of *almost* the same width in 3D space (Fig. 2b). The bounding contours of an approximately symmetric figure make the calculation plausible. As an alternative, we can also use the height information (in the same way as we use width) since the height of an object is more invariant from significantly different views.

Dynamic Mutual Calibration

In order to estimate the distance of a target, we need first to estimate the baseline and the orientation angles of the two panoramic cameras. In stereo vision, an *epipole* is defined as the projection of one camera center in the other camera's image plane. In a stereo vision system with normal FOVs, epipoles are usually outside the FOVs in the other cameras, therefore it must use a third target in the scene for stereo calibration. In contrast, panoramic stereo has two "visible epipoles" (P_{21} and P_{12} in Fig. 2a) in that the two panoramic cameras can see each other while looking at the environment simultaneously. We propose a special dynamic calibration procedure called *mutual calibration* [9, 10] based on the visible epipole property in panoramic stereo. Mutual calibration neither needs to set up any additional calibration targets, nor requires the use of a third object in the environment. Instead each of the panoramic cameras can use the other as the calibration target. The advantage of "sensors as calibration targets" in mutual calibration is that the geometric structures and the photometric properties of the sensors, as well their platforms can be well designed and are known *a priori*.

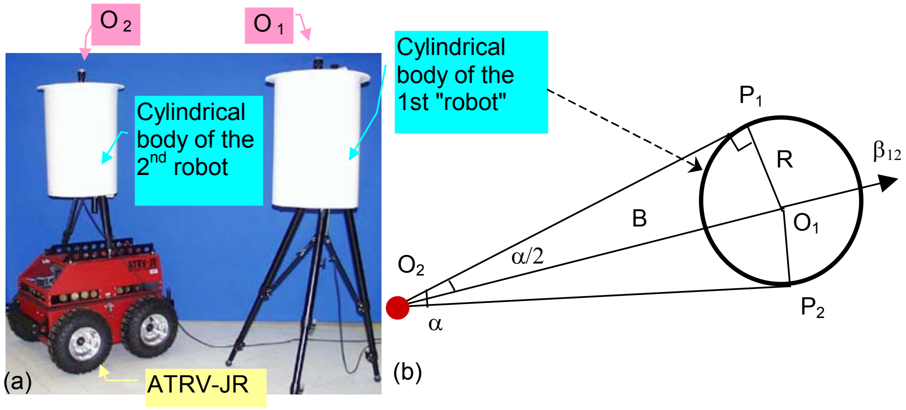
O 2
O 1
Cylindrical body of the 1st "robot"
Cylindrical body of the 2nd robot
ATRV-JR
(a)
(b)
P1
R
β12
B
O1
O2
α/2
α
P2

Fig. 3. Mutual calibration: finding the orientation and the distance between two panoramic sensors. (a). Setup of two robot bodies and cameras; and (b) top view of camera 2 viewing robot 1.

We have studied several practical approaches for this purpose by using special structures, such as cylinders, vertical bars and rectangular planar surfaces [9]. The basic idea is to make the detection and calculation robust and fast. One of the simplest approaches is to design the body of each robot as a cylinder with a distinctive appearance (e.g. white in the intensity images of our current implementation), which can be easily seen and extracted in the image of the other robot's camera (Fig. 3a). We assume that the rotation axis of each panoramic camera is coincident with the rotation axis of the cylindrical body of the corresponding robot. The baseline B between the two panoramic cameras can be estimated using the occluding boundary of either of the two cylinders, e.g., from the image of camera 2 we have (Fig. 3b)

$$B = R/\sin(\frac{\alpha}{2}) \qquad (3)$$

where α is the angle between two occluding projection rays of the target cylinder of robot 1 measured in the image of camera 2, and R is the known radius of that cylindrical body (e.g., R=0.18 meters in our experiment). The orientation angle (β_{12}) of the baseline O_2O_1 is simply the average of the bearings of two occluding boundary points P_1 and P_2. We can perform the same calibration in the image of camera 1. For an original circular image of size 768*576, the maximum perimeter of the cylindrical image is about 1800 pixels (=2π 576/2); therefore the highest angular resolution of our panoramic images in degrees would be 360/1800 = 0.2 degree/pixel.

