3D Localization of Multiple Moving People by an Omnidirectional Stereo System of Cooperative Mobile Robots*

Zhigang Zhu, K. Deepak Rajasekar

Edward M. Riseman, Allen R. Hanson

UM-CS-2000-014
March, 2000

Computer Vision Laboratory

Department of Computer Science

University of Massachusetts at Amherst

Amherst, MA 01003

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

---

* This work was supported by AFRL/IFTD under contract numbers F30602-97-2-0032 (SAFER), and DARPA/ITO DABT63-99-1-0022 (SDR Multi-Robot), and by the Army Research Office under grant number DAAD19-99-1-0016.

Keywords: Panoramic imaging, omnidirectional vision, multiple mobile robots, human detection and tracking, cooperative stereo, dynamic calibration, view planning, error analysis

Abstract

Flexible, reconfigurable vision systems can provide an extremely rich sensing modality for sophisticated multiple robot platforms. We propose a cooperative and adaptive approach of panoramic vision to the problem of finding and protecting humans by a robot team in an emergency circumstance (e.g. a rescue in an office building). A panoramic virtual stereo vision method is proposed for this cooperative approach, which features omni-directional visual sensors, cooperative mobile platforms, selected 3D matching, and real-time moving object (people) detection and tracking. The problems of dynamic self-calibration of moving platforms, robust 3D localization of moving human subjects, and cooperative strategies between two robots are discussed. We have found that the localization errors of a human target by the panoramic virtual stereo is subject to three different kinds of errors: the calibration error of the panoramic virtual stereo, matching error of widely separated views, and triangulation error of the panoramic stereo pair. A careful numerical analysis of the error characteristics of the panoramic virtual stereo is presented in order to derive rules for optimal view planning of moving sensing platforms (mobile robots) or multiple (more than two) stationary sensing platforms. Experimental results are given for detecting and localizing multiple moving objects using two cooperative robot platforms.

I. Introduction

Flexible, reconfigurable vision systems can provide an extremely rich sensing modality for sophisticated robot platforms. We propose a cooperative and adaptive approach to the problem of finding and protecting humans in emergency circumstances, for example, during a fire in an office building. Real-time processing is essential for the dynamic and unpredictable environments in our application domain, and it is important for visual sensing to rapidly focus attention on important activity in the environment. Any room or corridor should be searched quickly to detect people and fire. Field-of-view issues using standard optics are challenging since panning a camera takes time, and multiple targets/objectives may require saccades to attend to important visual cues. Using multiple cameras covering different fields of view could be a solution, but the cost of hardware (cameras, frame grabbers and computers) and software (multiple stream data manipulation) will increase. Thus, we employ a camera with a panoramic lens to detect and track multiple objects in motion in a full 360-degree view in real time.

We note that there is a fairly large body of work on detection and tracking of humans (Bri98; Har98; Lip98; Pap98; Pen97), motivated most recently by the DARPA VSAM effort. On the other hand, different kinds of omni-directional (or panoramic) imaging sensors have been designed (Nay97; Gre86; Nel96; Yag90; Yam93), and a systematic theoretical analysis of omni-directional sensors has been given (Bak98). Omnidirectional vision has become quite popular with many vision approaches for robot navigation (Yagi90; Zhu98), stereo reconstruction (Ish92; Kon98) and video surveillance (Bou99; Ng99). What is truly novel about our approach is the ability to compose cooperative sensing strategies across the distributed panoramic sensors of a robot team to synthesize robust "virtual" stereo sensors for human detection and tracking.

The idea of distributing sensors and cooperation across different robots stems from the requirements of potentially limited (sensor) resources for a large robot team. Nevertheless, the advantages of cooperative vision arise from more than this compromise. 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 the planning system to take advantage of the current context and goals in selecting viewpoints. This strategy can be implemented by a single camera generating sequential viewpoints over time in an active vision paradigm (Alo93). However, there are significant time delays involved in moving the camera to another position in the room as well as the difficulty of dynamic calibration.

In this paper, we focus on cooperative behavior involving cameras that are aware of each other, residing on different mobile platforms, to compose a virtual stereo sensor with a flexible baseline. In this model, the sensor geometry can be controlled to manage the precision of the resulting virtual sensor. The cooperative stereo vision strategy is particularly effective with a pair of mobile panoramic sensors that have the potential of almost always seeing each other. Once calibrated by "looking" at each other, they can view the environment to estimate the 3D structure of the scene.

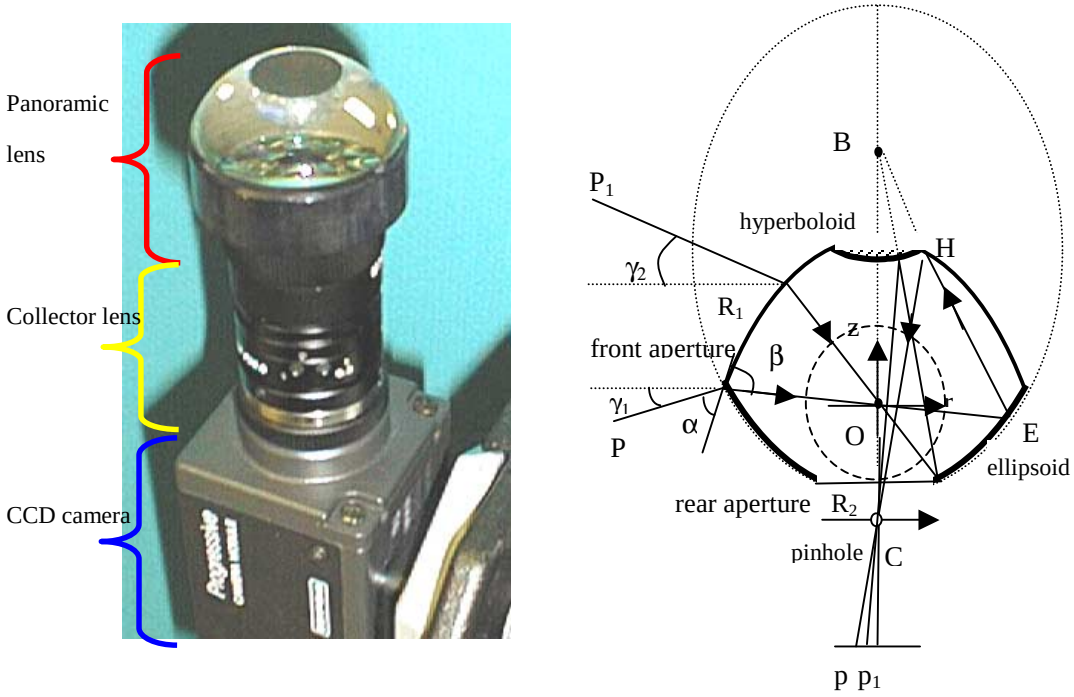
In the following sections, we will discuss the following critical issues: 1) the calibration and image warping of an omnidirectional vision system, 2) dynamic self-calibration among the two cameras on two separate mobile robots, which forms the dynamic "virtual" stereo sensor, 3) view planning by taking advantage of the current context and goals , 4) detection and tracking of moving objects from a stationary platform as well as from a moving platform (robot), and 5) correspondence of features between two views, given the possibly large perspective distortion, and 3D estimation.

II. Panoramic Imaging Geometry

2.1. Sensor geometry

In this paper we use the panoramic annular lens (PAL) camera system designed by Pal Greguss (1986) for its compactness and view angles. It can capture its surroundings with a field of view (FOV) of 360-degrees horizontally and -15 ~ +20 degrees vertically (Fig. 2.1). We have noticed

that many other omnidirectional cameras have their vertical FOV either entirely above or below the horizon. In a robotic application, a vertical viewing angle that spans the horizon is preferred. The PAL-3802 system that we are using includes a compact 40-mm diameter panoramic lens and a built-in 16-mm collector lens with a "C" mount (Fig. 2.1a). The panoramic lens is a piece of glass that consists of a 360-degree circular aperture (R_1), a rear aperture (R_2) connecting to the collector lens, a top mirror (H) and a circular mirror (E) (Fig. 2.1b). The geometry of the PAL imaging system is somewhat complex since there are two reflections and two refractions. Fortunately, we can obtain a rather elegant geometry of a single effective viewpoint under perspective projection (Gre86; Zhu99, Gre00) given that:



(a). PAL lens and camera (2) PAL lens geometry

Fig. 2.1. PAL lens and its geometric model

(1) the concave circular mirror (E) is *ellipsoidal* and the convex top mirror (H) is *hyperboloidal;*

(2) the long axis of the ellipsoidal mirror is aligned with the axis of the hyperboloidal mirror and the optical axis of the camera (C); and

(3) a locus (B) of the hyperboloidal mirror coincides with one locus of the ellipsoidal mirror, and the other locus coincides with the nodal point (C) of the real camera.

Thus, the viewpoint of the "virtual camera" is right at the second locus (O) of the ellipsoidal mirror[1]. Under this geometry, the PAL sensor can view the entire 360-degree scene around its vertical axis BC. The vertical field of view (Fig. 2.1) is determined by the effective sizes and the locations of the circular mirror E and the top mirror H. Usually the viewing angle is $[0^o, 90^o)$ vertically if we ignore the refraction effects.

The refraction does not add too much complexity; instead, it changes the vertical viewing angle. The first refraction through the ellipsoidal surface (R_1) changes the vertical viewing range from $[0^o, 90^o)$ to $[\gamma_1, +\gamma_2]$, where $\gamma_1<0^o<\gamma_2<90^o$, which is often desired for panoramic imagery in robotics applications. For the PAL-3802 camera system we have approximately $\gamma_1=-15^o$ and $\gamma_2=20^o$. The second refraction through the transparent planar surface (R_2) only moves the point of convergence of rays that are reflected from the top mirror up some distance.

In conclusion, a ray from a 3D point P is first refracted (from angle α to β) by an ellipsoidal surface R_1, and passes through one of the loci (i.e. the viewpoint O of the virtual camera) of the concave mirror E. Next, the ray is reflected by the ellipsoidal mirror E to its second locus B (which is also a locus of the hyperboloidal mirror H). Then it is reflected by the convex mirror H to the second locus of H, i.e. the nodal point C of the real camera. Thus the annular image in the target plane of the real sensor can be viewed as being captured by an omnidirectional "virtual" camera located at viewpoint O. The optics of real PAL lens could be more complicated than the above geometric model. However, this model can ease the difficulty in panoramic camera calibration and image transformation in the following subsections.

---

[1] An orthographic model has also been derived by Zhu, Riseman & Hanson (1999) where the hyperboloidal mirror is replaced by a paraboloidal mirror, and then a tele-centric lens is used instead of a pinhole camera.

Fig. 2.2 shows an image captured by the PAL sensor. The size of circular black hole in the center of a PAL image is decided by the size of the top mirror E and the size of the rear aperture (R_2) of the PAL block, and is roughly the projection of the top mirror. By analyzing the imaging geometry of the PAL image, we can find that the farther a point is from the image center, the higher the resolution in both radial and angular dimensions will be.

