SEGMENT-BASED MATCHING FOR VISUAL NAVIGATION[1]

Zhongfei Zhang
Richard Weiss
Edward M. Riseman
-
Computer and Information Science Department
University of Massachusetts
Amherst, MA 01003
Phone : (413)545-0528
NetAd : zzhang@cs.umass.EDU

May 15, 1994

Abstract

The imaging system involves an image produced from a reflection by a spherical mirror to produce a 360° projection of the environment. This paper extends previous work on an image-based navigation system in which the 360° view is compressed into a circular waveform. A navigation task is specified as a sequence of homing tasks on target locations. This is accomplished by matching landmarks from the current view with the next target view. This paper provides a new method for matching using qualitative geometric features of each of the waveforms. In addition, a geometric analysis allows us to do 3D reasoning about the environment, which is sufficient for computing the rotation and translation between the current location and the target location. It may eventually allow acquisition of a 3D model of the environment. We show that the system performs reliably in an indoor environment.

[1]This work was supported in part by the DARPA and Army ETL under contract DACA76-89-C-0017 and in part by NSF under grant DCR-8500332

1. INTRODUCTION

The problem of mobile robot navigation has many approaches including methods of motion analysis[1], associative homing[2], and model-based navigation[3]. This paper extends the work of Hong, Tan, *et al* [4] on navigation through image-based local homing. Homing is a navigation task in which the goal is one of a fixed set of target locations known to the robot. Unlike most homing systems, the navigation problem is treated here as a sequence of homing tasks. In our previous work, the environment was modeled as a set of snapshots of the world taken at target locations. A spherical mirror was used to project a full $360°$ view of the world onto a single image, which was then condensed into a compact, one-dimensional *location signature*. The system used local correlation to match between location signatures. That was accomplished under the assumption that there was no rotation and the step distance (i.e., the distance between two adjacent target steps) was small. The system worked well when this condition was satisfied. [4] showed an experiment in which the robot successfully moved 17 steps along a hallway.

The current work uses the same framework except that the location signature is represented symbolically as a sequence of segments, where each segment is one of three types: increasing, decreasing, and constant. The matching is done first qualitatively using symbolic sequences of the same shape, (i.e., a sequence of identical segment types) then hypothesized matches are verified quantitatively. In addition, a richer model of the environment is constructed including metric information, so that the rotation and translation between the current location and the target location can be computed, as well as distance to visible 3D landmarks. Since general motion (i.e., a combined motion with rotation and translation) can be modeled, there is no restriction on rotation and a longer step distance can be used for faster convergence. Moreover, the error after the robot has homed to each target is not accumulated in the navigation process provided the final pose in each homing stage is a valid initial pose of the next homing stage.

Other work on homing includes a system built by Zipser[2] in which every current view is compared with every stored view; then an averaged motion vector, weighted by the degree-of-match to each of its corresponding views, is computed to guide the homing process. Nelson[5] developed a similar system for homing but differed in that it computed the motion associated with the view that best matched its current view. Both methods are classified as associative homing, because the actions are associated or derived in some manner from the stored set of patterns. Biological systems utilize associative processing and it is a natural paradigm for parallel hardware.

Fennema *et al* [3] use three-dimensional models to generate projections of landmarks expected to be seen from the estimated current location; the robot then servoes directly on the image features, tracking them via correlation. This kind of 3D model-based navigation has the advantage that it can use precise information about the environment to guide the robot. Beveridge[6] and Kumar[7] in related work use 2D matches of projected landmarks to update the 3D location of the robot. Mechanisms have been developed to detect incorrect matches (i.e., outliers) so that the system can remain robust in real scenes.

Line tracking techniques have been applied to robot navigation. Ayache and Faugeras[10] developed a scheme by using an extended Kalman filter to build visual maps involved in a sequence of images. While this is an important research direction, there are significant difficulties in accurately recovering motion parameters and structure from motion[11, 12]. Dickmanns and Graefe[8, 9] also applieded Kalman filters to developing a technique for using image features in a real-time feedback control loop to control the motion of a vehicle. They applied this technique to successfully drive a vehicle at high speeds on the autobahn. The difference between their system and the one used by Fennema *et al*[3] is that the former accomplishes servoing by tracking image features based on an implicit model of the road whereas in the latter system, the tracked features are constructed from landmarks which have been selected from an explict 3D model. The former system is much more flexible but the latter system requires much less 3D specific information.

Recently, several omnidirectional (i.e., 360° view) image-based navigation techniques have been developed in order to reduce the difficulties in 2D image-based correspondence. An omnidirectional image has the potential advantage in that landmarks do not disappear because of the orientation of the camera or the vehicle. Zheng and Tsuji[13] presented an approach to landmark-based navigation in which they use a rotating slit scanner to produce a 360° panoramic view. Yagi and Kawato[14] developed a similar imaging system to ours except that they used a conical mirror instead of a spherical one, and are using the system to attempt reconstruction of a full 3D model of environment. Moreover, they did not compress the image so that their system could only be used in an environment rich in vertical landmarks.

In this paper, we continue our previous work on navigation[4] by presenting a more robust matching algorithm based on the waveform of signatures and incorporating 3D geometrical reasoning into the system.

2. SYSTEM OVERVIEW

The physical set-up of the navigation system presented in this paper is shown in Fig. 2(a) and has been described in [4], with the exception that now the camera, together with the spherical mirror, is put on top of the rotation platform of the robot instead of originally being put aside the platform. The camera sits directly below the spherical mirror so that a 360° image of the surrounding environment is acquired. Given planar motion, (i.e., translation in a plane and rotation about an axis normal to the plane), then there is a circle in the image which is invariant under this group of motion transformations. This circle is the "horizon circle" and is sampled from each image to form a 1D signal called the *location signature.* Fig.2(b) shows an image taken under this system. The white ring at the center of the image indicates the rotational center of the robot, which is also the origin of the 3D coordinate system (see the next section). The coordinate axes in the image plane, which actually are the X and Y axes in our 3D coordinate system, are indicated in the image. Together with the axes, a circle band composed of 360 ticks is also indicated. This circle band is the horizon circle. Each tick is a sample of the circle, which is a function of the azimuth orientation. The low-level processing consists of normalization, median filtering, and fitting a piecewise-linear function to the data, as we shall describe later in the paper. The result is represented symbolically as sequences of linear segments and some of these are selected as *characteristic features* for matching between images.

Before each incremental step of movement, the robot acquires an image of the current view, which is compressed and represented symbolically, and is compared with the target view, which has been processed *a priori* in a similar manner. Matching is done first on the basis of qualitative measures of the characteristic features and then quantitative measures are used to verify the match. Finally, 3D geometrical analysis is used to determine the next homing movement. The 3D geometric knowledge itself is computed from the matches of the characteristic features together with the distance traveled as measured by an odometer. Thus, there is no requirement of a 3D model and all feature matching is image-based.

3. GEOMETRICAL ANALYSIS

The geometry of the imaging process, as depicted in Fig. 3, is *not* the typical projection of a sphere onto a plane. It is a reflection of the environment off a spherical mirror. Therefore, the typical spherical geometry cannot be used. For the analysis we will use a spherical coordinate system centered at the origin of the image plane.

Let R represent the radius of the spherical mirror, d the distance between the apogee of the sphere and the focus of the camera, and let f be the focal length of the camera. In this configuration, the center of the sphere and the focus of the camera will both reside on the Z axis of the coordinate system. An arbitrary point P in 3D space can be uniquely determined by its camera-centered coordinates (ρ, α, β), as illustrated in Fig. 3. A projected sideview of the configuration (depicted in Fig. 4) will enable us to derive the mapping function from an arbitrary 3D point $P(\rho, \alpha, \beta)$ to its corresponding image point $P_i(r_i, \alpha_i, 0)$.

Let γ be the incident angle of P (relative to the perpendicular on sphere at incident point Q) and δ be the orientation angle between the incident line of P and the horizontal line of the incident point Q of P, as indicated in Fig. 4. If we let θ denote the angle between the reflecting line of Q and Z axis of the coordinate system, we can derive the following relationship between the 3D point $P(\rho, \alpha, \beta)$ and its corresponding image point $P_i(r_i, \alpha_i, 0)$: (see the Appendix for derivations of these equations)

$$\theta = 2\gamma - \delta - \frac{\pi}{2} \tag{1}$$

$$r_i = f \tan\theta = -f \cot(\delta - 2\gamma) \tag{2}$$

$$\alpha_i = \alpha + \pi \tag{3}$$

where γ and δ are the two parameters of the mapping function and are determined by the following two equations:

$$\tan\delta = \frac{\rho\sin\beta - (R + d + f - R\sin(\gamma - \delta))}{\rho\cos\beta - R\cos(\gamma - \delta)} \tag{4}$$

$$\frac{d}{R} + 1 = -\frac{\sin\gamma}{\cos(\delta - 2\gamma)} \tag{5}$$

From Fig. 5(a), it can be seen that the horizontal light rays which are incident to the mirror and are reflected through the focal point form a single plane, which is horizontal in 3D world and at a fixed height. All other horizontal rays will not be seen (see Fig.5(b)). Hence, the rest of an image is formed from reflection of non-horizontal rays. Moreover, every point in this specific horizon plane will project to the image plane as a point on a circle. We call this circle the *horizon* *circle*, and therefore, the *horizon plane* can be defined as the set of points whose image projection is the horizon circle. The equation of the horizon plane is:

$$\rho \sin\beta = d + R + f - R\sin\gamma \tag{6}$$

Since the horizon plane has $\delta = 0$, the general mapping functions in Eq. 1 to Eq. 5 will be simplified as

$$r_i = -f \cot 2\gamma \tag{7}$$

$$\alpha_i = \alpha + \pi \tag{8}$$

where γ is a fixed angle and is determined by Eq. 5 directly.