Error Characteristics and View Planning

It is important to understand the accuracy of any 3D localization process in the mobile robot applications we consider. Previous work (e.g. [4]) only gave the error distribution of the panoramic stereo with a fixed stereo geometry. Shum et al [7] studied the case of an omnidirectional camera moving within a circular region of the plane, and concluded that it was the best to choose a pair of cameras that are convergent on a point with maximum convergence angle in order to accurately localize the point. We have discussed a more general case where the spatial relationship between the two panoramic cameras can change arbitrarily, and we address the issue of deriving optimal configurations using a different error formulation [10]. Given the location of camera 1 at distance D_1 from the target, and applying our virtual stereo formulation, the key goal is determination of the error distribution of this distance as the location of camera 2 (and hence the configuration of baseline B and panoramic stereo angles) is varied. An obvious consideration is whether we can always achieve better distance estimation for distant targets with a larger baseline, which in turn needs dynamic calibration for the sensor pair, for example by our mutual calibration method. More generally, we seek a computational methodology for controlling a team of robots moving through an environment. In particular, we wish to provide capability of constraining the motion of one robot in the environment relative to another for localization of a target at any visible location, such that virtual panoramic stereo error will be minimized .

Eq. (1) and Eq.(2) show that the accuracy of distance estimation depends on the accuracy in estimating the baseline and the bearing angles. After careful mathematical analysis with several different cases of robot-robot-target relations [10], we arrive at the following conclusion in a most common case:

Conclusion 1. If the distance from camera 1 (the main camera) to the target, D_1, is significantly greater (e.g. 10 times greater) than the size of the robot (R), the best geometric configuration is

$$B \approx 2\sqrt{D_1 R}\,, \qquad \cos\phi_1 = \frac{3BD_1}{2D_1^2 + B^2} \qquad (4)$$

Eq. (4) indicates that the best baseline distance for minimum distance error is proportional to the square root of the distance of the target to the main camera; meanwhile the best baseline is also a square root function of the robot size, represented here by the radius of the robot. We remind readers that these specific "square root" relations are derived using our mutual calibration method, where error in calibration increases as a function of distance between cameras, i.e. baseline B. However the conclusion loosely holds for any adaptive stereo requiring dynamic calibration in the sense that :

1) more accurate baseline estimation can be obtained given a larger robotic target (i.e. larger R), since two cooperative robots *would* see each other better, hence the optimal baseline for estimating the target distance *can* be larger, and 2) the greater the target distance D, the larger the baseline *should* be. However, as the baseline distance increases, the accuracy decreases in estimating the baseline by dynamic calibration. Thus, there are competing constraint factors in constraining the optimal solution.

Let us examine the distance error against the absolute distance of the target to gain a better understanding of the relationship between baseline (B), distance (D_1) and robot size (R). In our experimental setting, Conclusion 1 holds for any target that is two meters away from the main camera (when the radius of the robot is R = 0.18 m). In this case, the distance between the two cameras is smaller than the distance of the target from the first camera, i.e., $B < D_1$, and we have found that the error under this optimal configuration (as stated in Conclusion 1) is proportional to the 1.5 power of the distance, $D^{1.5}$ as in Eq. (5):

$$\partial D_1 < 2D_1\sqrt{\frac{D_1}{R}}\partial\phi \qquad (5)$$

where $\partial\phi$ is the angular error in target measurement in the images. Eq. (5) also says that the distance error is inversely proportional to the square root of the robot size (R), which means that larger robot body leads to more accurate distance estimation. The reason stems from Eq. (4) of Conclusion 1, which says that larger robot size will give us a larger baseline distance in the best stereo view configuration, and consequently larger baseline distances will lead to a more accurate estimation of target distance. Of course higher distance accuracy can be achieved with higher image resolution under the same robot size.