2.2. Image Unwarping and Rectification

Using the above the PAL geometry, we have developed the mathematical model (Zhu99) for calibrating the PAL camera, which shows that the calibration of such a camera turns to be a difficult problem of solving nonlinear equations of 16 parameters. Here we use an empirical method to transform the PAL image, which consists of two simple steps:

(1) Center determination **-** First, we adjust the camera to point vertically upward so that projections of vertical lines in the world remain straight in a PAL image and they intersect at a single point in the center of the PAL image[2] (Fig. 2.2). If more than two such lines are detected in an original PAL image, the center point can be determined by their intersection. Once we have the center (x_0,y_0) of the PAL image $I(x,y)$, a cylindrical panoramic image $I(\rho,\theta)$ can be generated by the following polar transformation (Fig. 2.3)

$$\rho = \sqrt{(x - x_0)^2 + (y - y_0)^2}, \qquad \theta = \tan^{-1}\frac{y - y_0}{x - x_0} \tag{2-1}$$

[2] The projections of vertical straight lines will be curves in a PAL image if this condition does not hold. In such a general case, the calibration of the PAL camera would be much more difficult. The method by Geyer and Daniilidis(1999) for the calibration of an orthographic camera in front of a paraboloid mirror using two sets of parallel lines could be extended to calibrate the PAL camera in the general case.

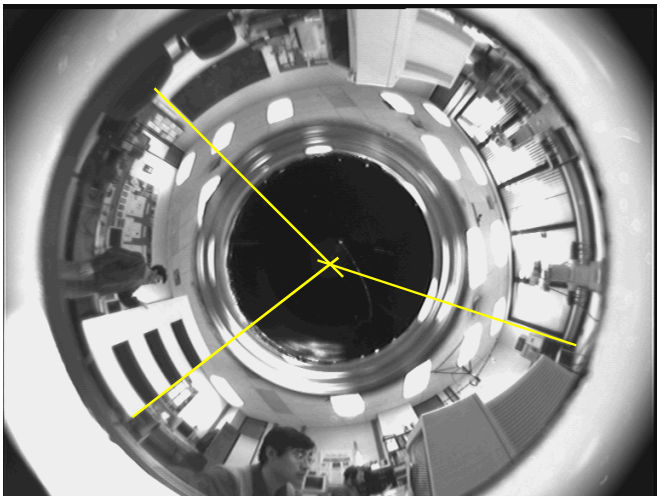


Fig. 2.2. Original PAL (Panoramic Annular Lens) image (768*576)

Fig. 2.3. Cylindrical panoramic image, without eliminating radial distortion

Fig. 2.4. Cylindrical panoramic image, after eliminating radial distortion

(2) Vertical distortion rectification **-** Distortion exists in the vertical direction of the unwarped cylindrical image (or the radial direction in the original PAL image) due to the non-linear reflection and refraction of the 2nd-order mirror surfaces. Note the unequal widths of the black-white bars on the white board in Fig. 2.3 caused by the vertical distortion (the widths are equal in the real world). We use an N-order polynomial to approximate the distortion along the vertical direction:

$$v = \sum_{i=0}^{N} a_i \rho^i \qquad (2\text{-}2)$$

where ρ is the vertical coordinate in the original cylindrical image, and v is in the rectified cylindrical image. By given the destination coordinates v_k of more than $N+1$ points ρ_k in the original cylindrical images (k=0,1,2…), we can compute the $N+1$ parameters a_i $(i=0,1,2, N)$ in Eq. (2-2), using the least mean square method. Fig 2.4 shows the rectification result using a 2nd-order polynomial approximation where only 3 point pairs are needed. We use the lease square method with more than 3 pairs of points. In this example, the destination black-white strips in Fig. 2.3 are constrained to have the same width and equals to the average strip width of the corresponding source strips. It is not surprising that we have the equal intervals of the black-white strips in the rectified cylindrical image. This distortion removal procedure gives us a good approximation of a linear perspective projection in the vertical direction of the panoramic images.

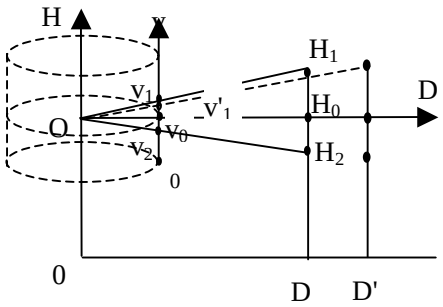


Fig. 2.5. Intrinsic parameter estimation

2.3. Calibration of the virtual cylindrical imaging sensor

After image unwarping and distortion rectification, we have a cylindrical image generated by a panoramic "virtual" camera from the virtual viewpoint O. The next thing we will do is to find the viewpoint and the effective focal length of the panoramic virtual camera. We assume that the sensors and all objects rest in a common planar surface (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 the effective focal length in the horizontal direction (i.e. the radius of the cylindrical image) can be expressed as

$$F_h = \frac{W_\theta}{2\pi} \tag{2-3}$$

where W_θ is the perimeter (in pixels) of the cylindrical image. The projection in the vertical direction is modeled as a linear perspective projection (after the distortion removal), so the effective "focal length" in the vertical direction can be estimated as (Fig. 2.5)

$$F_v = D\frac{v_1 - v_2}{H_1 - H_2} \tag{2-4}$$

where H_1 and H_2 are the heights of two points in a vertical pole measured from the floor, v_1 and v_2 are the vertical coordinates (with an arbitrary origin) of their image projections, and D is the distance of the pole from the camera (Fig. 2.4). By moving the same pole from distance D to distance D', we have a new projection v'_1 for the point H_1. Then the vertical coordinate (v_0) of the horizon circle and the height (H_0) of the virtual camera can be calculated by

$$v_0 = \frac{Dv_1 - D'v'_1}{D - D'}, \quad H_0 = H_1 - \frac{D(v_1 - v_0)}{F_v} \tag{2-5}$$

III. Panoramic Virtual Stereo Geometry

Assume we have two panoramic cameras with the same parameters. Both of them are subject to planar motion on the floor and are at the same heights above the floor. Suppose that in Fig. 3.1, O_1 and O_2 are the viewpoints of the two cameras and they can be localized by each other in the panoramic images as M_{21} and M_{12}. B is the baseline (i.e. distance O_1O_2) between them. The projection of a target point T is represented by T_1 and T_2 in the two panoramic images. Then a triangle O_1O_2T can be formed. By defining an arbitrary starting orientation for each cylindrical image, three angles ϕ_1, ϕ_2 (and ϕ_0) of the triangle can be calculated from the following four

bearing angles: θ_1 and θ_2, the bearings of the target in image 1 and image 2 respectively, β_{12} and β_{21}, the bearing angles of camera 1 in image 2, and camera 2 in image 1 respectively. Therefore the distances from the two cameras to the target can be calculated as

$$D_1 = B\frac{\sin\phi_2}{\sin\phi_0} = B\frac{\sin\phi_2}{\sin(\phi_1+\phi_2)}, \quad D_2 = B\frac{\sin\phi_1}{\sin\phi_0} = B\frac{\sin\phi_1}{\sin(\phi_1+\phi_2)} \tag{3-1}$$

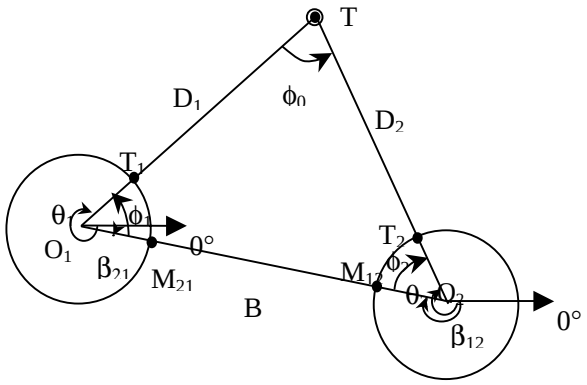


Fig. 3.1. Panoramic triangulation (top view)

A numerical analysis of the triangulation error will be given in section 3.2. In the extreme case, if the target is aligned with the baseline O_1O_2, the triangulation relation is invalid. However, we can still estimate the 3D location of the target by using the size-ratio of the target in two panoramic images

$$D_1 = B\frac{w_2\cos\phi_2}{w_1\cos\phi_1 + w_2\cos\phi_2}\cos\phi_1, \; D_2 = B\frac{w_1\cos\phi_1}{w_1\cos\phi_1 + w_2\cos\phi_2}\cos\phi_2 \tag{3-2}$$

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°. As an example, if the target lies between O_1 and O_2 (Fig. 3.2), the distances to them can be calculated as

$$D_1 = B\frac{w_2}{w_1+w_2}, \; D_2 = B\frac{w_1}{w_1+w_2} \tag{3-3}$$

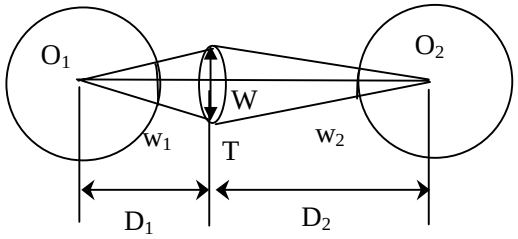
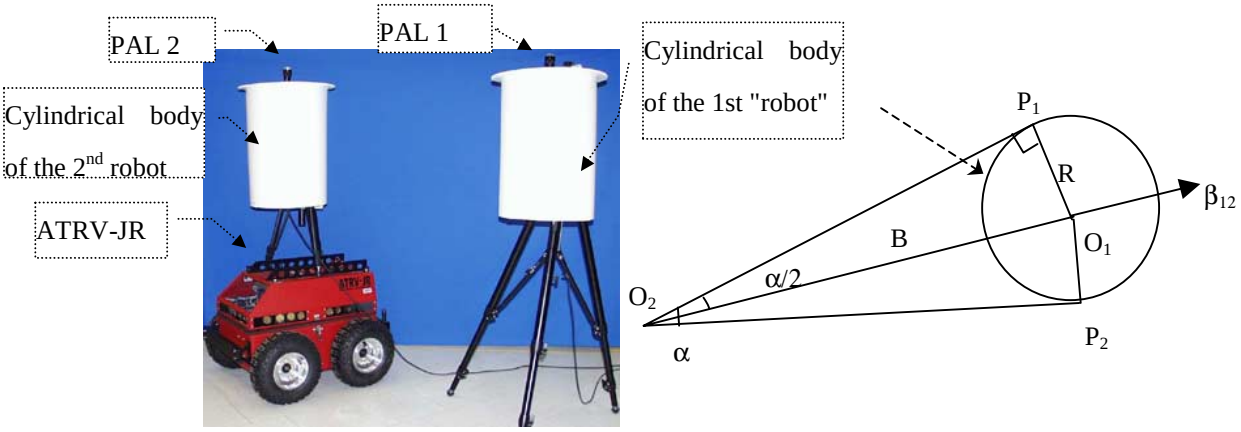


Fig. 3.2. Panoramic size-ratio method (top view)

Since the two cameras view the target (human) from exactly the opposite direction, the widths of the objects in the two images corresponds to *almost* the same width in 3D space (see Fig. 3.2, see also Fig. 3.6), which makes 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. However, it is only applicable when the top and/or bottom of the figure is visible in both of the panoramic images and can be accurately localized. In contrast, the width information is easier to extract and more robust since we can integrate the results from different heights of the object. Realizing that the object and the robots may occlude (part of) each other in the case of the alignment, we will use the width and height information adaptively.