Therefore, under this configuration, the mapping from 3D space to the image plane can be decomposed into three parts:

- the $360°$ view of the horizon plane, i.e., the horizon circle;
- the scene above the horizon plane (open half-space), consisting of those image points outside the horizon circle;
- the scene below the horizon plane (open half-space), consisting of the image points inside the horizon circle.

Theoretically, this partition of the image plane is preserved as the (ideal) robot moves on a perfectly planar surface, i.e., the scenes projected to the points outside the horizon circle can potentially move to any place outside the horizon circle, but never go onto or inside the circle; the scene projected to the points inside the horizon circle can potentially move to any place inside the circle, but never get onto or outside the circle; the scene projected onto the horizon circle can potentially move to any place on that circle, but never depart from that circle. Therefore, taking the horizon circle in the image plane as the raw data of the navigation system will not only take advantage of this geometric invariance, but also help compress the conventional 2D image data to a 1D signal to reduce the huge amount of computation in the feature matching problem, and also save great amount of storage.

4. SYMBOLIC ENCODING OF LOCATION SIGNATURE

As stated in the previous sections, the horizon circle is extracted from an image as the raw data of our system. In practice, the horizon circle is actually derived from a thin annular band with a width of 5 pixels and sampled every one degree, as is shown in Fig.2(b). This will serve to allow some immunity to noise and encross in calibration of the location of the horizon circle[4]. The summation of the grey-levels of the 5 pixels along each radial slice results in a waveform that serves as a one-dimensional location signature as depicted in Figs. 6 – 8.

The general approach involved in the low level processing is to segment the location signature to produce intervals that are flat, monotonically increasing, or monotonically decreasing. Transformation of the original location signature via low-level processing of the waveform has been developed to obtain reliable partitions in the face of noise and variability due to changes in illumination, digitization, sensor noise, etc. The processes involved are: **normalization**, **median filtering**, and **piecewise fitting**.

If a location signature is denoted as $V = \{v_1, v_2, ..., v_{360}\}$, then *normalization* is achieved by dividing the measurements by the maximum value, i.e.,

$$v_i' = \frac{v_i}{\max\{v_j | v_j \in V\}} \qquad i = 1, ..., 360 \tag{9}$$

Median filtering with a window length of 5 removes many of the impulse errors. That is,

$$v_i'' = median\{v'_{i-2}, v'_{i-1}, v'_i, v'_{i+1}, v'_{i+2}\} \qquad i = 1, ..., 360 \tag{10}$$

Piecewise fitting is accomplished with the Land-McCann retinex algorithm[15, 16, 17] as follows:

$$v_i''' = \begin{cases} v''_i & \text{if } v''_i - v''_{i-1} > \sigma_v'' \\ v''_{i-1} & \text{otherwise} \end{cases} \qquad i = 1, ..., 360 \tag{11}$$

where σ_v'' is a threshold determined dynamically as the standard deviation of v''.

Figs. 6 – 8 show the raw data of sample current and target location signatures for translation and rotation with the range expected during movement of our robot. Fig. 9 depicts transformation of a sample raw signature from 0° to 100° into a symbolic encoding of the waveform.

As a result of this processing, a signature is represented as a sequence of symbolically labeled segments in a way that simplifies the matching problem by providing a first stage of qualitatve matches for landmarks.

5. MATCHING BETWEEN VIEWS

The matching process uses qualitative measures to hypothesize correct matches between the current location and target location signatures. The first step of this process is the decomposition of each location signature into segments with one of three types of topological properties: monotonically increasing, monotonically decreasing, or roughly constant. Thus, we attach a label to each segment which is either $+$, $-$, or 0. By definition no two adjacent segments can have the same type. Fig. 10 is a portion of two segmented signatures for which there are 5 segments with the same sequence.[2]

The "shape" of a sequence of segments will be defined as the distinct symbolic encoding of the segments into a label sequence. It is obvious that with the small number of segment types, there is still a manageable number of shapes for sequences of n segments, where n is 5 or less. With $n = 5$, since an adjacent segment may only be one of two types (by the definition of the encoding), there are a total of $3 \times 2 \times 2 \times 2 \times 2 = 48$ qualitative shapes. Although a signature is symbolically expressed in terms of these qualitative properties, it is still necessary to describe it in a precise quantitative form. Three measurements describe a segment quantitatively: the "jump", the "span", and the "azimuth". Since the waveform was sampled at every degree, each point of a segment is indexed by its angle. Formally, given a segment $\{v_i, v_{i+1}, ..., v_{i+k}\}$, the *jump* of the segment is defined as $||v_{i+k} - v_i||$; the *span* of the segment is defined as $k + 1$; and the *azimuth* of the segment is defined as the azimuth angle of the center of the segment, i.e., $(i + \frac{k}{2}) \; mod \, 360$. The qualitative topological "label" sequence together with these three quantitative measures form the description of a segment.

5.1 Characteristic Features

In terms of the signature, a *characteristic feature* will be defined as the segments that are most distinctive and reliable for matching, i.e., those parts of the waveform where there is a large slope. Thus, a characteristic feature is a segment with a large ratio of *jump* to *span*. In order to solve the correspondence problem, we first extract the strongest characteristic features for matching signatures by selecting the L segments in the target view which have the greatest slopes. The context around a characteristic feature will be expanded by k segments to both sides to form a *landmark template* of length $2k + 1$ segments. Finally, each landmark template is searched against

[2]In the following context, we sometimes use the term "signature" synonymously with the term "segmented signature".

the current view to see if there is a match by applying the following two-stage process: *hypothesis generation* and *statistical verification*

5.2 Hypothesis Generation

Generation of candidate matches consists of two steps:

- **Qualitative Matching**: The shape (i.e., sequence of labels) of each landmark template is used to match between current and target views to decrease the size of the search space.
- **Quantitative Matching**: The *jump* to *span* ratios of the segments of the landmark template are used to produce a quantitative measure of the match against each candidate subsequence obtained in qualitative matching.

The quantitative evaluation function between the landmark template and a candidate subsequence of the same shape is defined as follows:

$$f(t, c) = \sum_{i=-k}^{k} \eta_i g(c_i, t_i) \tag{12}$$

Where $t = \{t_i | i = -k, ...k\}$ is the template, $c = \{c_i | i = -k, ..., k\}$ is the candidate, t_i and c_i are the relevant segments, η_i is a weight coefficient, which may weight the middle terms more than the ends, and $g(c, t)$ is a similarity function of two segments c and t, based on their *jumps* and *spans*. There are several ways to determine the similarity between two segments which have the same label. A simple and efficient way to measure the similarity between segments of the same label type is to directly calculate the relative similarity between the two in terms of the *jumps* and *spans*. For example, the relative similarity in *jump* can be used:

$$g_{jump}(c, t) = 1 - \frac{\|jump(c) - jump(t)\|}{jump(t)} \tag{13}$$

For simplicity, the function is truncated at zero, so that negative values do not occur. A similar definition can be applied to *span* to get g_{span}. Hence, a linear combination of g_{jump} and g_{span} is used to represent the similarity between two corresponding segments, i.e.,

$$g(c, t) = \alpha_j g_{jump}(c, t) + \alpha_s g_{span}(c, t) \tag{14}$$

where the weight for g_{jump} (i.e., α_j) is greater than the weight for g_{span} (i.e., α_s) because the *jump* measurement preserves more invariance under low-level processing than the *span* measurement.

5.3 Statistical Verification

Since the hypothesis generation is strictly local, it is possible that if there are several similar characteristic features, mismatches of landmarks will occur. An effective way to remove mismatches is to do a global statistical analysis of the change in *azimuth* for each landmark. In other words, if we assume that the visible surfaces in the environment are not close to the robot and the translation of the robot is relatively small, then none of the landmarks can move much more than the average motion of the others. (see the next section). Let us define the difference between the azimuth of a landmark and that of its potential match as the *displacement* of the landmark. The mean of the displacements can be used to retain only those that are within some given deviation around the mean. After this of removing potential outliers, the resultant pairs are called matched pairs. Fig. 11 shows two views symbolicaly represented matched with each other.