It is also interesting to compare the adaptive panoramic stereo with fixed baseline stereo (a stereo head). Assume that in a fixed baseline stereo system of a robot, two cameras are mounted as far away as possible in a robot with a cylindrical body of radius *R*, so the maximum baseline is *B=2R*. Let us assume that there is no error in stereo camera calibration (i.e. the baseline *B* is *accurate*). We always have B < D_1 in fixed-baseline stereo, and in the best case [10] the distance error is

$$\partial D_1^{fix}\big|_{B=2R} \approx \frac{D_1^2}{R}\partial\phi \qquad (6)$$

Comparing Eq. (6) with Eq. (5), we reach the following conclusion (for a more accurate statement see [10]:

Conclusion 2. The depth error of the adaptive stereo vision is proportional to 1.5 power of the camera-target distance ($D^{1.5}$), which is better than the case of the best possible fixed baseline stereo in which depth error is proportional to the square of the distance (D^2).

Conclusion 2 tells us that adaptive panoramic stereo can achieve more accurate distance estimation for distant object because the adaptive baseline configuration, even though the dynamic calibration involves error. What is happening here is that with mobile robots, they can *reactively* move as a function of target location and distance to an optimal geometric configuration.

The conclusions above can be used in optimal view planning for mobile robots. To illustrate the analysis, we represent the distance error map under different viewpoint locations of camera O_2 in Fig. 4. Although the error map was plotted for the case of D_1= 34R = 6m, the error as a function of location and angle to the second camera generally holds. The colors in the figure represent different ranges of error as robot O_2 might move in the environment (note the vertical error bar on the right side of Figure 4.) Minimum error is achieved in a stereo configuration with *B*=2.2 m and $\phi_1 = 62.1^\circ$. (One of the best locations for O_2 is at a vertex of the triangle O_1TO_2, which is also pointed to by an arrow in Fig. 4; note that we have two such symmetric locations). The error map also shows that how the second camera can move to a better location for more accurate distance estimation: after initial estimation of the target distance from the main camera is made under any initial geometric stereo configuration, the adaptive stereo vision system can use this initial estimation to drive the second robot to approach the best location to minimize the error in distance estimation. Even if the target is moving (which is supposed to be), one or both of the cameras can move to track the moving target in a feedback control loop to approach the best stereo configuration guided by the error map.

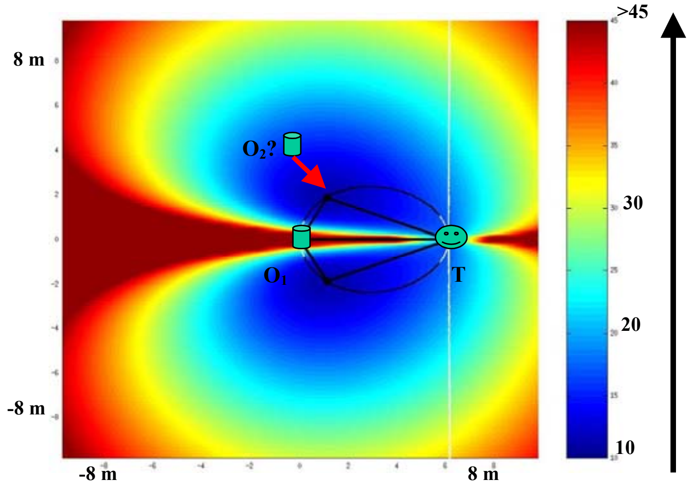

Fig. 4. Error map for distance D_1 when camera O_2 is in different locations of the map by fixing camera O_1 and the target T ($D_1 = 34R = 6m$, R = 18 cm). The labels on the two axes are distances (in meters); each point on the elliptic curve indicates where the minimum error is achieved for viewpoint O_2 on a circle with radius R around O_1 , under the given baseline B; different baselines will have different minimum errors, and the "global" minimum is achieved under the constraints of Conclusion 1 when the second camera O_2 is in the location pointed to by the red arrow. The error value ($\partial D_1 / D_1 \partial \phi$) is encoded in color – red means large error and blue means small error: see the vertical color error bar.