(a) Setup (b) Geometry

Fig. 3.3. Find the orientation and the distance by a cylinder (top view)

3.1. Dynamic 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 by a dynamic calibration procedure. Several

practical approaches have been proposed for this purpose (Zhu99). The basic idea is to make the detection and calculation robust and simple. One of the approaches is to design the body of each robot as a cylinder with some vivid colors (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. 3.3a). We assume that the rotation axis of each panoramic camera is coincident with the rotation axis of the cylindrical body of the corresponding robot, therefore the baseline 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

$$B = R / \sin(\frac{\alpha}{2}) \qquad (3\text{-}4)$$

where α is the angle between two occluding projection rays measured in the image of camera 2, and R is the radius of the 1st cylindrical body (Fig. 3.3b). The orientation angle (β_{12}) of the line O_2O_1 is simply the average of the bearings of two occluding boundary points P_1 and P_2. We can do the same in the image of camera 1.

(image of the cylindrical body of the second robot)

(a) Pano 1: F_h =159.15 (pixels), α=11.52° (32 pixels), β_{21} = 23.76°, B = 180 cm

(image of the cylindrical body of the first robot)

(b) Pano 2: F_h =159.15 (pixels), α=11.52° (32 pixels), β_{21} = 227.88°, B = 180 cm

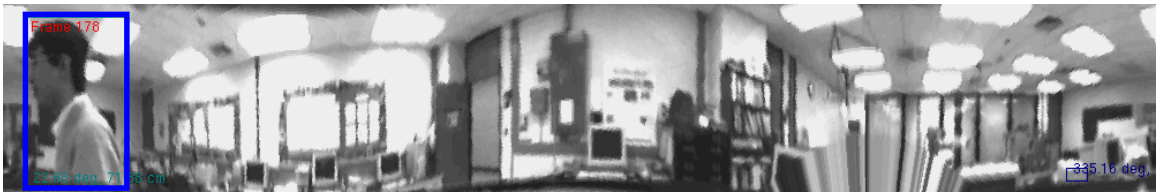
Fig. 3.4. Dynamic calibration by cylinders (which are pointed by arrows)

Fig. 3.4 shows a calibration result. The cylindrical body of each robot (pointing at by an arrow in Fig. 3.4(a) and (b)) is detected and measured in the panoramic image of the other robot. In the experiment, the perimeter of the cylindrical image is 1000 pixels, so the angular resolution in degrees is 360/1000 = 0.36 ° per pixel. We define the *angular resolution* of the panoramic image as χ in radians for future use, which is 6.28 mrad/pixel. The radius of the cylindrical body of each robot is R=18.0 cm.

(a) Pano 1: ϕ_1=48.60° (θ_1=72.36°, β_{21} = 23.76°), D_1 = 359 cm

(b) Pano 2: ϕ_2=92.52° (θ_2=135.36°, β_{12} = 227.88°), D_2 = 208 cm

Fig. 3.5. 3D estimation by triangulation



(a) Pano 1: ϕ_1=1.08° (θ_1=22.68°, β_{21} = 23.76°), D_1 = 72 cm

(b) Pano 2: ϕ_2=0.00° (θ_2=227.88°, β_{12} = 227.88°), D_2 = 108 cm

Fig. 3.6. 3D estimation by size-ratio method ($D_1 + D_2 = B$)

Fig. 3.5 and Fig. 3.6 show the experimental results of 3D estimation of a target using the above calibration result. In Fig 3.5, triangulation method is used, while in Fig. 3.6, the size-ratio method is applied. In both cases, the relative error is about 5% of the distance (D_1 or D_2), which was caused by the errors in dynamic calibration and the image match of two different views of the target (see Sec. 3.2). In the co-linearity case, the two cameras view the target (human) from the opposite direction, which makes the use of size (width) information reasonably good. Note that a 1-pixel error in the width of the robot cylinder in a panoramic image will introduce about 5-cm error in calculating the baseline in the configuration of Fig. 3.4. This error is a function of the resolution of the panoramic image and the size of the cylindrical robot (Eq. (3-6) in Sec. 3.2).

3.2. Error analysis

Eq. (3-1) and Eq.(3-2) show that the accuracy and resolution of distance estimation depends on the accuracy in estimating the baseline and the bearing angles. Here we derive an analysis of the error of estimating distance D_1 from the first camera to the target. The estimated triangulation error can be computed by partial differentials of Eq. (3-1) as

$$\partial D_1 = \left|\frac{\sin\phi_2}{\sin(\phi_1+\phi_2)}\right|\partial B + B\left|\frac{\sin\phi_2\cos(\phi_1+\phi_2)}{\sin^2(\phi_1+\phi_2)}\right|\partial\phi_1 + B\left|\frac{\sin\phi_1}{\sin^2(\phi_1+\phi_2)}\right|\partial\phi_2$$

or

$$\partial D_1 = \frac{D_1}{B}\partial B + D_1\left|\cot(\phi_1+\phi_2)\right|\partial\phi_1 + \frac{D_2}{\sin(\phi_1+\phi_2)}\partial\phi_2 \qquad (3\text{-}5)$$

where ∂B is the error in computing the baseline B, and $\partial\phi_1$ and $\partial\phi_2$ are the errors in estimating the angles ϕ_1 and ϕ_2 from the two panoramic images. Analyzing Eq. (3-5), we have found that the distance error comes from three separate error sources: calibration error, matching error and triangulation error, which will be discussed below.

3.2.1. Calibration error

Dynamic calibration estimates the baseline B, and the bearing angles β_{12} and β_{21} of the two cameras. The error in estimating the baseline by Eq. (3-4) can be derived as

$$\partial B = \frac{B\sqrt{B^2 - R^2}}{2R}\partial\alpha \leq \frac{B^2}{2R}\partial\alpha \qquad (3\text{-}6)$$

where R << B, and $\partial\alpha$ is the error in estimating the angle α in an image. From Eq. (3-6) we can find that the baseline error ∂B is inversely proportional to the dimension of the cylindrical body for dynamic calibration given the same angle error $\partial\alpha$. Given the radius R and the angle error, the baseline error is roughly proportional to the square of the baseline itself. The angle error ($\partial\alpha$) is determined by the errors in localizing the occluding boundaries of the second (or first) cylinder in the first (or second) panoramic image (Fig. 3-3). The errors in the bearing angles β_{21} and β_{12} will introduce errors to ϕ_1 and ϕ_2 (Fig. 3.1). Since each bear angle is the average of the orientations of the two occluding boundaries, they can be roughly modeled as the same as $\partial\alpha$, i.e. $\partial\beta_{21} = \partial\beta_{12} = \partial\alpha$.

3.2.2. Matching error

Assume that we want to find the distance of a *given* point T_1 in view 1 by finding its corresponding point T_2 in view 2. In this sense, there will be no error in providing the bearing angle θ_1 in view 1, which implies that the error $\partial\phi_1$ is solely determined by β_{21} via calibration, i.e. $\partial\phi_1 = \partial\alpha$. However, the view difference in O_1 and O_2 will introduce a "matching error" (denoted as $\partial\theta$) in θ_2, the localization of T_1's matching point T_2, which could be a function of the location of the view point O_2 (related to O_1). Thus, $\partial\phi_2 = \partial\alpha + \partial\theta$ is a (complicated) function of the viewpoint location and is generally larger than $\partial\phi_1$.

3.2.3. Triangulation error

Now we want to find a numerical result of the following problem: for a certain distance D_1 from camera 1 to the target, what is the error distribution for different locations of camera 2, which determines configurations of baselines and angles of the panoramic stereo? Since it is hard to give a numerical function of the error $\partial\phi_2$ vs. the location O_2, we will use the same measure error bounds in all the angles, i.e. $\partial\alpha = \partial\phi_1 = \partial\phi_2 \equiv \partial\phi$. Later on we will re-consider this matching error qualitatively. We decompose the analysis into two steps. First, by fixing the baseline, we find the optimal angle ϕ_1. It is equivalent to finding the optimal position of O_2 on a circle of origin O_1 and radius B (Fig. 3-7). Second, under the optimal angle configuration, we find the optimal baseline B. An additional consideration is that a human has a size comparable to the robots, so the distances between a robot and the target should be at least greater than the dimension of the robot, $2R$.

In the first step, we are trying to find the minimum value of the error due to the second and third terms of Eq. (3-5), i.e.

$$\partial D_1^{\phi} = D_1 \left|\cot(\phi_1 + \phi_2)\right| \partial\phi_1 + \frac{D_2}{\sin(\phi_1 + \phi_2)} \partial\phi_2 \qquad (3\text{-}7)$$

It is equivalent to find the optimal position of O_2 on a circle of origin O_1 and radius R. We first consider the case where $B < D_1$. In this case, Eq.(3-7) can be re-written as a function of ϕ_1

$$\partial D_1^{\phi} = \frac{B^2 + 2D_1^2 - 3BD_1 \cos\phi_1}{B \sin\phi_1} \partial\phi \qquad (3\text{-}8)$$

where we assume that the same measure errors in angles, i.e. $\partial\phi_1 = \partial\phi_2 = \partial\phi$. By some mathematical deductions, we can find that the minimum error can be achieved when $\cos\phi_1 = \frac{3BD_1}{2D_1^2 + B^2}$. The minimum error is

$$\partial D_1^{\phi}\big|_{\min} = \frac{\sqrt{(D_1^2 - B^2)(4D_1^2 - B^2)}}{B}\partial\phi < \frac{2D_1^2}{B}\partial\phi \qquad (3\text{-}9)$$

Th error in Eq. (3-8) increases from the minimum value to ∞ when the angle ϕ_1 changes from the optimal value to 0° and 180° respectively (Fig. 3.4a).

In the second step, we will find the optimal baseline in the case of optimal angle. Inserting Eq. (3-6) and Eq. (3-9) into Eq. (3-5) and assuming that the angle error $\partial\alpha$ in Eq. (3-6) also equals to $\partial\phi$, we have

$$\partial D_1 = (\frac{D_1\sqrt{B^2 - R^2}}{2R} + \frac{\sqrt{(D_1^2 - B^2)(4D_1^2 - B^2)}}{B})\partial\phi < D_1(\frac{B}{2R} + \frac{2D_1}{B})\partial\phi \qquad (3\text{-}9)$$

It is intuitive that the larger is the baseline, the better the triangulation will be (term 2 in Eq.(3-9), however the estimated error in the baseline is also larger (term 1). The minimum value can be achieved when $B = 2\sqrt{D_1 R}$, which means that

(1) more accurate baseline estimation can be obtained given a larger cooperative robotic target (i.e. R), hence the optimal baseline for estimating distance D_1 *can* be larger, and

(2) the farther the target is, the larger the baseline *should* be.

Assuming that the human object has a size comparable to the robots, the distances between a robot and the target should be at least greater than the dimension of the robot, *2R*. So Eq.(3-9) is only valid when $D_2 \geq 2R$, hence we should have $D_1 \geq B+2R$. Similarly, we can find the optimal solutions when $B = D_1$ and $B > D_1$.