6. 3D MOTION ANALYSIS

By using the image motion of the projected environmental surface and the partial information about the robot navigation, the distance from the robot to the matched landmarks can be derived. Thus, it is possible to compute the current pose (position and orientation) of the robot with respect to the target pose. In other words, we can compute the translation and rotation that will take the robot to the target location. This contrasts with our previous approach in [4], where the rotation was assumed to be very small and only the direction of translation was determined[3] The problem of computing the motion parameters is simplified in the case of a 360° view. While for image sequences with a narrow field of view, the effects of translation and rotation are confounded in the image, this is not the case for a 360° view.

6.1 Motion Model

Let us consider Fig. 12. If there is only a pure translation involved between the current view and a target view, the landmarks appearing directly in front or 180° behind the translation direction

[3]Note that this original assumption was not completely unreasonable because the spherical mirror was mounted independently from the rotational platform, and therefore rotational changes only enter via slippage in tires, vibrational side effects, ect. However, over time rotation did enter and require recalibration.

will have no displacement in the two signatures. All others will be displaced some distance around the horizon circle moving away from the direction of the translation as a function of the magnitude and direction of the robot movement and the distance of the surface. This case is shown in Fig. 6. On the other hand, if there is only a pure rotation involved, every landmark will have a constant angular displacement between the two signatures. This fact is shown in Fig. 7. The general case of motion with a combined translation and rotation can be interpreted as a rotation followed by a translation. Therefore, in order to let the robot steer itself to home precisely, the motion from an initial pose to a target pose needs to be decomposed into *rotation* and *translation* parameters. Since we assume that the robot is moving on a surface that is approximately planar, there is only one rotation parameter (perpendicular to the plane) and two translation parameters. The first step of the analysis is to estimate the rotation parameter.

6.2 Estimation of rotation

In a general motion model, it has been proved that the rotation and the translation are linearly separable[18], i.e., the rotation and translation can be solved sequentially. This property is also valid in our system, where an image is taken from the reflection off a spherical mirror. Let us consider Fig. 12 again. Suppose we have two views taken from poses that differs by both rotation and translation, as depicted in Fig. 12(c). Imagine that the rotation parameter ω_R, which is the difference of the heading orientations of the two poses, is already known. Then the robot can first rotate that angle and results in a new pose which has the same heading orientaion as pose 2 and only has a pure translation remained. Thus if rotation is known then the combined motion problem boils down to a pure translation problem as indicated in fig. 12(a). Therefore, a combined motion problem can be separated into two independent problem: rotation and translation, and can be attacked by solving rotation first.

Now, let us see Fig. 12(a) again, where there is only a pure translation involved. As indicated in this Figure, a landmark appears to the left side of the translation direction will undergo a counter-clockwise circular motion flow; similarly, a landmark appears to the right side of the translation direction will undergo a clockwise circular motion flow. Moreover, since we have pointed out that the landmarks appearing directly in front or 180° behind the translation direction have no displacement in the two signatures, they will form two foci in the motion flow distribution on the horizon circle, one being focus of expansion and the other being focus of contraction. Even if there is no landmark appearing directly in front or 180° behind the translation direction, these two foci will still exist in the motion flow on the horizon circle. This effect is illustrated in Fig. 1. Therefore,

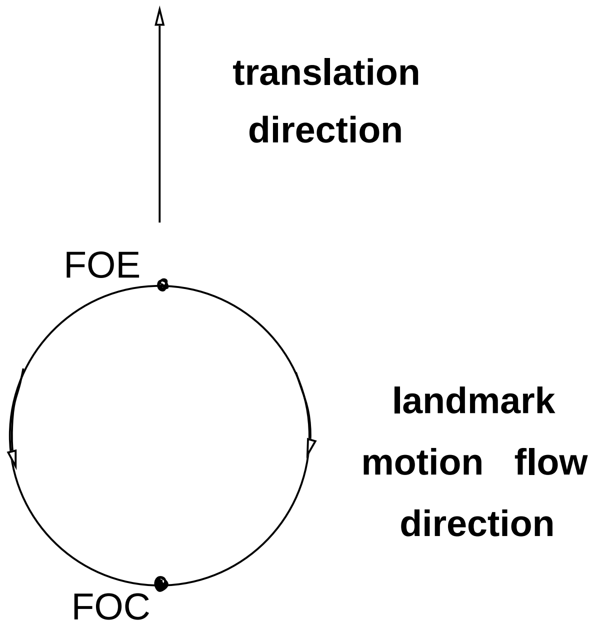

Figure 1: The translation direction can be determined by the motion flow of the landmarks. Here FOE and FOC stand for Focus Of Expansion and Focus Of Contraction

if we assume the environment is symmetrically distributed on both sides of the translation direction, The statistical mean of displacements of landmarks is zero, because the displacements located on the counter-clockwise part and on the clockwise part will cancel with each other. Hence, since the combined motion can be regarded as a superposition of a pure rotation followed by a pure translation as discussed above, the pure rotation angle is exactly the mean of the displacements of the landmarks. In practice, the environment may not be symmetrically distributed. But the same effect can be achieved by weighting the displacement of each landmark by the distance to its nearest neighbor. Once the rotation angle is computed, the current signature is transformed by the inverse of the rotation, so that the mean displacement is zero.

6.3 Estimation of Translation

Since the rotation is estimated first, we can assume that the motion is a pure, non-zero translation. In this case, as we pointed above, there is a focus of expansion (FOE) and a focus of contraction (FOC) on the horizon circle. From Fig. 1, it can be seen that the translation direction is the azimuth angle of the focus of expansion. Note that both the FOE and FOC are simultaneously visible and can be used. Hence, the translation direction can be estimated by searching around the motion flow of the horizon circle to get that focus.

The translation magnitude can be estimated if the distance between two successive views is available via odometry. In our case, if the robot moves straight ahead, then encoders on the motors can be used to obtain this information (within limits imposed by slippage of the wheels, and disregarding noise).

As shown by Fig. 13,

$$\omega_i = \frac{v}{r_i}\sin(\phi - \theta_i) \tag{15}$$

where r_i is the distance to the ith landmark, v is the translation velocity of the robot, ω_i is the velocity on the horizon circle of the ith landmark (i.e., its motion flow), ϕ is the estimated translation direction (in terms of the azimuth angle of FOE), and θ_i is the azimuth angle of the ith landmark. All the quantities are measured in terms of the current pose. In our experiment, since $r_i \geq$ 3ft., and $v \leq$ 2ft./step, the theoretical upper bound for ω_i is 0.67 rad./step.

For discrete motion, the equation can be rewritten as:

$$\delta\theta_i = \frac{\delta d}{r_i}\sin(\phi - \theta_i) \tag{16}$$

Thus, we can estimate the translation distance by first moving a nominal amount, then computing environmental distances r_i from each landmark. By matching the new current view against the target view, we can use Eq. 16 to estimate the distance δd to the target. Since each landmark can give a different value for δd, the median is used.

It should be pointed out that a similar analysis on derotation and estimation of translation direction is also independently done by Nelson and Aloimonos[19]. However, their algorithm for derotation and estimation of the translation direction is quite different from ours. They approached the problem by searching around a 2D parameter space for rotation and translation direction. That results in a $O(n^3)$ algorithm. In our case, we assume that the robot moves in the ground plane, instead of in the general 3D space. Moreover, since calculating the mean of displacements and searching for FOE around motion flow on the horizon circle in our algorithm are both linear, the whole algorithm is in $O(n)$. Note the problem in our case is slightly differently posed from theirs. The basic difference is the 180° criterion vs. average displacement. They approached the derotation problem under the assumption that only two images (i.e., only one motion flow image) are given, whereas in our case, we can "test on the fly", i.e., get an image and derotate; then get a new one and see if we need to derotate again. Therefore, they need more computation than ours. We implemented their algorithm and simulated it by using our experimental data. The results (see Sec. 7.) show that their algorithm worked well for estimation of rotation angle, but not well for estimation of translation direction. Moreover, they didn't estimate the translation distance and we did.

7. EXPERIMENTAL RESULTS

The performance of our system has been demonstrated experimentally. The mobil robot is a Denning DRV-1 model, named Harvey. The physical system has already been shown in Fig. 2(a). In each of the navigation experiments, we first take a sequence of target images at different locations along the navigation path. This is an off-line procedure. The on-line navigation process is accomplished by homing to each target along the path. For each homing process, a combined motion between the current location and the target location is assumed and thus our motion model is applied, i.e., after taking a current view, the matching procedure between the current view and the target view is called and the rotation parameter is computed; then the robot derotates the angle to eliminate outliers. This process continues until the rotation is cancelled. At this stage, the translation direction is determined. The translation magnitude can be computed if the landmark distance have been already determined. Otherwise, a nominal distance is used via odometry. This process continues until the displacement between a current view and the target view is reduced to the level of a standard deviation. This guarantees that the robot arrives at the target precisely.