Multiple Moving Object Localization

In a human tracking and localization application, it often will be necessary to detect multiple moving objects and simultaneously determine their 3D locations. This requires an ability to match primitives, in our case blobs representing humans (or other moving objects), extracted from a pair of panoramic images and unambiguously determining correspondences. The bearing angles of the blobs in both images (as well as the orientations and distances of the two cooperative robots) is used for panoramic triangulation, whereas the size-ratio method also needs the width of each blob as well as the bearing angles.

This section will discuss how to define and reliably detect moving image blobs for stereo matches. Since the difficulty of stereo correspondence increases significantly when the images of the same object differ in size and appearance due to large perspective distortions from rather different viewing angles, the selection of matching features in the blobs is crucial for 3D localization. Note that we demonstrate our matching approach initially by assuming stationary cameras for using background subtraction, a technique readily applicable to a sensor network with many distributed stationary sensors. However, the general matching method could be applied to freely moving platforms with the use of the dynamic mutual calibration module, and the use of new modules for extracting moving objects from moving platforms, as in our recent work [8].

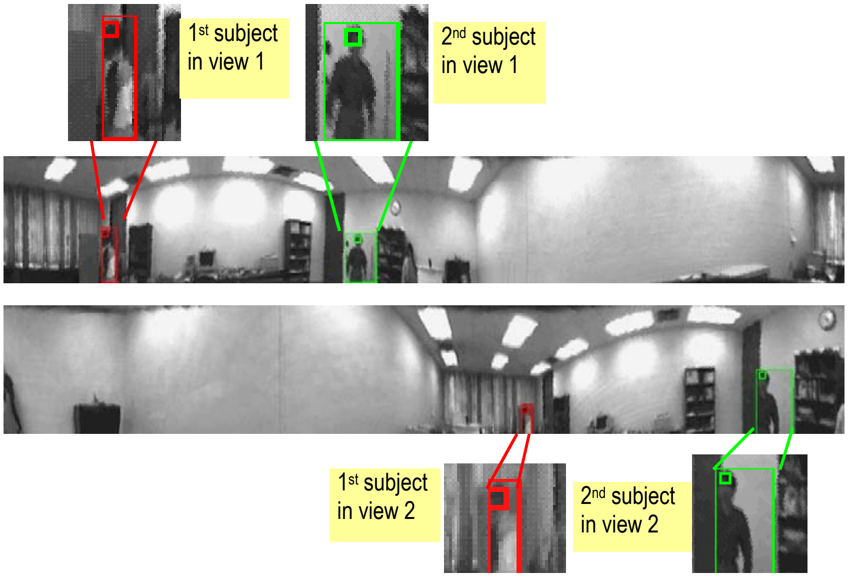

Fig. 5. Head extraction and bearing estimation. The large rectangle on each human subject is the bounding rectangle of each blob, and the small rectangle inside indicates the centroid of the head.

Moving object detection and extraction

Given a stationary camera, moving objects can be detected and extracted by the standard technique of background subtraction, obtained by subtracting the current image from a background image acquired through averaging of pixels in a stationary environment. However, during a long monitoring period, illumination of the background may change, either gradually or abruptly, which may introduce false detection. We deal with this problem by integrating background subtraction, frame differencing, failure/abnormal detection and background updating [9]. The subtraction image from background subtraction is analyzed by a grouping algorithm that generates *blobs* by merging connected and nearby regions to produce a contour of possible moving object candidates. Each of the extracted moving object blobs is checked to determine that enough pixels in the difference image have changed inside each region of a candidate target. A *blob image* is produced by selecting only the intensity values of the pixels for validated moving objects. During the detection process, the background image is updated pixel-wise by a weighted average of the existing background pixel and the current image *only in the non-object regions*. The frame rate for multiple object detection (and tracking) is about 5 Hz in a Pentium 333MHz PC for 1080*162 panoramic images, and therefore achieves effective real-time processing merely by upgrading the CPU.