In conclusion, we have the following results:

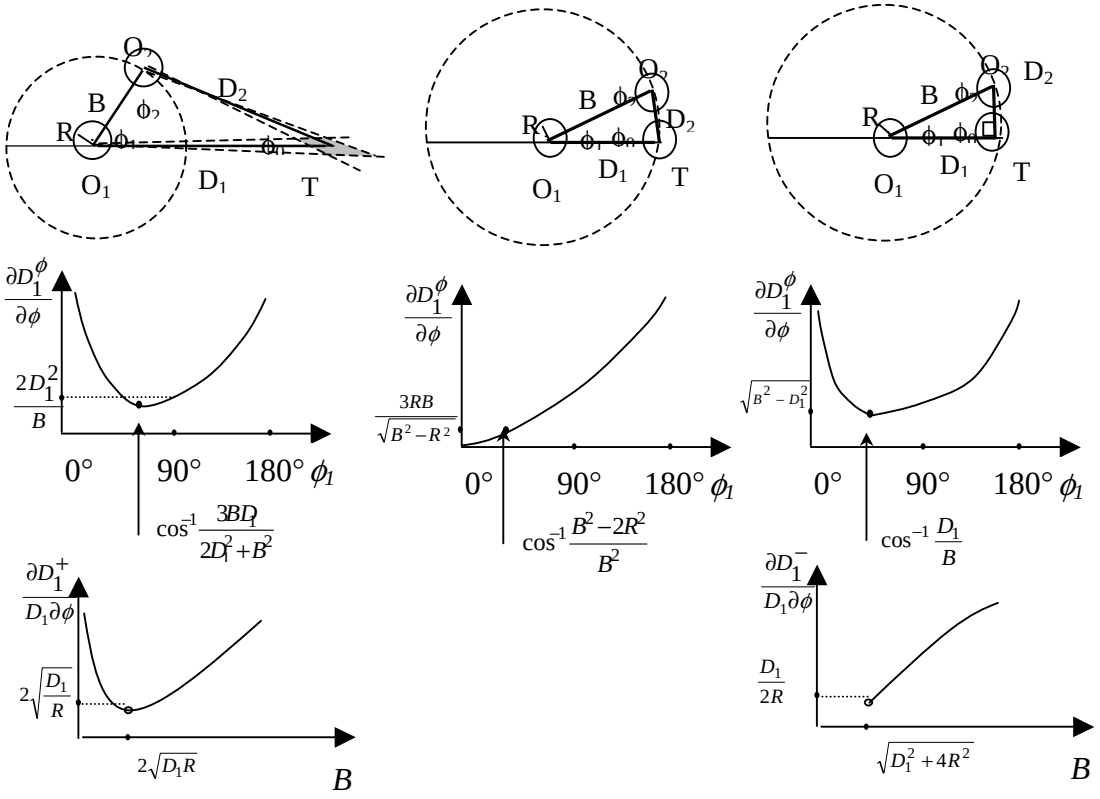
Case (1) . When $B \leq D_1\text{-}2R$, the best estimation can be achieved when

$$B = 2\sqrt{D_1 R}\,, \quad \cos\phi_1 = \frac{3BD_1}{2D_1^2 + B^2} \qquad (3\text{-}10)$$

and the error in the optimal configuration is

$$\partial D_1^+ = D_1\left(\frac{\sqrt{4D_1R - R^2}}{2R} + \frac{\sqrt{(D_1 - 4R)(D_1 - R)}}{\sqrt{D_1 R}}\right)\partial\phi < 2D_1\sqrt{\frac{D_1}{R}}\partial\phi \qquad (3\text{-}11)$$

Note that in this case, the minimum error is achieved when $\phi_1 < 90^\circ$, $\phi_2 > 90^\circ$ and $\phi_0 < 90^\circ$. For example, when R=0.18 m, D_1=4.0 m, $\partial\phi$ = 6.28 mrad (1 pixel), then we have B=1.70 m, $\phi_1 = 54.2^\circ$, $\partial D_1 / D_1$=5.4%.



(a). $B \le D_1$-$2R$ (b). $B \in (D_1$-$2R, \sqrt{D_1^2 + 4R^2})$ (c). $B \ge \sqrt{D_1^2 + 4R^2}$

Fig. 3.7. Best view angles and baselines

Case (2) . When $B \in (D_1\text{-}2R, \sqrt{D_1^2 + 4R^2})$, the best estimation can be achieved when

$$B = D_1, \ \cos\phi_1 = \frac{B^2 - 2R^2}{B^2} \tag{3-12}$$

and the error in the optimal configuration is

$$\partial D_1^0 = D_1\left(\frac{\sqrt{D_1^2 - R^2}}{2R} + \frac{3R}{\sqrt{D_1^2 - R^2}}\right)\partial\phi \tag{3-13}$$

Note that in this case, $\phi_1 < 90^\circ$ is the minimum angle by physical constraint of the minimum object distances, and $\phi_2 = \phi_0$.

Case (3) . When $B \geq \sqrt{D_1^2 + 4R^2}$, the best estimation can be achieved when

$$B = \sqrt{D_1^2 + 4R^2}, \ \cos\phi_1 = \frac{D_1}{B} \tag{3-14}$$

and the error in the optimal configuration is

$$\partial D_1^- = D_1\left(\frac{\sqrt{D_1^2 + 3R^2}}{2R} + \frac{2R}{D_1}\right)\partial\phi \tag{3-15}$$

Note that in this case, the minimum error is achieved when $\phi_1 < 90^\circ$, $\phi_2 < 90^\circ$ and $\phi_0 = 90^\circ$.

By some tedious mathematics comparing Eq. (3-11) and Eq. (3-15) under different D_1, we arrive at the following observation[+]:

[+] It can be also proved that we always have $\partial D_1^- < \partial D_1^0$, which means that it is better to set the baseline slightly greater than the distance D_1 when they have to be approximately equal. (In addition, the equality condition cannot be satisfied before we have an accurate estimation of D_1).

If the distance from camera 1 (the main camera) to the target is greater than 11.5 times the radius of the robot, i.e. D_1 > 11.5 R, we have $\partial D_1^+ < \partial D_1^-$, which means that the best configuration is $B = 2\sqrt{D_1 R}$, $\cos\phi_1 = \frac{3BD_1}{2D_1^2 + B^2}$ *(Eq. (3-10)). Otherwise, we have* $\partial D_1^+ >= \partial D_1^-$, *i.e. the best configuration is* $B = \sqrt{D_1^2 + 4R^2}$, $\cos\phi_1 = \frac{D_1}{B}$ *(Eq. (3-14)).*

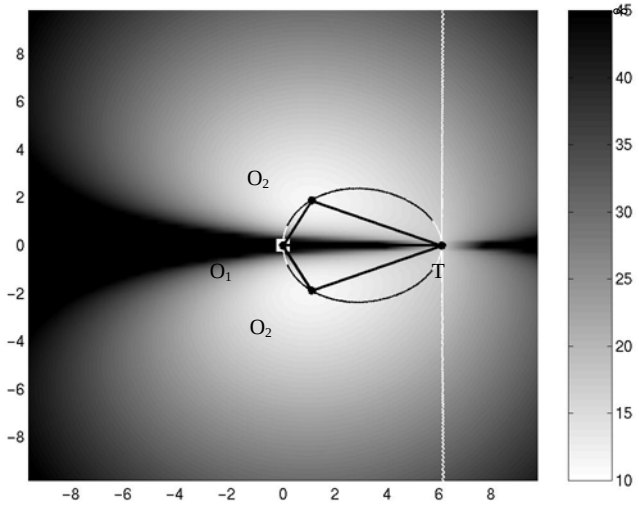


Fig. 3.8. Error map for distance D_1 when camera O_2 is in different locations of the map by fixing camera O_1 and the target T (D_1 = 34R = *6m*, R = 18 cm). The labels in the two axes are distances (in meters); the black-white curve shows where the minimum errors can be achieved for viewpoint O_2 on circles around O_1 *(see explanation in the text)*; the error value ($\partial D_1 / D_1 \partial\phi$) is encoded in intensity: see the corresponding bar.

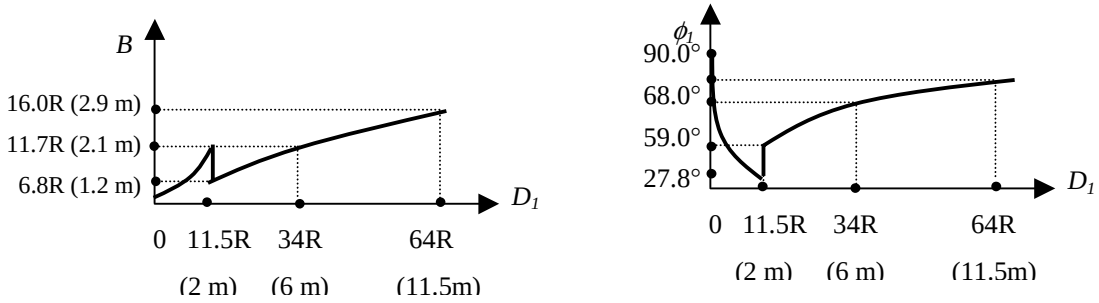
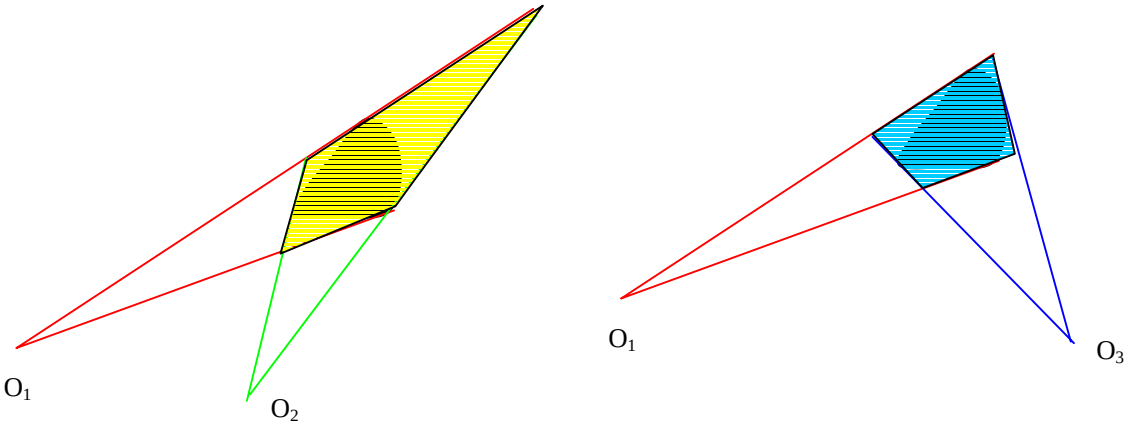


Fig. 3.9. Best baselines and angles vs. distance curves (The numbers in the parentheses are given when R = 0.18m)

3.3. View planning and further discussions

The above observation will be used in the optimal view planning. The distance error map under different viewpoint of camera O_2 is given in Fig. 8 in the case of D_1= 34R = 6m to verify the above conclusion. Minimum error is $\partial D_1 / D_1 \partial\phi$ =11.2 when B=220cm, $\phi_1 = 62.1^\circ$ (We have two such symmetric locations for O_2). The upper bound of the relative error is $\partial D_1 / D_1$ =7.0% when $\partial\phi$ is equivalent to 1 pixel. The selection of optimal viewing angle and baseline for different distance is shown in Fig. 3.9. Note that parameters in Fig. 3.9 are slightly different from those in Fig. 3.8 because the curves in Fig. 3.9 are drawn using Eq. (3-11) and Eq. (3-15) with some approximation and practical consideration. A comparison of the error between the flexible baseline and the fixed baseline, triangulation method and size-ratio method is given in Appendix 1, which shows that the flexible baseline triangulation method is almost always more accurate. The error analysis can also be used in the integration of the results from more than two such stationary sensors.