Fig. 14(a) shows a sample picture from one of the experiments, in which the robot navigated along a curved path around tables in a lab. The environment was cluttered, as shown in the figure. However, the robot homed to each target successfully. Fig. 14(a) is the target view of the homing process and Fig.14(b) is the initial view which is located about 2ft. away from the target with a heading of about $40°$ from the homing target. The horizon circle and the coordinate axes are visually marked in the figures. Table 1 lists the matching results for these two views. The order of the matched pairs is listed in terms of the "significance" of the characteristic features in the target view, i.e., in the order from the feature with the greatest slope to the feature with the smallest one. The last column in each of the tables indicates whether a computed match is actually correct. A rotation estimate of $29.9°$ was obtained. This was used to eliminate some of the matches, and a new estimate of the rotation was obtained from the remaining ones. The matching results at this stage is shown in Table 2. After the second iteration was performed (see Table 3), the displacements are less than 5°[4] and the translation direction is estimated as 325°.

To compare the performance of our algorithm with that of Nelson and Aloimonos [19], we produced the motion flow maps around the horizon circle by linear interpolation based on those matched characteristic feature pairs shown in Table 1 and 2. In order to let the produced motion flow as "pure" as possible, we did not include those "pseudo-matched" pairs. Table 6 shows the results of their algorithm. It can be seen that their derotation estimation is very close to ours, whereas their estimation of translation direction has very big errors. The reason is that, as they mensioned in their paper, there is a trade off between precise determination of derotation and precise determination of translation direction. The key problem comes from how precise a motion flow "shape" is. However, the problem itself involves many hard problems, such as matching. Therefore, it is difficult to evaluate how accurately the motion parameters can be determined when their algorithm is applied in practical applications.

Once the rotation and translation direction are computed, the robot moves 0.5 ft. At this stage, a new matching list is output as shown in Table 4 and the averaged displacement angle is 2.5°. This angle is small enough so that the algorithm deems that the rotation is 0, and the robot continues to translate in the same direction. Based on the matching results of Table 3 and 4 and Eq. 16, the algorithm determines the next translation distance to be 1.3 inches and executes that motion to arrive at the pose as shown in Fig.14(c). The matching result at this stage is listed in Table 5 and the maximum displacement is 4° and averaged absolute displacement is within 2°. The algorithm determines that the target and current signatures are essentially the same and terminates.

The final pose is 3 inches away from the target position and 3° away in heading when starting from 2ft. and 40°. This error is tolerable for the purpose of homing. Moreover, the error is not accumulated in the navigation process because the final pose in the current homing stage is the initial pose of the next homing stage, which makes no assumptions about the pose of the robot. It can been seen from the experiments that the correct feature matching rate is around 90%. This represents a significant improvement over the previous approach [4] involving only individual normalized segments. Furthermore, the current system works for general planar motion and computes distance to landmarks.

[4]Empirically the final error of the derotation is about 3° to 5°. Hence we set 5° as the threshold

8. CONCLUSION

In this paper, we presented a navigation system that involves an image-based homing scheme using 360° views. A robust feature matching algorithm is developed using qualitative geometric descriptions of waveform segments around prominent features. The robot successfully homes to a sequence of target views in a complex environment. As part of the homing control, the motion parameters are approximately decomposed into rotation and translation. The rotation is estimated first, followed by translation direction. The distances to landmarks in the environment are computed and used to estimate the distance from the current location to the target location.

In the future, we will investigate the feasibility of automatically acquiring a model of the environment that includes the 3D relationships between the target locations. This will provide a foundation for more sophisticated planning and a variety of navigation experiments.

ACKNOWLEDGEMENTS

The first author would like to especially thank one of his advisors, Prof. Allen Hanson for his encouragement. We also thank Brian Pinette for his offering the code of Land-McCann retinex algorithm and Valerie Conti, Robert Heller and Jonathan Lim for their technical help.

APPENDIX

We here derive those equations appeared in Sec. 3.. For the sake of clearity of the derivation, we redraw Fig. 4 in Fig. 15.

In order to get Eq. 1, see $\angle CQE$, where CQ is the horizontal line to the incident point Q of P, QE is a vertical line passing Q. Hence $\angle CQE$ is a right angle. Now note $\angle DQF$ is the reflectant angle which is equal to the incident angle $\angle PQD = \gamma$, we have

$$\angle EQF = \angle O^{'}FP_i = \gamma$$

$$\angle DQE = \gamma - \theta$$

$$\angle CQD = \gamma - \delta$$

Thus,

$$(\gamma - \theta) + (\gamma - \delta) = \frac{\pi}{2}$$

Then, Eq. 1 immediately follows.

Eq. 2 follows the fact that in the right trianble $O^{'}FP_i$, we heve

$$r_i = ||P_iO^{'}|| = ||FO^{'}|| \tan\theta$$

Eq. 3 follows the fact that for all 3D point P, its image point P_i always has a π phase shift in XY plane (see Fig. 3).

Now, in the right triangle $\triangle OBQ$,

$$||OB|| = ||OQ|| \sin \angle OQB$$

Since

$$\angle OQB = \angle CQD = \gamma - \delta$$

$$||OQ|| = R$$

Hence,

$$||OB|| = R\sin(\gamma - \delta)$$

Since

$$||OO^{'}|| = R + d + f$$

Hence,

$$||BO^{'}|| = ||OO^{'}|| - ||OB|| = (R + d + f - R\sin(\gamma - \delta))$$

Since in the right triangle $\triangle PQO^{'}$,

$$||PG|| = ||PO^{'}|| \sin \angle PO^{'}G = \rho \sin\beta$$

Hence

$$||AB|| = ||AO^{'}|| - ||BO^{'}|| = ||PG|| - ||BO^{'}||$$

i.e.,

$$||AB|| = \rho\sin\beta - (R + d + f - R\sin(\gamma - \delta))$$

Assume C is the intersection point of the two perpendicular lines PG and QC, then

$$||QC|| = ||BC|| - ||BQ||$$

Since

$$||BQ|| = ||OQ|| \cos \angle OQB = R\cos(\gamma - \delta)$$

$$||BC|| = ||O^{'}G|| = ||PO^{'}|| \cos \angle PO^{'}G = \rho \cos \beta$$

We have

$$||QC|| = \rho \cos \beta - R\cos(\gamma - \delta)$$

Thus, in $\triangle PQC$,

$$\tan \gamma = \frac{||PC||}{||QC||}$$

That is followed by Eq. 4 and Eq. 6 is immediately followed by substituting $\delta = 0$ for Eq. 4. Now, let us see $\triangle OQF$. Obviously,

$$\angle OFQ = \angle O^{'}FP_i = \theta$$

$$\angle OQF = \angle OQB + \angle BQF$$

Since

$$\angle OCB = \gamma - \delta$$

$$\angle BQF = \frac{\pi}{2} - \theta = \pi + \delta - 2\gamma$$

Hence

$$\angle OQF = \pi - \gamma$$

Therefore,

$$\frac{||OQ||}{\sin \angle OFQ} = \frac{||OF||}{\sin \angle OQF}$$

That is immediately followed by Eq. 5.

REFERENCES

[1] M. Subbarao. *Interpretation of Visual Motion: A computational Study*, Pitman Publishing, Lodon, 1988.

[2] D. Zipser. "Biologically plausible models of place recognition and goal location", *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Volume 2: Psychological and Biological Models*, MIT Press, Cambridge, Massachusetts, 1986.

[3] C. Fennema, A. Hanson, E. Riseman, J. Beveridge and R. Kumar. "Model-Directed Mobile Robot Navigation", To appear in *IEEE Transactions on Systems, Man and Cybernetics.*

[4] J. Hong, X. Tan, B. Pinette, R. Weiss and E. Riseman. "Image-Based Navigation Using 360° Views", *Proceedings of Image Understanding Workshop 1990*, Pittsburg, Pennsylvania, 1990.

[5] R. Nelson. "Visual Homing Using an Associative Memory", *Proceedings of Image Understanding Workshop 1989*, Morgan Kaufman, California, 1989.

[6] J. R. Beveridge, R. Weiss and E. M. Riseman. "Combinatorial Optimization Applied to Variable Scale 2D Model Matching", *Proceedings of the Tenth International Conference on Pattern Recognition*, Atlantic City, New Jersey, June, 1990.

[7] R. Kumar and A. R. Hanson. "Robust Estimation of Camera Location and Orientation from Noisy Data Having Outliers", *Proceedings of the Workshop on Interpretation of 3D Scenes*, Austin, TX, Nov., 1989.

[8] E. Dickmanns and V. Graefe. "Dynamic Monocular Machine Vision", *Machine Vision and Applications*, Vol.1, 1988. pp. 223-240.