Head extraction and blob representation

We have found in experiments that the bearing of the centroid of an entire blob is subject to the changes of the positions of arms and legs, and errors in body extraction. We also noticed that the bearing of a person's head is more accurate than the entire blob of the subject for three reasons: (1) the head is usually visible in the panoramic images; (2) the head contour is almost symmetric from all directions of the robot's viewpoints; and (3) it is easy to extract from the background. The quasi-symmetry property of a human head makes it more suitable for matching across two widely separated views. The primary idea behind head extraction is to select invariant features of human subjects for stereo matching, which can be further extended by extracting different parts of a human blob for partial match between two views.

The head part of a blob is extracted by using the knowledge that it is the topmost portion and roughly has a fixed height-width ratio (e.g., 3:2) in a panoramic image. Here the exact height of the head part is not critical since we only use the bearing angle of the head for triangulation. Fig. 5 shows results of human blob and head extraction from a pair of panoramic images. It can be seen that the bearings of heads are more suitable for building up correspondence between a pair of human blobs from two widely separated views. From each panoramic image, a set of objects (blobs) is extracted, which is represented by a vector of several parameters $\mathbf{T}^{(k)} = \{T_i^{(k)} = (I_i^{(k)}, \theta_i^{(k)}, w_i^{(k)}, h_i^{(k)}), i = 1, \ldots, N_k\}$ where k (1 or 2) denotes the camera view, $I_i^{(k)}, \theta_i^{(k)}, w_i^{(k)}, h_i^{(k)}$ are the photometric feature, bearing angle of the head of the target *i* in camera *k*, the width of the image blob, and the vertical coordinate of the top of the blob (indicating the height of the human).

3D Blob-Based Stereo Match

Since the appearance of an object will vary significantly from largely separated views, the information only from 2D images may produce ambiguity or inaccuracy in object matching. We will explain our 3D matching algorithm based on *annotated image blobs* of human subjects. Extension of this method to the contours of humans and other objects is straightforward.

Based on the annotated blob representation, we have explored the 3D-related measurements as well as 2D photometric features. For each object blob *i* in $\boldsymbol{T}^{(1)}$, we will perform a match with every object blob *j* in set $\boldsymbol{T}^{(2)}$, and we derive the following confidence measurements for each pair of possible matches [10]:

- *Degree of intensity similarity* $r_{is}(i, j)$ measures the photometric similarity between a pair of blobs;

- *Degree of ray convergence* $r_{rc}(i, j)$ indicates that a meaningful match must satisfy the condition that two rays from the viewpoints to the target images converge;

- *Degree of width consistency* $r_{wc}(i, j)$ indicates that matching $T_i^{(1)}$ and $T_j^{(2)}$ after accounting for recovered distances $D_i^{(1)}$ and $D_j^{(2)}$ to hypothesized targets in viewpoints O_1 and O_2 by using Eq. (1) or Eq. (2), the size of the hypothesized target estimated from two images should be very close for a correct match; and

- *Degree of height consistency* $r_{hc}(i, j)$ uses the ratio of the estimated 3D heights of the top of the hypothesized object from both images, and should be close to 1;

Then a match measurement matrix **M** = $[r(i,j)]_{N1xN2}$ can be constructed where the element indexed by (i,j) is the total match confidence measurement of the stereo match $i \leftrightarrow j$, i.e.

$$r(i,j) = r_{is}(i,j) r_{rc}(i,j) r_{wc}(i,j) r_{hc}(i,j) \in [0,1] \qquad (7)$$

Note that in the match features, all items except the first one contain 3D information, namely orientation constraints, width constraints or height constraints.

The task of our blob match algorithm is to find for each object in the first panoramic image the correct match in the second panoramic image based on the match matrix. We have implemented a simple but effective "greedy" match algorithm for stereo matches of primitives across two widely separated panoramic views [10]. The “greedy” algorithm is very fast even if many moving objects (10 ~ 20) are detected in the scenes. The time complexity of this algorithm is O($\frac{1}{3}N^3$) where $N=max(N_1, N_2)$. However, this algorithm may not be able to find (correct) matches for some of the objects due to missing detection, occlusion and view distortions, and a more sophisticated global matching algorithm was given in our technical report [10]. In the improved matching algorithm, a global confidence measure is maximized over all the possible match pairs.