(1). small viewpoint difference (2) large viewpoint difference

Fig. 3.10. Viewing differences and distance/dimension estimation

The best triangulation configuration is derived when all the angular errors $(\partial\alpha, \partial\phi_1, \partial\phi_2)$ are treated as the same and independent to the view configuration of the panoramic stereo. However,

as we discussed in Section 3.2.2, the error $\partial\phi_2$ may be a function of the position of O_2 (given the locations of O_1 and T). A quantitative result can be derived in the same manner as above if the function is known or can be approximated; but here we only give a qualitative analysis. These error map in Fig. 7(d) also shows that there is a relatively large region with errors that are less than twice that of the minimum error. (In Fig. 7(d) it is the region around the black part of the minimum-error curve, which is a function of baseline B.) The large errors only occur when angle ϕ_0 is very close to 0° and 180°. It implies that a tradeoff can be made between the matching error due to large view difference and the triangulation error due to the small view difference. As we will do in Section IV, the match is between the centroids of the head of a human subject in two panoramic images, thus the 3D estimation gives the distance of a point near the center of the human target. In addition, it is interesting to note that larger view difference can give a better measurement of the dimension of the 3D object (person), which is similar to the volume intersection method (Fig. 3.10).

IV. Selected 3D Matching Approach

Since our primary goal is to detect and to track moving targets (humans) in 3D space, the primitives of the panoramic virtual stereo are objects (image blobs of human subjects) as a whole that have already been extracted from the two panoramic images (see Sec. 4.1). The triangulation approach basically needs the bearing angles of the objects in both images (as well as the orientations and distances of the two cooperative robots), whereas the size-ratio method also needs the width of each region as well as the bearing angles. This section will discuss how to reliably match object image blobs in two panoramic images subject to large perspective distortions due to rather different viewing angles. 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.

4.1. Moving object detection and extraction

A fast moving object extraction and tracking algorithm by using a stationary panoramic vision system has been developed (Zhu99). For completeness, we briefly describe the algorithm here. First, the look-up table (LUT) technique is used to map a circular image into a cylindrical image that ensures real time operation in moving object detection. The input of the moving object detection is a live video of cylindrical images.

Given a stationary camera, moving objects can be detected and extracted in a subtraction image $S_i(x,y)$, which is a subtraction of a current image by a background image. 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 a frame difference method with a background updating approach.

A frame difference $D_i(x,y)$ is calculated to detect any change between the current image and the previous image (assuming that there is very small inter-frame change in the background part). A connected- and/or nearby-region grouping algorithm is then used to find the *blob* - region and its contour of each possible object - in the subtraction image $S_i(x,y)$. Then we use this frame difference $D_i(x,y)$ to verify if a region extracted from the subtraction of the current frame from the background image is really a moving object instead of the dynamic change of the background. A set of moving object blobs is extracted by checking that enough pixels in the difference image have changed inside each region of a candidate subject. The object *blob image* is the image that copies the original intensity values from the current image only for those regions with moving objects.

An initial background is generated at the beginning of the object detection process by estimating the median value of each pixel in multiple frames (e.g., 30 frames). Then, during the detecting process, the background image is being updated pixel-wise by a weighted average of the existing background pixel and the current image *only in the non-object regions*.

(a) Cylindrical image, with bounding rectangles of moving objects superimposed

(b) Background image

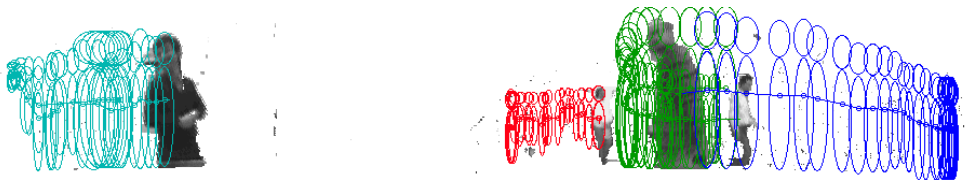
(c) Object image of three people

Fig. 4.1 Multiple object segmentation and virtual zooming

Fig. 4.1a shows one frame in a detection example where 3 people walked inside a room. Fig. 4.1b shows the current background image that had been generated for the first 24 frames when all the three peoples were walking around, and then was being updated for each processing frame, i.e. 5 times per second. Fig. 4.1c shows the object image where all the three people are almost completely extracted from the background and the boundaries of the people are smoothly along the contours of their bodies.

In the current implementation, multiple objects are tracked based on such features as size, aspect ratio and position of each object. Fig. 4.2 depicts multiple moving human object detection and tracking procedure. Multiple moving objects (4 people) were detected in real-time while moving around in the scene in an unconstrained manner; the panoramic sensor is stationary. Each of the four people was completely extracted from the complex background as depicted by the bounding rectangle, direction, and distance to each object. The dynamic track, represented as a small circle and icon (elliptic head and body) for the last 30 frames of each person is shown in Fig. 4.2b in different colors. The final object image is depicted at the end of the corresponding track. The frame rate for multiple object detection and tracking is about 5 Hz in a Pentium 300M Hz PC for 1080*162 panoramic images.

(a) Cylindrical images with bounding rectangles of moving objects superimposed



(b) Object tracks, each track is for the last 32 frames

Fig. 4.2 Multiple moving object tracking

4.2. 3D features based stereo match

Since the appearances of an object will vary significantly from largely separated views, the information only from 2D images will produce ambiguity in object match. Hence, we have explore two ways to improve the accuracy and robustness in blob matches across such widely separated views - head extraction and 3D measurements.

4.2.1. Head extraction and blob annotation

We have realized that the bearing of the centroid of an entire blob is subject to the effects of the positions of arms and legs, and the errors in body extraction. We have found that the bearing of the head of a human is more accurate than the entire blob of the human subject for three important reasons - it is usually visible in the panoramic images, it is almost symmetric, and it is easy to extract from the background (see Fig. 4.1- Fig. 4.3). The quasi-symmetry of a head makes it more suitable for matching across two widely separated views. This idea can be further extended by extracting different parts of a human blob for partial match between two views.

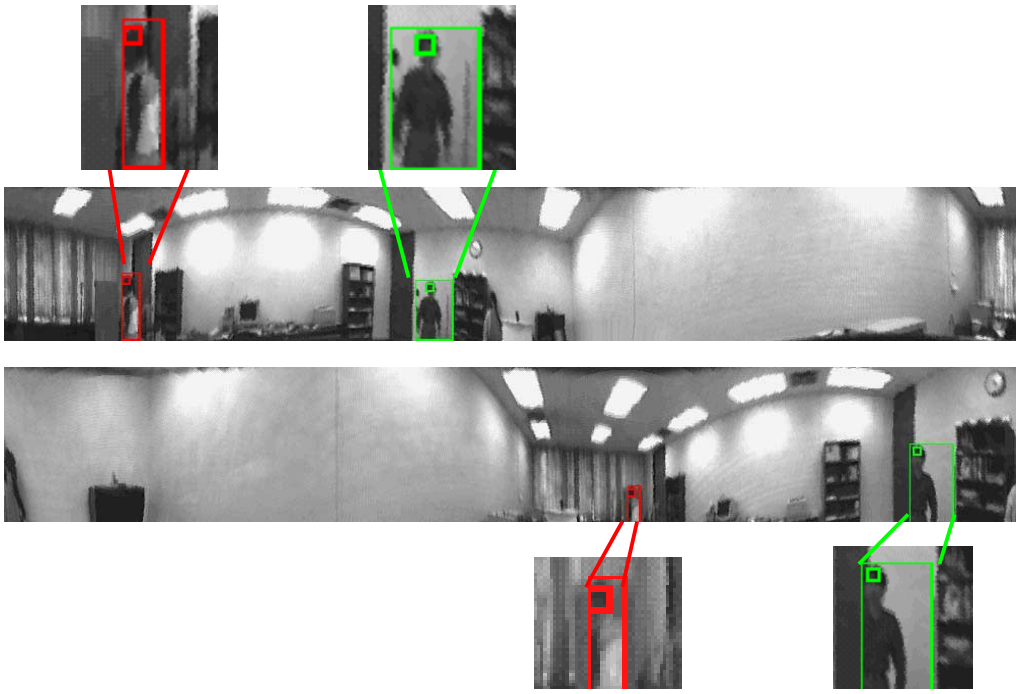


Fig. 4.3. Head extraction and bearing estimation. The large rectangle on each human subject is the bounding rectangle of each blob, and the small rectangle inside indicates the centroid of the extracted head.

The head part of a blob is extracted by using the knowledge that it is the topmost part of the blob and it has certain height-width ratio (e.g., 3:2) in a panoramic image. Here the exact height of the head part is not so critical since we only use the bearing angle of a head for triangulation. Fig. 4.3 shows results of human blobs and heads 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 blobs from two widely separated views. Notice that the centroid of each head region gives

correct bearing of the head even if the indicated height is not accurate and not consistent across the corresponding image pair. The second human subject in the images gives a good example showing that the bearing of the head is more accurate than the entire blob, which is a inaccurate detection of the human body due to the intensity closeness of the clothes and the door (left side) and the detected shadow on the wall (right side). The shadow is not so obvious to human eyes, but it was detected by the system.

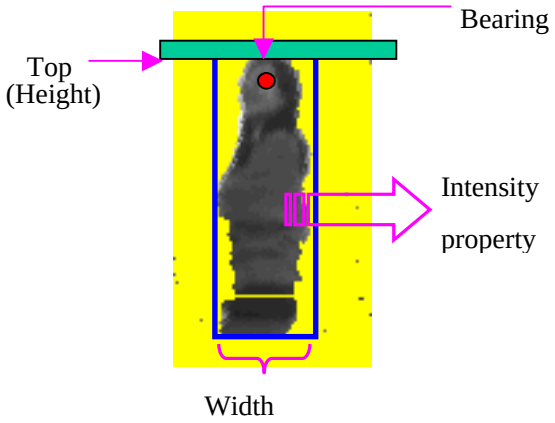


Fig. 4.4. Annotated human blob

From each panoramic image, a set of objects (blobs) is extracted, which is *annotated* by the following parameters (Fig. 4.4)

$$\mathbf{T}^{(k)} = \{T_i^{(k)} = (I_i^{(k)}, \theta_i^{(k)}, w_i^{(k)}, h_i^{(k)}), i = 1,..., N_k\} \qquad (4\text{-}1)$$

where k (1 or 2) is the number of cameras, $I_i^{(k)}, \theta_i^{(k)}, w_i^{(k)}, h_i^{(k)}$ are the photometric feature, the width of the image blob, bearing angle of the head of the target i in camera k, and the vertical coordinate of the top of the blob (indicating the height of a person).

4.2.2. 3D-related match measurements

Based on the above annotated blobs, we have explored the 3D-related measurements as well as 2D photometric features. For each object i in $T^{(1)}$, we try to match it with every object j in set $T^{(2)}$, and we derive the following measurements for each pair.

(1). ***Illuminant similarity*** - Assuming that illuminant feature is a scalar value (e.g. the average intensity of the region), then the following illuminant ratio can be derived

$$r_{is}(i,j) = \frac{\min(I_i^{(1)}, I_j^{(2)})}{\max(I_i^{(1)}, I_j^{(2)})} \in [0,1] \qquad (4\text{-}2)$$

In the current algorithm, we use a scale value - the median intensity of the image region of an object as the illuminant feature. More complex features can be used to calculate the illuminant ratio between two image regions.