[9] E. Dickmanns and V. Graefe. "Applications of Dynamic Monocular Machine Vision", *Machine Vision and Applications*, Vol.1, 1988. pp. 241-292.

[10] N. Ayache and Faugeras. "Building, Registrating, and Fusing Noisy Visual Maps", *The International Jornal of Robotics Research*, Vol.7, No.6, Dec. 1988, MIT Press.

[11] G. Adiv. "Inherent Ambiguities in Recovering 3-D Motion and Structure from a Noisy Flow Field", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.11, No.5, May, 1989.

[12] R. Dutta and M. A. Snyder. "Robustness of Correspondence-Based Structure from Motion", *Proceedings of Third International Conference on Computer Vision*, Osaka, Japan, Dec., 1990.

[13] J. Zheng and S. Tsuji. "Panoramic Representations of Scenes for Route Understanding", *Proceedings of Tenth International Conference on Pattern Recognition*, IEEE Computer Society Press, 1990.

[14] Y. Yagi and S. Kawato. "Panorama Scene Analysis with Conic Projection", *Proceedings of IEEE International Workshop on Intelligent Robots and Systems*, IEEE Computer Society Press, 1990.

[15] E. H. Land and J. J. McCann. "Lightness and Retinex Theory", *Journal of the Optical Society of America*, Vol.61, No.1, 1971. pp. 1-11.

[16] E. H. Land. "Recent Advances in Retinex Theory and Some Implications for Cortical Computations: Color Vision and the Natural Image", *Proceedings of National Academy of Sciences*, Vol.80, No.16, 1983. pp. 5163-5169.

[17] B. K. P. Horn. *Robot Vision*, MIT Press, Cambridge, Massachusetts, 1986. pp. 185-200.

[18] R. Y. Tsai and T. S. Huang. "Uniqueness and Estimation of Three-Dimensional Motion Parameters of Rigid Objects with Curved Surfaces", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, No.6, pp. 13 – 26.

[19] R. C. Nelson and J. Aloimonos. "Finding Motion Parameters from Spherical Motion Fields (Or the Advantages of Having Eyes in the Back of Your Head)", *Biological Cybernetics*, 58, pp 261 – 273, 1988.

Figure 2: (b) A sample image taken in our experiment. The white ring indicates the rotation center of the robot and the marked ticks are the horizon circle, together with the coordinate axes

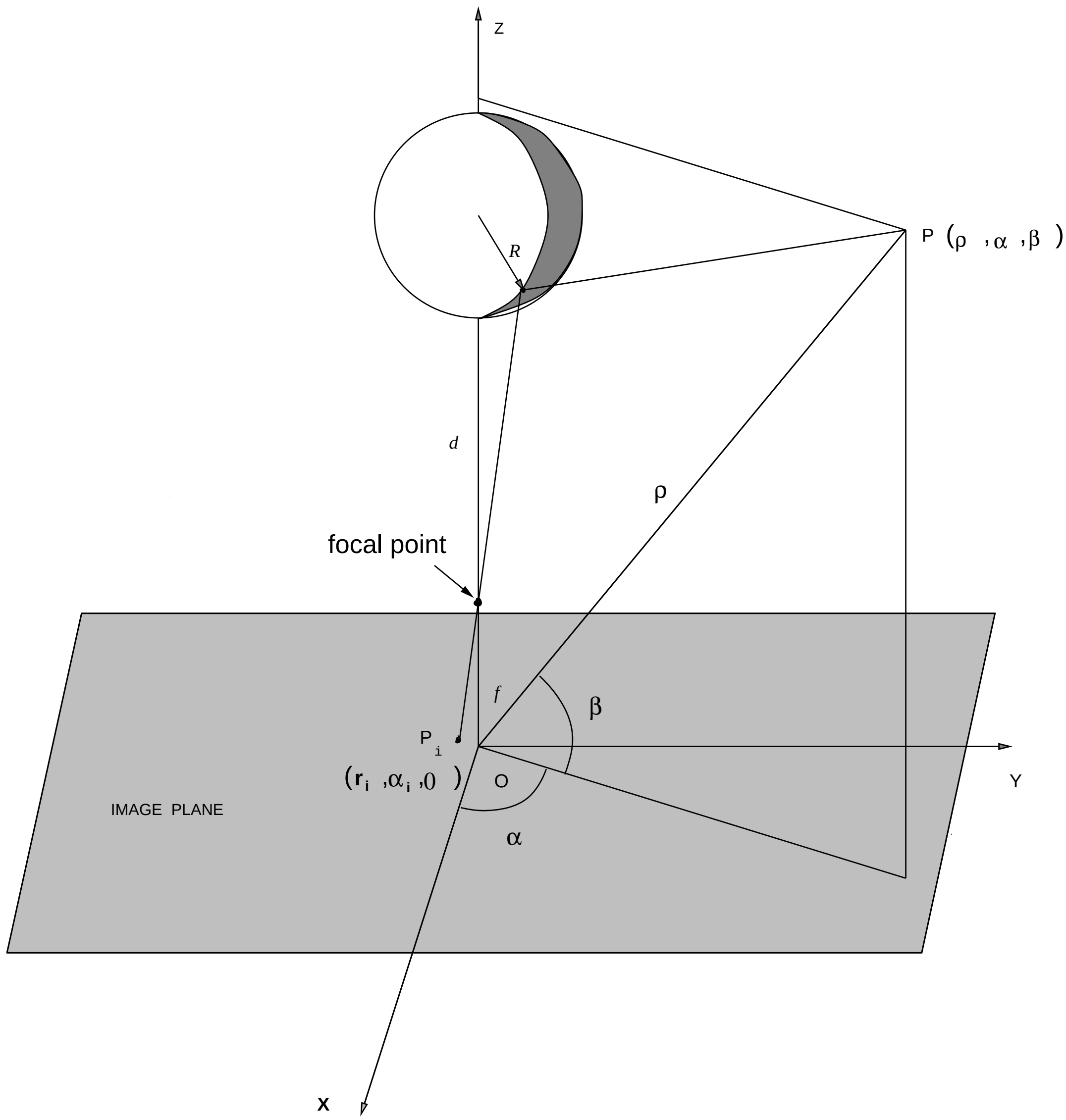

Figure 3: Geometrical model of the system. Every point P in 3D space maps a point P_i in the image plane through the camera focus. Here R is the radius of the sphere; f is the focal length of the camera; d is the distance between the apogee of the sphere and the focus of the camera.

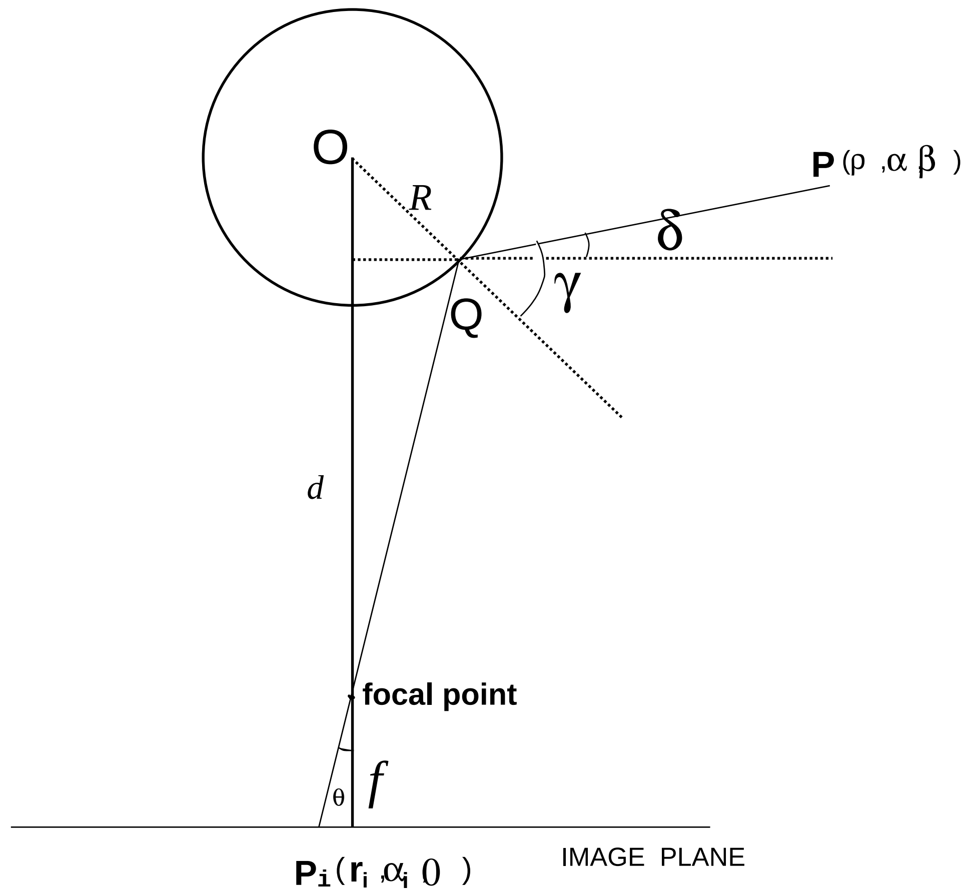

Figure 4: Side-view of the geometrical model of the system. A 3D point P maps onto image point P_i