Fig. 6 shows an example where two moving objects were detected in panoramic image 1 (Pano 1) and three moving objects were detected in Pano 2. Fig. 6(c) shows the geometrical relation between images and 3D object locations. By applying the "greedy" match algorithm or the global match algorithm, the correct final matches are $(T_1^{(1)}, T_2^{(2)})$ and $(T_2^{(1)}, T_1^{(2)})$. The estimates of distances and heights of the two people are labeled in Fig. 6(c). The stereo match can be improved by using temporal tracking information [10]. The temporal tracking information is incorporated by adding a temporal match consistency term to each final confidence measurement $r(i,j)$. In this way, the correct matches of the previous frames enhance the match confidence measure in the current frame. The match algorithms are still applicable by just replacing the stereo match measures by the new measurements.

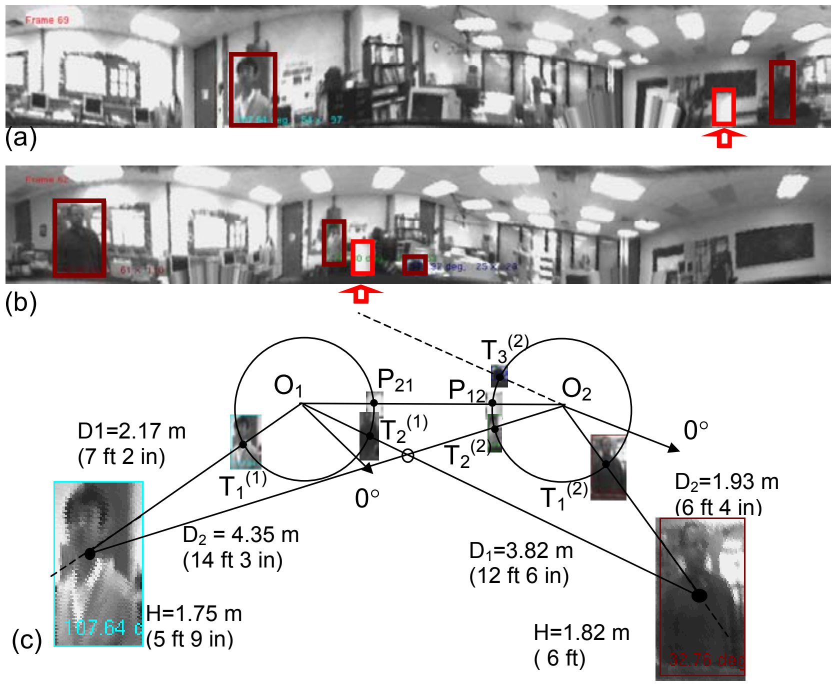

Fig. 6. 3D estimation of multiple objects ($T_i^{(k)}$: ith blob of kth camera, k= 1,2; P_{21} and P_{12}: projections of the 1st (2nd) cylindrical platforms in the 2nd (1st) panoramic images). (a) Pano1 - from left to right: blobs (with bounding boxes) $T_1^{(1)}$, P_{21} , $T_2^{(1)}$; (b) Pano 2- from left to right: $T_1^{(2)}$, $T_2^{(2)}$, P_{12}, $T_3^{(2)}$; (c). Geometric relation of 2D images and 3D objects

Experimental System and Results

In our experimental system, we mounted one panoramic camera on an RWI ATRV-Jr. robot, and the other panoramic camera on a tripod (Fig. 3a). Two Matrox-Meteor frame grabbers, each connected to a camera were installed on the ATRV-JR and a desktop PC respectively, both with 333M Hz PII processors. The communication between the two platforms is through sockets over an Ethernet link (wireless Ethernet communication is now used). 3D moving object detection and estimation programs run separately on the two machines at about 5 Hz. Real-time processing is obviously feasible with newer CPUs. Only camera and object parameter data (i.e., baseline, bearing angles, sizes, and photometric features were transmitted between two platforms so the delay in communication can be ignored at the current processing rate (5Hz). In the implementation of experiments shown here, we assumed that the most recent results from both platforms correspond to the events at same time instant. Temporal synchronization of image blobs is achieved now by using network time protocol (NTP) and by data interpolation in order to avoid the time/motion delay of moving objects in two images [8].