(2). ***Ray convergence*** - A meaningful match must satisfy the condition that two rays from the viewpoints to the target images converge. The degree of ray convergence is calculated as

$$r_{rc}(i,j) = \begin{cases} 0: & \text{if rays } O_1T_i^{(1)} \text{ and } O_1T_j^{(2)} \text{ diverge} \\ \phi_0 & \phi_0 \le B / D_{\max} \\ 1 & \text{otherwise} \end{cases} \qquad (4\text{-}3)$$

where ϕ_0 is the angle between rays $O_1T_i^{(1)}$ and $O_1T_j^{(2)}$ (refer to Fig. 4.6) **,** D_{max} is the maximum distance that can be detected by the panoramic stereo system.

(3). ***Width consistency*** - By calculating the distances $D_i^{(1)}$ and $D_j^{(2)}$ of the hypothesized target to viewpoints O_1 and O_2 by matching $T_i^{(1)}$ and $T_j^{(2)}$ and using Eq. (3.1) or Eq. (3.2), the size of the hypothesized target estimated from two images should be very close for the correct match. A width ratio is calculated to account for this width consistency in two images

$$r_{wc}(i,j) = \frac{\min(w_i^{(1)} D_i^{(1)}, w_j^{(2)} D_j^{(2)})}{\max(w_i^{(1)} D_i^{(1)}, w_j^{(2)} D_j^{(2)})} \in [0,1] \qquad (4\text{-}4)$$

(4). ***Height consistency*** - First we calculate the height of the top of the hypothesized object from both images, as $H_i^{(1)}$ and $H_j^{(2)}$ (using Eq.(2-5)), which should be the same. Then a height ratio accounting for the degree of height consistency can be calculated as

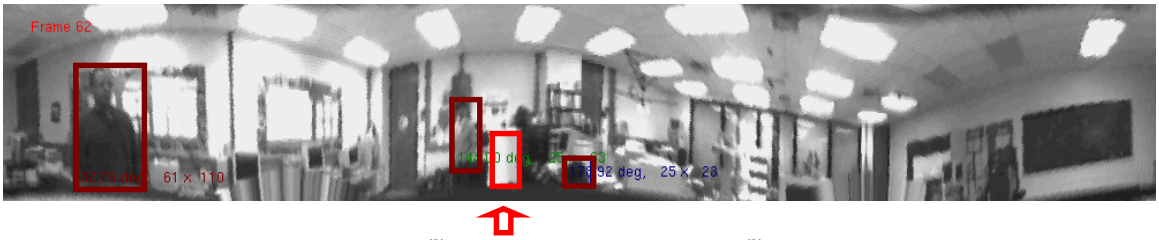
$$r_{hc}(i,j) = \frac{\min(H_i^{(1)}, H_j^{(2)})}{\max(H_i^{(1)}, H_j^{(2)})} \in [0,1] \qquad (4\text{-}5)$$

Therefore, a match measurement matrix $\mathbf{M} = [r(i,j)]_{N1xN2}$ can be constructed where the element indexed by (i,j) is the total "goodness" measurement of the match $i \leftrightarrow j$:

$$r(i,j) = r_{is}(i,j)r_{rc}(i,j)r_{wc}(i,j)r_{hc}(i,j) \in [0,1] \qquad (4\text{-}6)$$

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

(a) Pano1 - from left to right: $T_1^{(1)}$=(123,54,97,107.64,97), M_{21} = (213,24,47,313.56,47) $T_2^{(1)}$=(57, 30,78,338.40,88)



(b) Pano 2- from left to right: $T_1^{(2)}$=(57,61,110,32.76,117), $T_2^{(2)}$=(106, 25,63,144.00,86), M_{12} =(213,24,49,156.60,54), $T_3^{(2)}$=(49, 25,23,178.92,36)

Fig. 4.5. 3D estimation of multiple objects

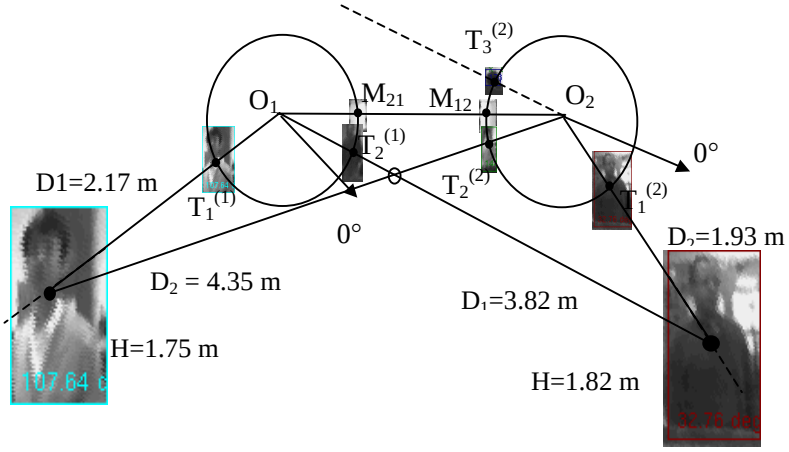


Fig. 4.6. Geometric relation of 2D images and 3D objects

	$T_1^{(2)}$	$T_2^{(2)}$	$T_3^{(2)}$
$T_1^{(1)}$	0.0 (0.46, 0.0, /, /)	**0.72** **(0.86,1.0,0.93,0.91)**	0.0 (0.40,0.0, /, /)
$T_2^{(1)}$	**0.988** **(1.0,1.0,0.99,0.99)**	0.32 (0.54,1.0,0.62,0.95)	0.0 (0.86,0.0, /, /)

Fig. 4.7. Match measurement matrix

4.2.4. Blob match algorithms

[Algorithm 1] A "greedy" match algorithm

Now the task is to find for each object in the first image the correct match in the second image based on the match matrix. The first algorithm is an exclusive "greedy" match algorithm:

Step 0: Initialization: m = 0; $\boldsymbol{M}^{(0)} = \boldsymbol{M}$*;*

Step 1: Find the maximum value in matrix $\boldsymbol{M}^{(m)}$*, and its corresponding indices (i, j). Then object i in the first image matches object j in the second image;*

Step 2. Delete row i and column j in the matrix $\boldsymbol{M}^{(m)}$ *to form a sub-matrix* $\boldsymbol{M}^{(m+1)}$ *; assign m = m+1;*

Step 3. Go to step 1 until no rows or columns left in $\boldsymbol{M}^{(m)}$.

This algorithm is very fast even if many 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 (see discussions in Section 5.2).

[Algorithm 2] A global optimal match algorithm

The second algorithm is an exhausted search of the global optimal and exclusive matches for all the possible match pairs. Assume that $N_1 \leq N_2$, a valid match set $\{T_{i_m}^{(1)} \leftrightarrow T_{j_n}^{(2)}\}$, or simply denoted as $\{i_m \leftrightarrow j_n\}$, is a 1-1 match set, and it satisfies the following conditions:

(1) $m = 1,\ldots,N_1$, $n = 1,\ldots,N_1$ ($N_1 (\le N_2)$ pairs of matches)

(2) $1 \le i_m \le N_1$ and $1 \le j_n \le N_2$ (i_m and j_n are the indexes of objects in the 1st and 2nd images respectively)

(3) $i_m \ne i_p$ if m ≠ p, and $j_n \ne j_q$ if n ≠ q (exclusive matches)

In all such valid match sets, we try to find the best one $\{ i_m^* \leftrightarrow j_n^* \}$ that maximizing the total goodness measurements of all the matches, i.e.

$$\sum_{m,n=1,\ldots,N_1} r(i_m^*, j_n^*) = \max \sum_{m,n=1,\ldots,N_1} r(i_m, j_n) \qquad (4\text{-}7)$$

The time complexity of the global optimal match is O(N!) where $N=max(N_1, N_2)$, which is much higher than that of the "greedy" match algorithm when the number of objects are large. For example, when N =10, O(N!) : O($\frac{1}{3}N^3$) is 10886, but when N =5, O(N!) : O($\frac{1}{3}N^3$) is only 3. So the second algorithm is preferable since better results can be achieved in a reasonable computation time when the object number is small (see results in Section 5.2).

Fig. 4.5 and Fig 4.6 show an example where two objects were detected in panoramic image 1 (Pano 1) and three objects were detected in Pano 2. Fig. 4.6 visually shows the geometrical relation between images and 3D objects. Fig. 4.7 shows the match measurement matrix $\{r(i, j) : [r_{is}, r_{rc}, r_{wc}, r_{hc}]\}$. 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. 4.6. In this experiment, the panoramic image parameters are F_h =159.15 pixels, F_v =258.0 pixels, v_0= 58.4 pixels, H_0=137.66 cm, β_{21} = 313.56°, β_{21} = 156.60°, and B = 239.41 cm.

V. Cooperative Strategy in the Real System

In the panoramic stereo vision approach, we face the same problems as in traditional motion stereo: dynamic calibration, feature detection, and matching. In our scenario, we are dealing with moving objects before 3D matching, which seems to add more difficulties. Fortunately, the following cooperative strategies can be explored between two robots (and their panoramic sensors) to ease these problems: monitor-explore working mode, mutual awareness, information sharing and view planning. In this section, we will first discuss these cooperative strategies, then we will briefly describe our experimental system.

5.1. Cooperative strategies

5.1.1. Monitor-explore mode

In the two-robot scenario of human searching, one of the robots is assigned as the "monitor" and the other as the "explorer". The role of the "monitor" is to monitor the movements in the environment, including the motion of the "explorer". One of the reasons that we have a "monitor" is that it is advantageous for a stationary camera (mounted on the "monitor") to detect and extract moving objects. On the other hand, the role of the "explorer" is to follow a moving object of interest and/or find a "better" viewpoint for constructing the virtual stereo geometry with the camera in the "monitor". However, the motion of the "explorer" introduce complications in detecting and extracting moving objects, so we assume that the explorer remains stationary in the beginning of an operational sequence in order to easily pick up any moving objects. Then a tracking mechanism is activated as soon as it begins to move (the tracking procedure may integrate the motion and texture information, which need future work). We also expect that the explorer will stay still in an advantageous location after it has found a good viewpoint for 3D estimation. The role of the "monitor" and the "explorer" can and will be exchanged during mission execution.

5.1.2. Mutual awareness and information sharing

Mutual awareness of the two robots is important for their dynamic calibration of relative orientations and the distance between the two panoramic cameras. In the current implementation, we have designed a cylindrical body with known radius and color so it is easy for the cooperating robots to detect each other. It is interesting to note that while the motion of the explorer encounters difficulty in tracking other moving objects by itself, it is helpful for the awareness of its existence by the monitor. It is also possible to directly use more complicated but known *natural appearances* and geometrical models of a pair of robots to implement the mutual awareness and dynamic calibration.

The two panoramic imaging sensors have almost identical geometric and photometric properties. Thus it is possible to share information between them about the targets as well as themselves in the scene. For example, when a certain number of moving objects are detected and extracted by the stationary "monitor", it can pass the information of the object number, geometric and photometric features of each object to the explorer who may be in motion. Thus it makes the explorer easier to track the same objects. Information sharing is especially useful for the mutual detection of "cooperative" calibration targets since the models of the robots are already known *a priori*. In our simplified case, the cylindrical bodies of both robots always have the same appearances from any viewing angles. Therefore, whenever the "monitor" has detected the cylindrical body of the moving "explorer", it can estimate the bearing and distance of the robot "explorer". Then this piece of information is passed to the "explorer" so that the "explorer" can try to search for the cylindrical body of the "monitor" in its image with a good prediction of size and color under the current configuration and illumination condition.