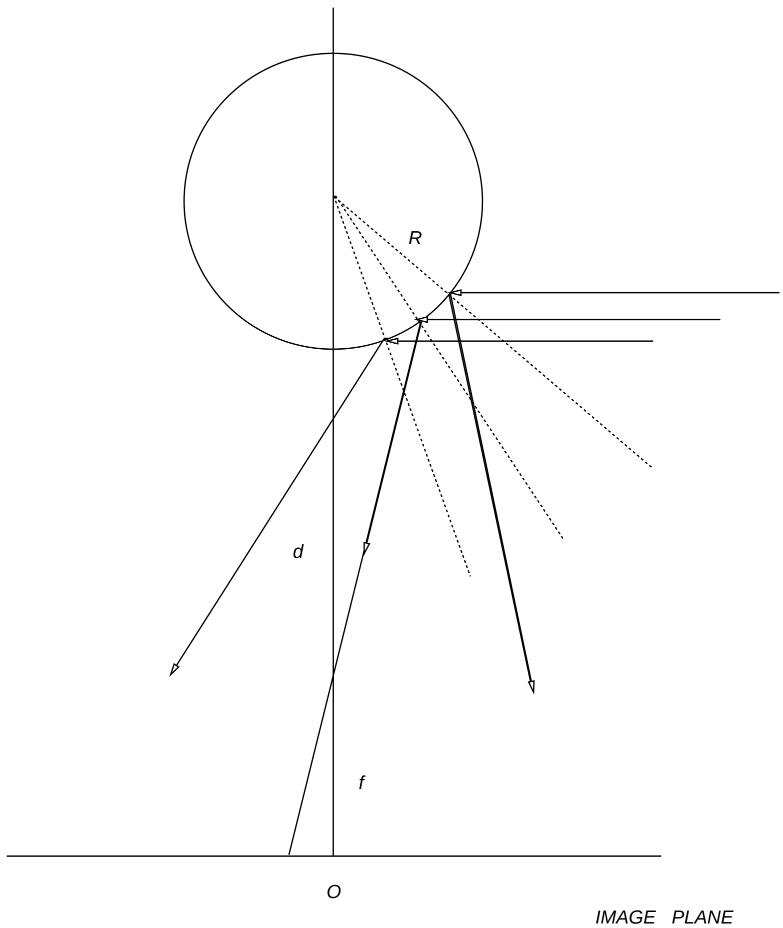

Figure 5: (a) Side-view of the geometrical model. Only one horizon line incident to the spherical mirror can pass through the focus of the camera, i.e., has its corresponding map on the image plane

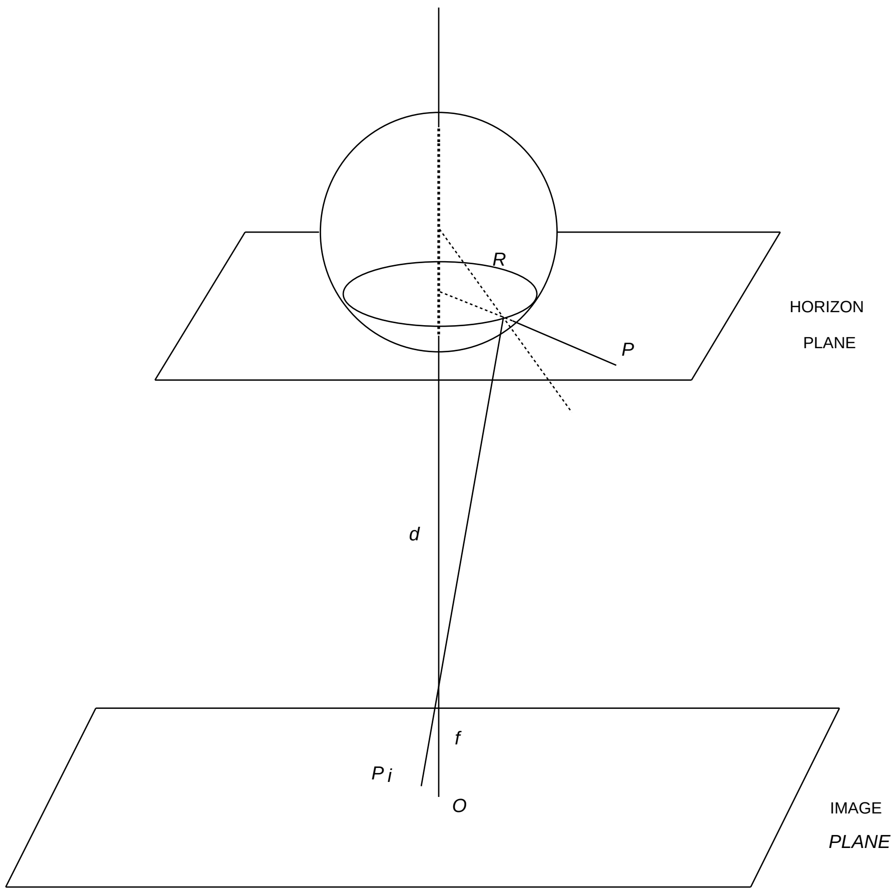

Figure 5: (b) Only one horizon plane can be seen

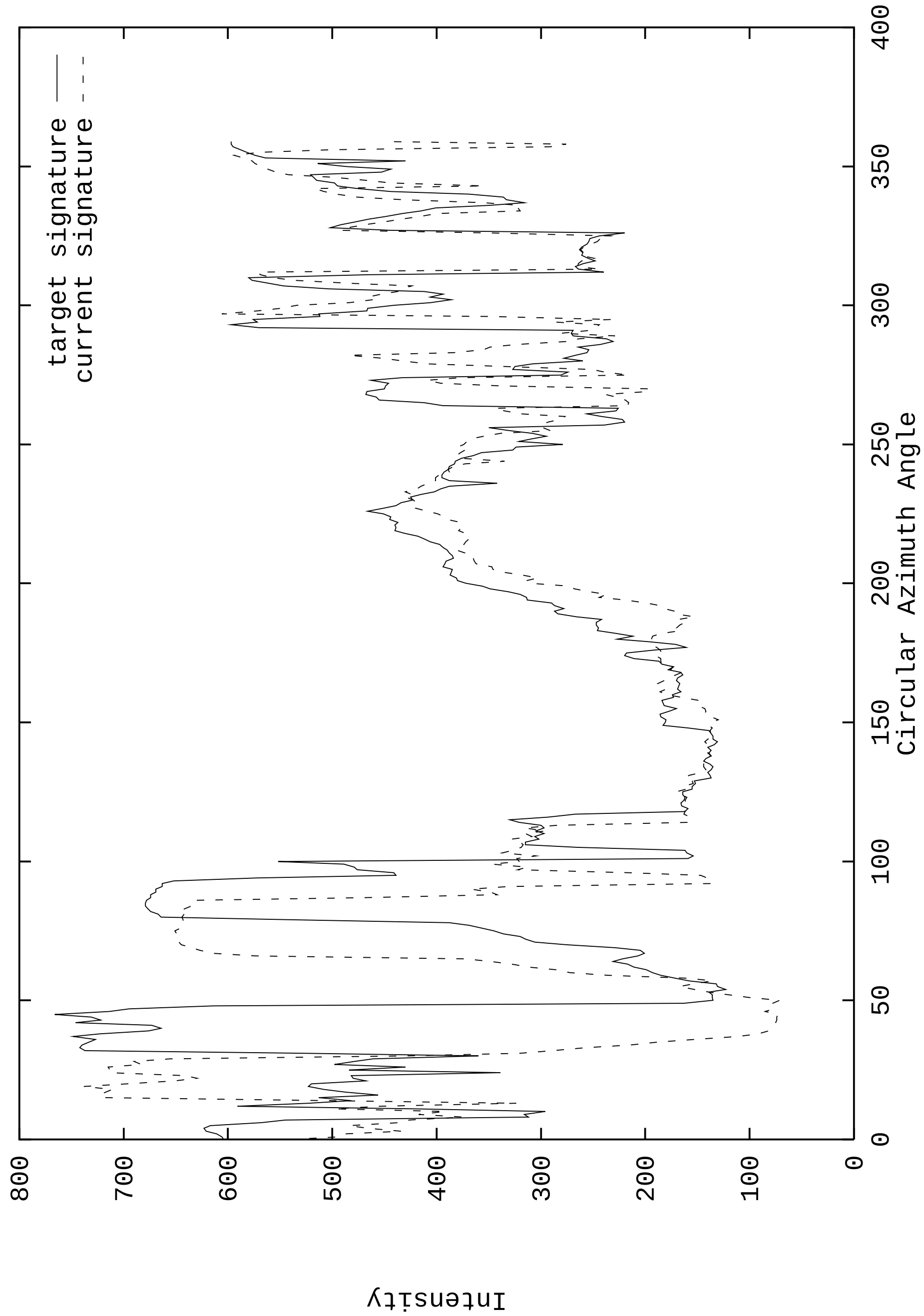

Figure 6: Raw data of a target location signature and a current location signature with a pure translation about 1.5 ft.