Fig. 7 shows results of real-time 3D tracking and localization of a one and two moving subjects. The experiment in Fig. 7a was designed to evaluate the panoramic stereo's localization accuracy when a single person was walking along a known rectangular path when the two cameras were stationary. Each red dot represents a location of the person. The six dense clusters of dots show the six locations where the person purposively made turns at each such location. We used two methods to localize the moving subject – the triangulation method when a good triangle among the target and the two cameras can be formed, and the size-ratio method when the target was near locations of colinearity. The target (T) position where the theoretical best triangulation on this track can be expected is shown in the figure, which is consistent with the real experimental results. Even when the localization errors in images could be greater than 1 pixel, the average error of the estimated track is ±10 cm, which is comparable to the theoretical error bounds.

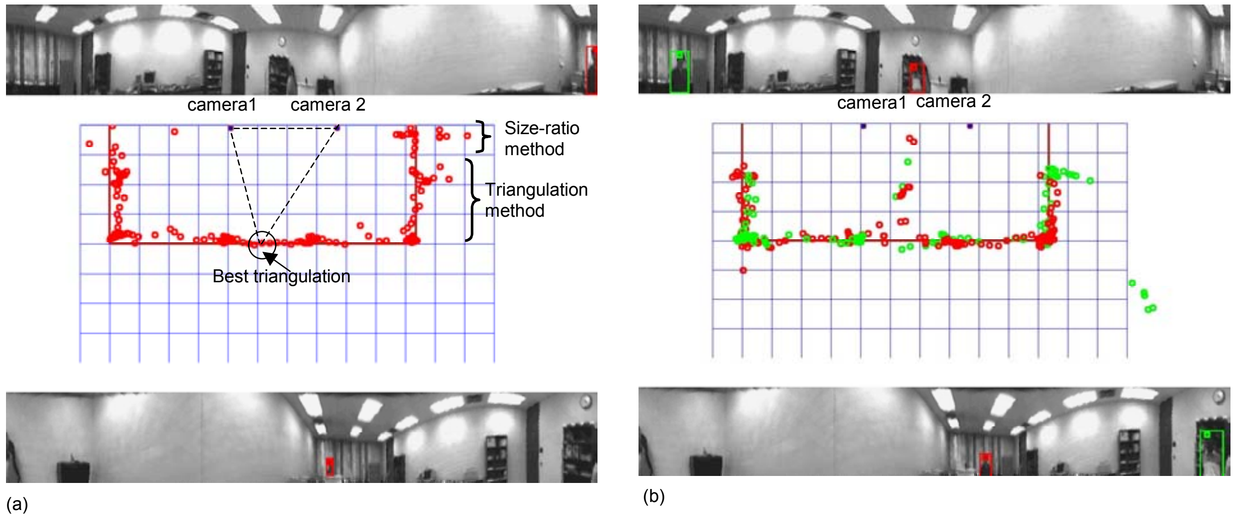

Fig. 7. Panoramic stereo tracking results of (a)one subject and (b) two subjects. In each case, the images at the top of the bottom are the panoramic image pair from two panoramic cameras respectively, and in the middle are the real localization results plotted in a top view of the room where each grid is 50x50 cm^2. Each red /green dot represents a location of the person during motion.

Fig. 7b shows the results of detecting and tracking two people who walked from opposite directions along the same known rectangular path. In this example, the simple “greedy” match algorithm was used. In the 2D map of the room (center of Fig. 7b), the red dot sequence shows the path of one person, and the green dot sequence shows that of the other. The proposed 3D match, localization and tracking algorithms produced rather good results with consistent 3D localization for the two people. The average localization error is about 20 cm.