5.1.3. View planning

View planning is applied whenever there are difficulties in object detection and 3D estimation. In our case, we define the view planning as the process of adjusting the view point of the exploring camera so that best view angles and baseline can be achieved for the monitoring camera to

estimate the distance to the target of interest. Occlusion of the human or the robot may occur when an object (either a human or a robot) is between the observing camera and the target. It is also the configuration when triangulation has larger error (of course the size-ratio method can be used in that situation for an initial estimation). According to the analysis in section 3.2, the guidelines for "best" viewing planning are as follows:

(1) Observation rule **-** This rule is applied when the two robots "observe" the target from a distance. *If the initial estimated distance from the viewpoint O_1 to the target D_1 is greater than 11.5R, the "explorer" should move as close as possible to an optimal position that satisfies the minimum distance error conditions, i.e., baseline constraint* $B = 2\sqrt{RD_1}$ *and the viewing angle constraint* $\cos\phi_1 = \frac{3BD_1}{2D_1^2 + B^2}$.

(2). Approaching rule **-** This rule is applied when the two robots are close to the target and the explorer is trying to "approach" the target. *If the estimated distance is smaller than 11.5R, the "explorer" should move as close as possible to an optimal position that satisfies the baseline constraint* $B = \sqrt{D_1^2 + 4R^2}$ *and the viewing angle constraint* $\cos\phi_1 = \frac{D_1}{B}$.

(3). Mutual-awareness rule **-** *Given the angular resolution of the panoramic image, χ, the size of the cylindrical robot body, R, and minimum detectable pixel number of the robots, w, the maximum distance of the baseline is $B = 2R/w\chi$ where two panoramic cameras are aware the existence of each other.* For example, assume that *w*=10 pixels is the minimum detectable width, then the maximum baseline is *B*=2.8 m given R=0.18 m and χ = 6.28 mrad/pixel.

(4). Navigation rule **-** *View planning strategy should also consider the cost of moving in finding a navigable path to the selected position. This cost is also a function of distance, straightness (degree of turns) and time to travel.*

Note that the explorer is always trying to find a best position in the presence of a target's motion.

These strategies can be extended to more than two cooperative robots, and in fact more than two robots will make the work much easier. For example, we can keep two of the three robots in a team stationary so that they can easily detect the moving objects in the scene, including the third robot in motion. Thus the locations of all the moving objects can be estimated from the pair of stationary panoramic cameras. Then, for a target in interest, we can find (dynamically) the best viewpoint for the third robot in order to estimate its distance from either of the two stationary robots. By using the knowledge of the (dynamic) locations of the target, other moving objects and the three robots, a navigable path for the third robot can be planned to the desirable goal. These measurements can also facility the detection of the target and the two stationary robots by the moving robots, for example, by tracking the objects with visual features inherited from the other tow robots. Thus the cooperative triangulation can be constructed between the moving and the stationary platforms.

5.2. Experimental System and Results

In our experimental system, we mounted one panoramic annual lens (PAL) camera on an RWI ATRV-Jr. robot (the "explorer"), and the other PAL camera on a tripod (the "monitor")(Fig. 3.3a). Two Matrox-Meteor frame grabbers were installed on the ATRV-JR and a desktop PC respectively, both with 333M Hz PII processors. The communication between two platforms is through sockets over an Ethernet link (wireless Ethernet communication will be used in the future system). 3D moving object detection and estimation programs run separately on the two machines at about 5 Hz. Only camera and object parameter data (i.e., baseline, bearing angles, sizes, and illuminate features) were transmitted between two platforms so the delay in communication can be ignored at the current processing rate (5Hz). In the current implementation, we assume that the most recent results from both platforms correspond to the events at same time instant. Synchronized image capture is being considered by using the 30-frame buffering capability of the frame grabbers and by data interpolation in order to avoid the time/motion delay of moving objects in two images.

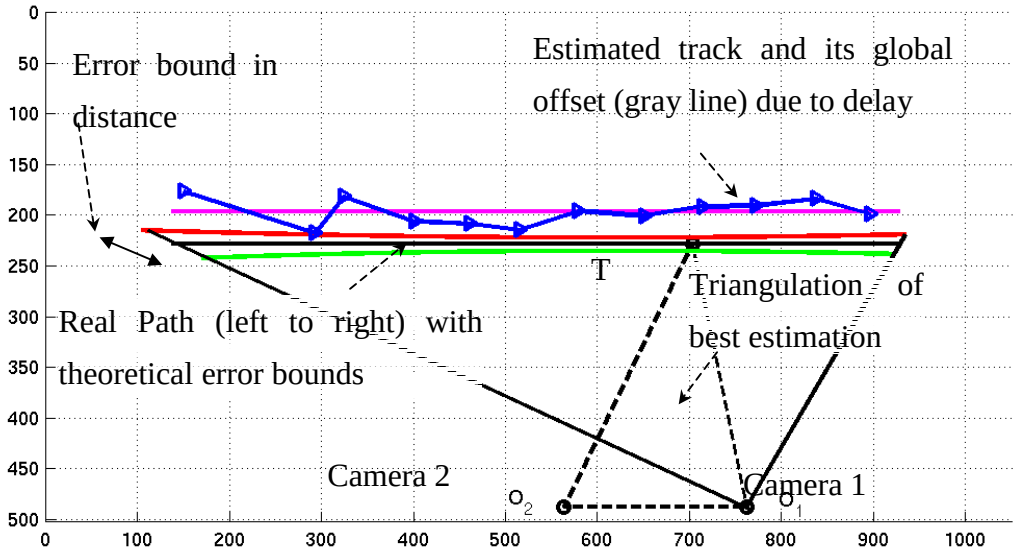


Fig. 5-1. Panoramic stereo tracking result (axis in cm)

Fig. 5-1 shows the result from an experiment to evaluate the panoramic stereo's performance of tracking a single person walking along a known path when the two cameras were stationary. The theoretical error bounds were computed assuming that all the angular errors in Eq. (3-5) and Eq. (3-6) were equivalent to 1 pixel. The target (T) position where the theoretical best triangulation on this track can be expected is shown in the figure, which is validated by the real experimental result. Even if the localization errors in images may be larger than 1 pixel, the average error of the estimated track is comparable to the theoretical error bounds, taking a global offset into account (The offset of the estimated track from the real path is due to the delay of the processing of the explorer (O_2) working in "telnet" mode).

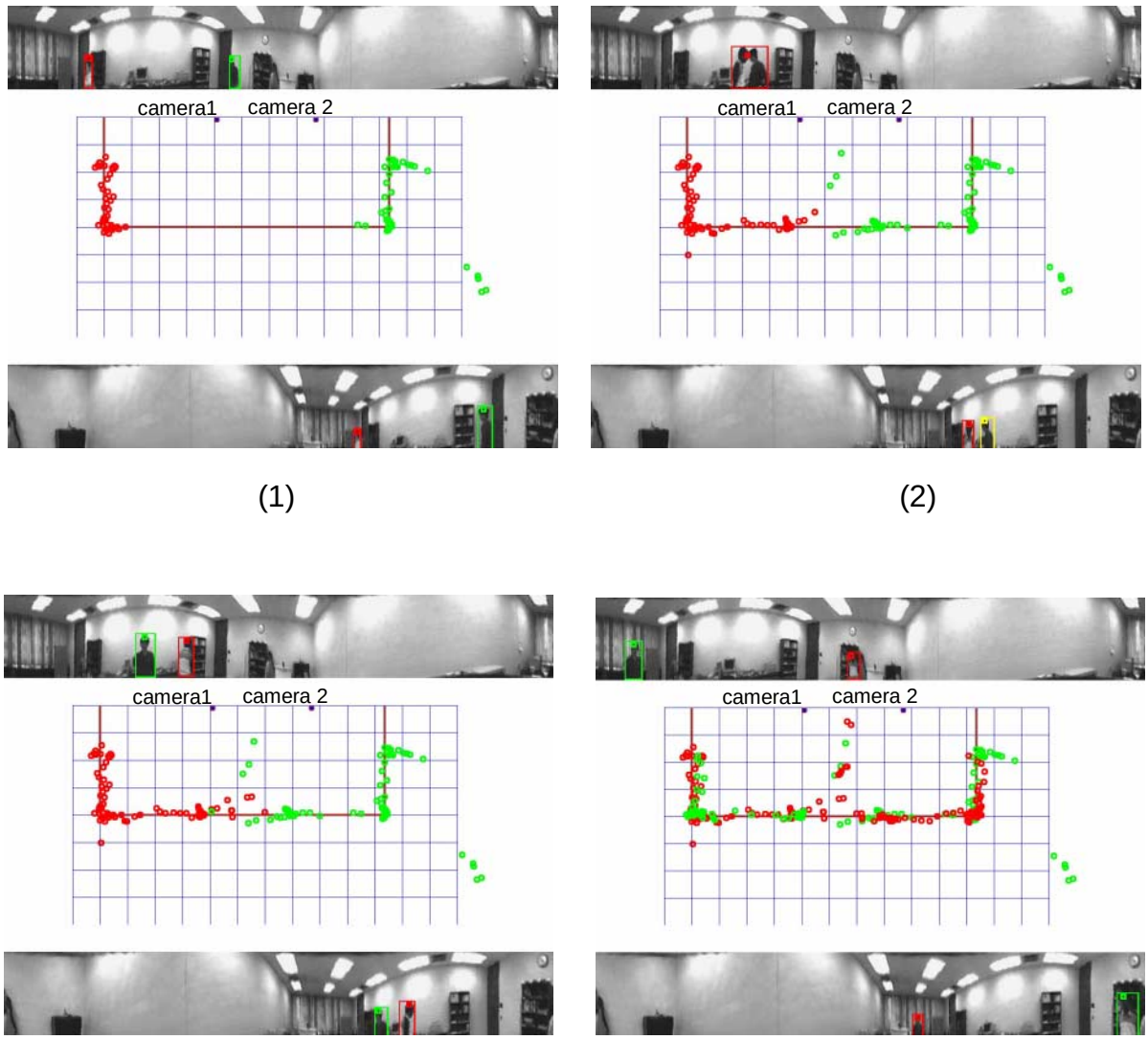
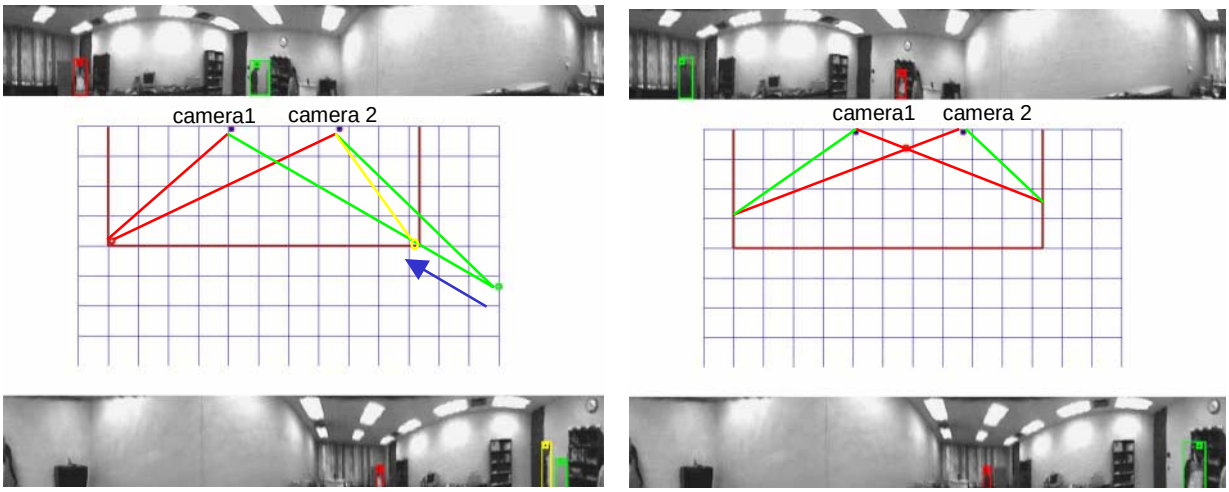


Fig. 5-2. Panoramic stereo tracking two people. The four pictures show localizing and tracking results before they met, when they met, after they departed and when they arrived their goals, out of 214*2 localization results. In each picture, the top and bottom images are the panoramic image pair from two panoramic cameras. Each image is actually the corresponding background with the superimposed blob images and their annotations of the blobs. In the center is the top view of the room where each grid is 50x50 cm^2. Each red or green dot represents a location of the corresponding person.