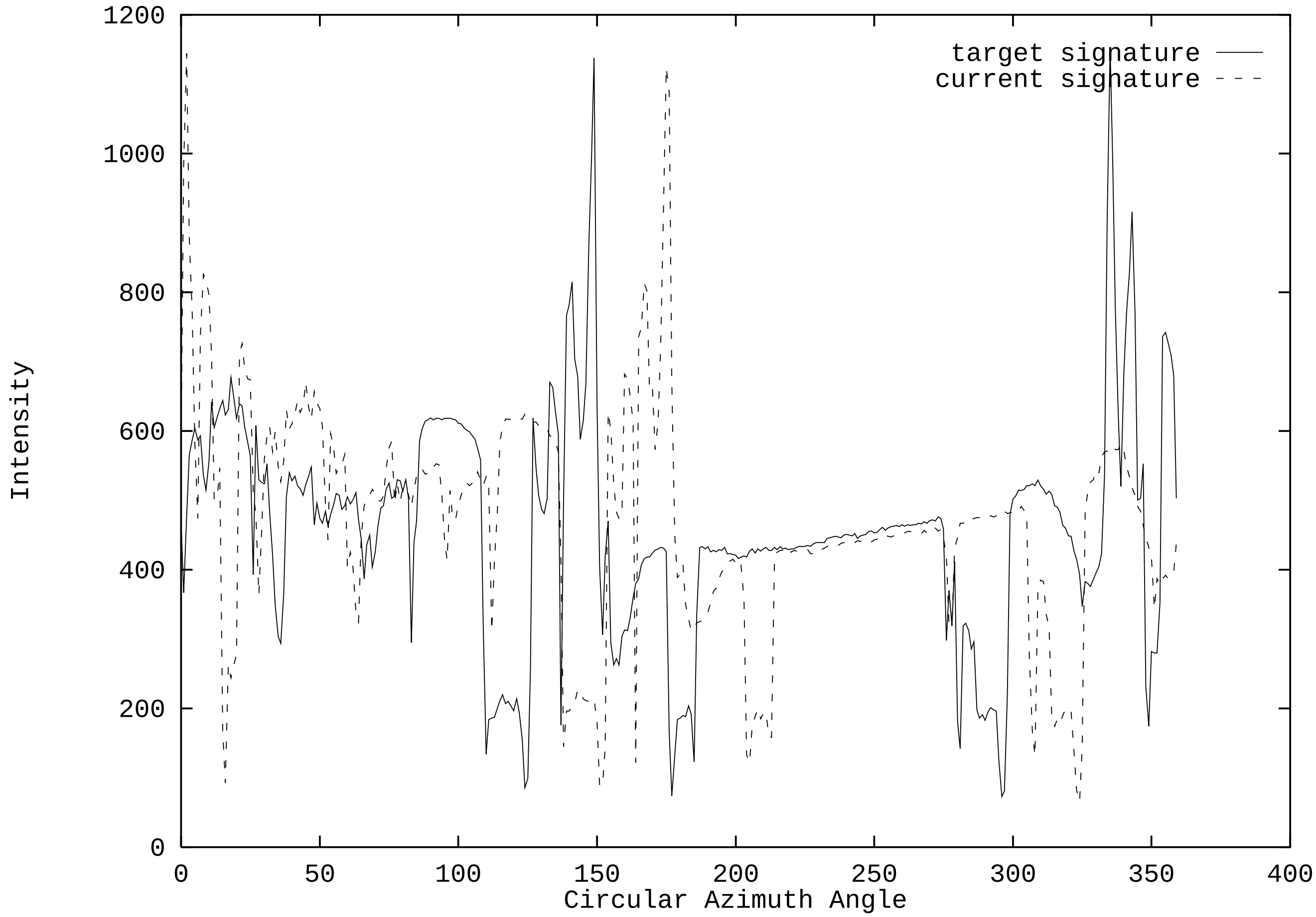

Figure 7: Raw data of a target location signature and a current location signature with a pure rotation about 30°

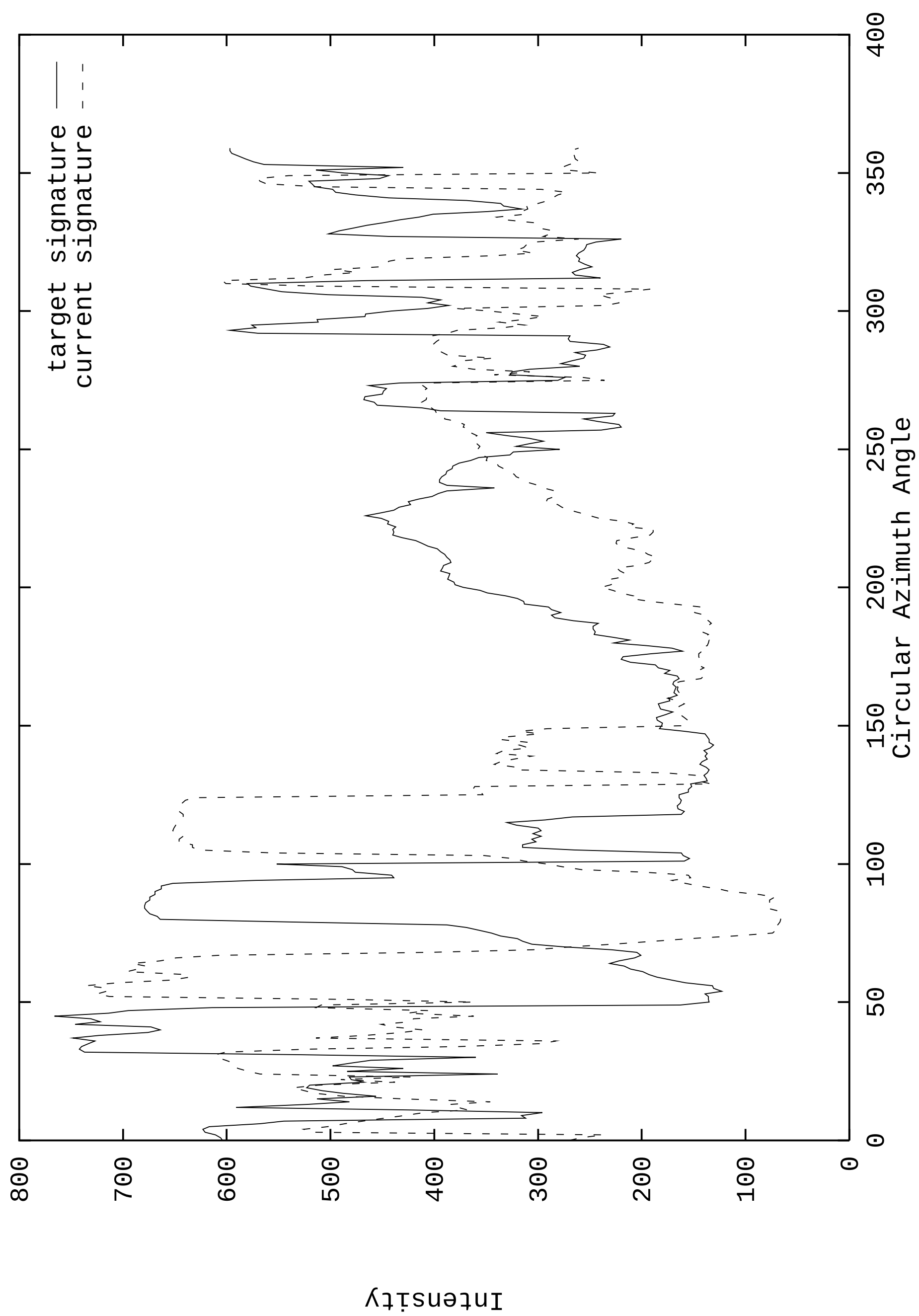

Figure 8: Raw data of a target location signature and a current location signature with a combined translation and rotation involved with about 2 ft. translation and about 40° rotation