Conclusion and Discussion

In this article we present an adaptive panoramic stereo approach for two cooperative mobile platforms. There are four key features in our approach: (1) omni-directional stereo vision with an appropriate vertical FOV, and a simple camera calibration method, (2) cooperative mobile platforms for mutual dynamic calibration and best view planning, (3) 3D matching after meaningful object (human subject) extraction and (4) real-time performance. The integration of omni-directional vision with mutual awareness and dynamic calibration strategies allows intelligent cooperation between visual agents. A 3D-based matching process provides a nice way to solve the problems of limited resources, view planning, occlusion, and motion detection of movable robotic platforms. Experiments have shown that this approach is quite promising.

Based on the adaptive panoramic stereo vision approach, a distributed sensor network architecture comprising of three levels of hierarchy has been proposed in our recent work of a smart room project with mobile platforms and heterogeneous sensors [8]. The hierarchy consists are sensor nodes, resource manager and user agents. The resource manager acts as a proxy between the sensor nodes and the user agents allowing many user agents to simultaneously share sensor resources. The system was currently implemented using two types of vision sensors, panoramic cameras and pan-tilt-zoom cameras. A simple, cost-effective way of synchronizing data streams from heterogeneous sensors using network time protocol (NTP) has been discussed and the experimental results showed the practical utility of this approach. Adaptive panoramic stereo is one of the important modules for localizing multiple moving human subjects. The system is being evaluated for its fault-tolerance performance and accuracy of tracking, where the error modeling and the view planning strategy developed in this paper is being employed.

References

[1]. VSAM- Video Surveillance and Monitoring Session, *DARPA Image Understanding Workshop Proceedings*, Monterey, November 1998

[2]. S. Baker and S.K. Nayar, "A theory of single-viewpoint catadioptric image formation," *International Journal of Computer Vision*, Vol. 35, No. 2, pp. 1 - 22, 1999.

[3]. P. Greguss, "Panoramic imaging block for three-dimensional space," *U.S. Patent 4,566,763*, 28 Jan, 1986.

[4]. K. Yamazawa, Y. Yagi and M. Yachida, "Omnidirectional imaging with hyperboloidal projections," in *Proc. IROS'93* , pp.1029-1034, vol.2, July 1993.

[5]. T. Sogo, H. Ishiguro, M. M. Trivedi. "N-Ocular Stereo for Real-time Human Tracking," *Panoramic Vision: Sensors, Theory and Applications*, (R. Benosman and S. B. Kang, eds.), Springer Verlag, 2000.

[6]. J. Spletzer, A.K. Das, R. Fierro, C.J. Taylor, V. Kumar, and J.P. Ostrowski, "Cooperative localization and control for multi-robot manipulation," in *Proc. IEEE/RSJ IROS'01,* pp. 631-636*,* Maui, Hawaii, Oct. 29 - Nov 3, 2001 .

[7]. H.-Y. Shum, A. Kalai and S. M. Seitz, "Omnivergent stereo," in *Proc. ICCV'99*, pp 22 – 29, September 1999.

[8]. D. R. Karuppiah, Z. Zhu, P. Shenoy and E. M. Riseman, "A fault-tolerant distributed vision system architecture for object tracking in a smart room," presented at *ICVS'01*, Vancouver, Canada, July 2001. B. Schiele and G. Sagerer (Eds.), Springer Lecture Notes in Computer Science 2095, pp 201-219.

[9]. Z. Zhu, E. M. Riseman and A. R. Hanson, "Geometrical modeling and real-time vision applications of panoramic annular lens (PAL) camera," *Tech. Rep. TR #99-11*, CS Dept., UMass-Amherst, Feb., 1999.

[10]. Z. Zhu, K. D. Rajasekar, E. Riseman, A. Hanson. "Panoramic virtual stereo vision of cooperative mobile robots for localizing 3D moving objects," in *Proc. IEEE Workshop on Omnidirectional Vision*, Hilton Head Island, pp 29-36, June 2000. An extended version can be found as *Tech. Rep. TR #00-14*, CS Dept, UMass-Amherst, March 2000.