Match "goodness"

Image 1 \ Image 2	1	2	3
1	0.37	0.43	0.62
2	0.00	0.00	1.00

Match "Goodness"

Image 1 \ Image 2	1	2
1	0.00	0.92
2	0.89	1.00

(1) **(2)**

Fig. 5-3. Failure examples in "greedy" match algorithm. (1) The second blob (green, human +shadow) in the 1st image mis-matched with the shadow (green) of the second human (yellow) (2). Mis-match when two people met. For both (1) and (2): row 1 – the first image, row 2- top view of the floor, row 3- the second image, and row 4 – match "goodness" measurements. Note that the measurements are normalized to 0~1 for easy judgments.

Fig. 5.2. shows the results of detecting and tracking two people who walked from the 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 each picture in Fig. 5.2), the red dot sequence shows the path of one person, and the green dot sequence shows that of the other. It can be seen that the proposed 3D match, localization and tracking algorithms produced good results. The average localization error is about 20 cm. There are less than 5% mis-matches, which

happened in two places. One place is when the shadow of a person was projected on the wall and was detected and mis-matched by the system (Fig. 5-3(1)). In the match "goodness" table, green-green match is only slightly "better" than green –yellow match. This problem could be solved by a partial match method- parts of the (green) blob in the 1st image should match the two blobs (green and yellow) in the second image. The second place is when the two people met (Fig. 5-3(2)). It happened in a place where one person (red) is very small in the first image, while the other person (also marked as red) is small in the second image. These two small image blobs constructed a good 3D configuration and unfortunately the images are too small to give any useful intensity information (The green pair is a invalid 3D configuration – see the match "goodness" table). This failure could be fixed by a finding matches that maximize the overall match "goodness" measures. For the example in Fig. 5-3(2), global optimal matches will be green-red and red-green since this valid match set gives the maximum global match "goodness" measure (0.92+0.89 instead of 0.00+1.00 of the "greedy algorithm). Fig. 5.4 shows the preliminary results of using the global match algorithm to the two-person tracking example. Better localization results can be obtained (please compare the tracks of Fig. 5.4 with that of Fig. 5.2(4)). For example, some mis-matches (including the one show in Fig. 5.3(2)) have been corrected. Further improvements and experiments on stereo match, view planning and evaluation are being undertaken.

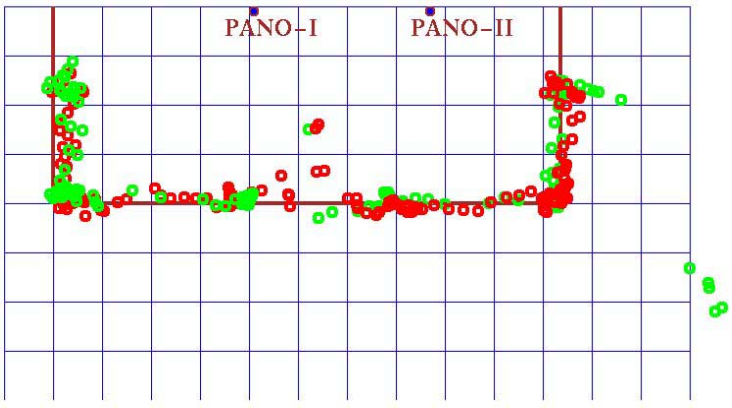


Fig. 5-4. Preliminary results using the global match algorithm.

VI. Conclusion and Discussion

This paper has presented a panoramic virtual stereo approach for two cooperative mobile platforms. There are four key features in our approach: (1) omni-directional stereo vision with single viewpoint geometry and a simple camera calibration method, (2) cooperative mobile platforms for mutual dynamic calibration and best view planning, (3) 3D matching after object extraction and (4) near 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, occlusions and motion detection of movable robotic platforms. Experiments have shown that this approach is encouraging. On-going and future work include the following topics:

(1) ***Improvement of the calibration accuracy*** - By integrating a panoramic camera with a pan/tilt/zoom camera, the system can increase the capability in both viewing angle and image resolution to detect the cooperative robots as well as the targets. The robust and accurate dynamic calibration is the key issue in the cooperative stereo vision.

(2) ***Improvement of 3D matching*** - By using the contours of object images and more sophisticated features (color, texture, etc), more accurate results can be expected. This is another main factor that affects the robustness and accuracy of the 3D estimation.

(3) ***Tracking of 3D moving objects*** - We need to develop sophisticated algorithms to track the moving objects in both 2D images and 3D spaces, in the presence of occlusion, and by moving cameras as well as stationary cameras.

References

1. Aloimonos, Y. (ed.), "Active perception - Advances in Computer Vision," Lawrence Erlbaum Associates, Hillsdale, NJ, 1993.
2. Baker, S. and S. K. Nayar, A theory of catadioptric image formation, In Proceedings of the 6th International Conference on Computer Vision, IEEE, India, January 1998.
3. Boult, T., E., R. Micheals, X. Gao, P. Lewis, C. Power, W. Yin, A. Erkan, Frame-Rate omnidirectional surveillance and tracking of camouflaged and occluded targets, Second IEEE Workshop on Visual Surveillance, June 1999: 48-58.
4. Brill, F. Z., T. J. Olson and C. Tserng, Event Recognition and Reliability Improvements for the Autonomous Video Surveillance Systems, In Proceedings of DARPA Image Understanding Workshop, volume 1, pages 267- 284, November 1998.
5. Geyer, C. and K. Daniilidis, "Catadioptric Camera calibration", In *Proc. Int. Conf. on Computer Vision*, Kerkyra, Greece, Sep. 22-25, pp. 398-404, 1999.
6. DARPA Image Understanding Workshop Proceedings, VSAM- Video Surveillance and Monitoring Session, Monterey, November 1998
7. Greguss, P., Panoramic imaging block for three-dimensional space, U.S. Patent 4,566,763 (28 Jan, 1986)
8. Greguss, P., PAL lens (email). Personal communication, 2000.
9. Haritaoglu, I., D. Harwood and L. Davis, W4S: A Real-time System for Detection and Tracking People in 2.5D, In Proceedings of ECCV, 1998.
10. Ishiguro, H., M. Yamamoto and S. Tsuji, Omni-directional Stereo, IEEE Trans. PAMI, Vol. 14, No.2, 1992: 257-262
11. Konolige, K. G., R. C. Bolles, Extra set of eyes, In Proceedings of DARPA Image Understanding Workshop, volume 1, pages 25- 32, November 1998.

12. Lipton, A. J., H. Fujiyoshi, R. S. Patil, Moving Target Classification and Tracking from real-time Video, In Proceedings of DARPA Image Understanding Workshop, volume 1, pages 129- 136, November 1998.

13. Nayar, S. K., S. Baker, Catadioptric image formation, Prof. DARPA Image Understanding Workshop, May 1997:1431-1437

14. Nelwa, V., A true omnidirectional viewer, Technical Report, Bell Lab, Holmdel, NJ, Feb, 1996

15. Ng, K. C., H. Ishiguro, M. Trivedi and T. Sogo, Monitoring dynamically changing environments by ubiquitous vision system, Proc. Workshop on Visual Surveillance, June 1999: 67-73

16. Papageorogiou, C., T. Evgeniou, and T. Poggio, A Trainable Object Detection System, In Proceedings of DARPA Image Understanding Workshop, volume 2, pages 1019-1024, November 1998.

17. Pentland, A., A. Azarbayjani, N. Oliver and M. Brand, Real-time 3-D Tracking and Classification of Human Behavior, In Proceedings of DARPA Image Understanding Workshop, volume 1, pages 193-200, May 1997.

18. Powell, I., Panoramic lens, Applied Optics, vol. 33, no 31, Nov 1994:7356-7361

19. Yagi, Y., S. Kawato, Panoramic scene analysis with conic projection, Prof. IROS, 1990

20. Yamazawa, K, Y. Yagi and M. Yachida, Omnidirectional imaging with hyperboloidal projections, Prof. IROS, 1993.

21. Zhu, Z., E. M. Riseman, A. R. Hanson, Geometrical modeling and real-time vision applications of panoramic annular lens (PAL) camera, *Technical Report TR #99-11*, Computer Science Department, University of Massachusetts Amherst, February, 1999.

22. Zhu, Z, S. Yang, G. Xu, X. Lin, D. Shi, Fast road classification and orientation estimation using omni-view images and neural networks, *IEEE Trans Image Processing*, Vol 7, No 8, August 1998: pp. 182-1197.

Appendix 1. Comparison Analysis

Comparison between fixed baseline and flexible baseline **-** Assume that in a fixed baseline stereo system of a robot, two cameras are mounted as far away as possible in a robot with 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. *B* is accurate). Since we always have B < D_1 in fixed-baseline stereo, we can use Eq. (3-9) to estimate the distance error, i.e.

$$\partial D_1^{fix}\big|_{B=2R} \approx \frac{D_1^2}{R} \qquad \text{(a-1)}$$

Comparing Eq. (a-1) with Eq. (3-11), we have the conclusion that $\partial D_1^{+}\big|_{B=2\sqrt{D_1 R}} < \partial D_1^{fix}\big|_{B=2R}$ when $D_1 > 4R$, which is almost always true.

Comparison between triangulation and size-ratio approach **-** The error for the size-ratio method can be calculated in a similar way, For example, the distance error for Eq. (3-3) is

$$\partial D_1 = \frac{1}{B}\partial B + \frac{D_1}{w_1 + w_2}\partial w_1 + \frac{B - D_1}{w_1 + w_2}\partial w_2 \qquad \text{(a-2)}$$

or

$$\partial D_1 = D_1\left(\frac{\sqrt{B^2 - R^2}}{2R} + \frac{B - D_1}{W}\right)\partial w \qquad \text{(a-3)}$$

where W is the width of the target, and we assume that $\partial w_1 = \partial w_2 = \partial\alpha = \partial w$ (w is measured in radian). Obviously, we have $B>D_1$, $D_1>2R$ and $D_2>2R$. Eq. (a-3) implies that a larger target means better distance estimation. The minimum error is obtained when the second camera O_2

moves as close as possible to the target, i.e. $D_2 = 2R$ (or $B = D_1 + 2R$). So the minimum error can be expressed by

$$\partial D_1^s = D_1\left(\frac{\sqrt{(D_1+R)(D_1+3R)}}{2R} + \frac{2R}{W}\right)\partial w \qquad \text{(a-4)}$$

We always have $\partial D_1^s > \partial D_1^-$ given that $B > D_1$, , $\partial w = \partial \phi$ and $W << D_1$.