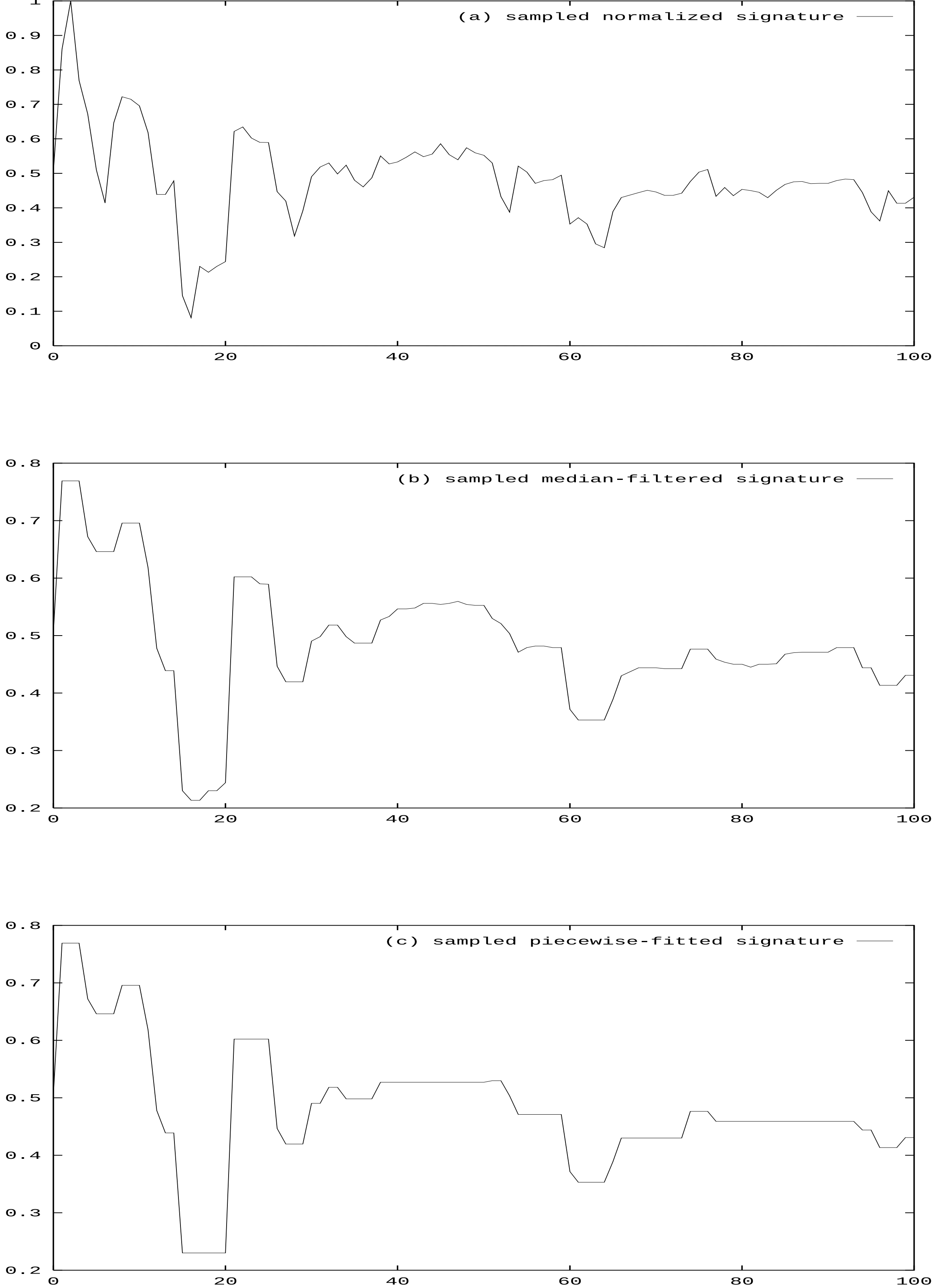

Figure 9: Symbolic encoding of the waveform during low-level processing. An enlarged portion of processing results of a location signature is shown. The raw data are the current location signature shown in Fig. 8. (a) result of normalization. (b) result of median filtering. (c) result of piecewise fitting to produce symbolic encoding that serves for qualitative matching

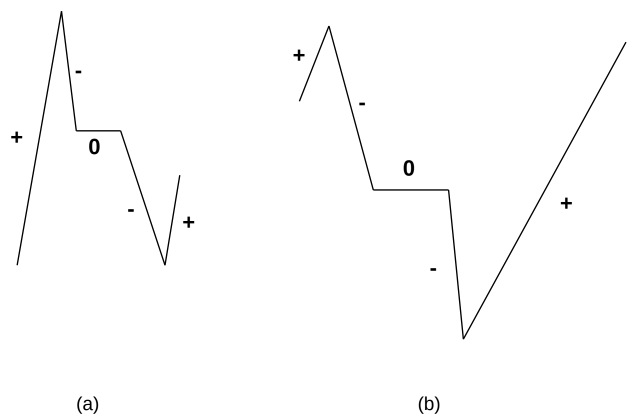

Figure 10: The qualitative shape of a sequence of segments. The two portions of the signatures have the same topological properties, and the shape is encoded as $+-0-+$

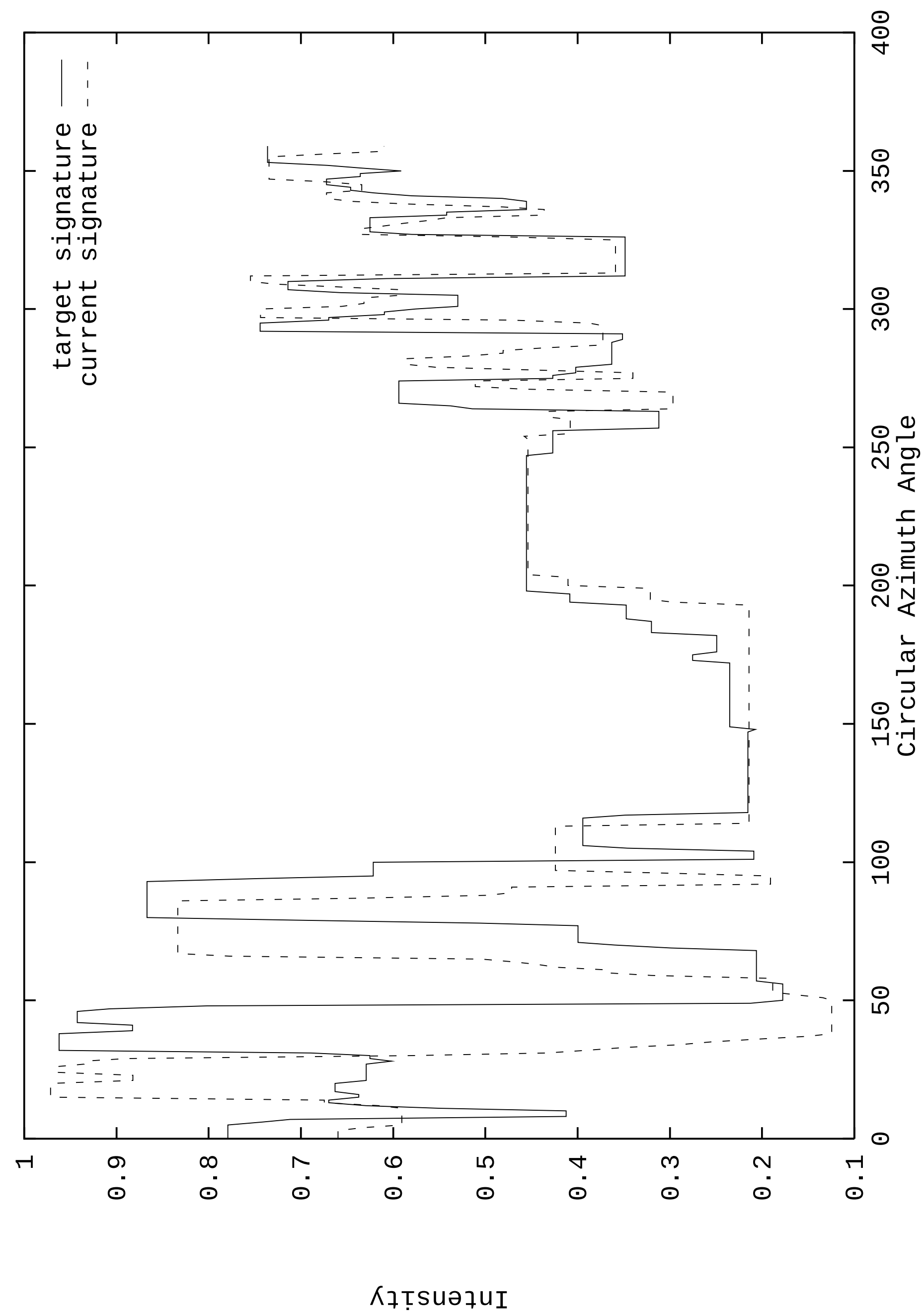

Figure 11: An example of matching between views. Here the two signatures are symbolically represented (i.e., after piecewise fitted). The solid segments in the dashed signature (current view), together with their context, are matched against their corresponding segments expressed as darkened in the solid signature (target view). To be graphically clear, the two views only involves a pure translation

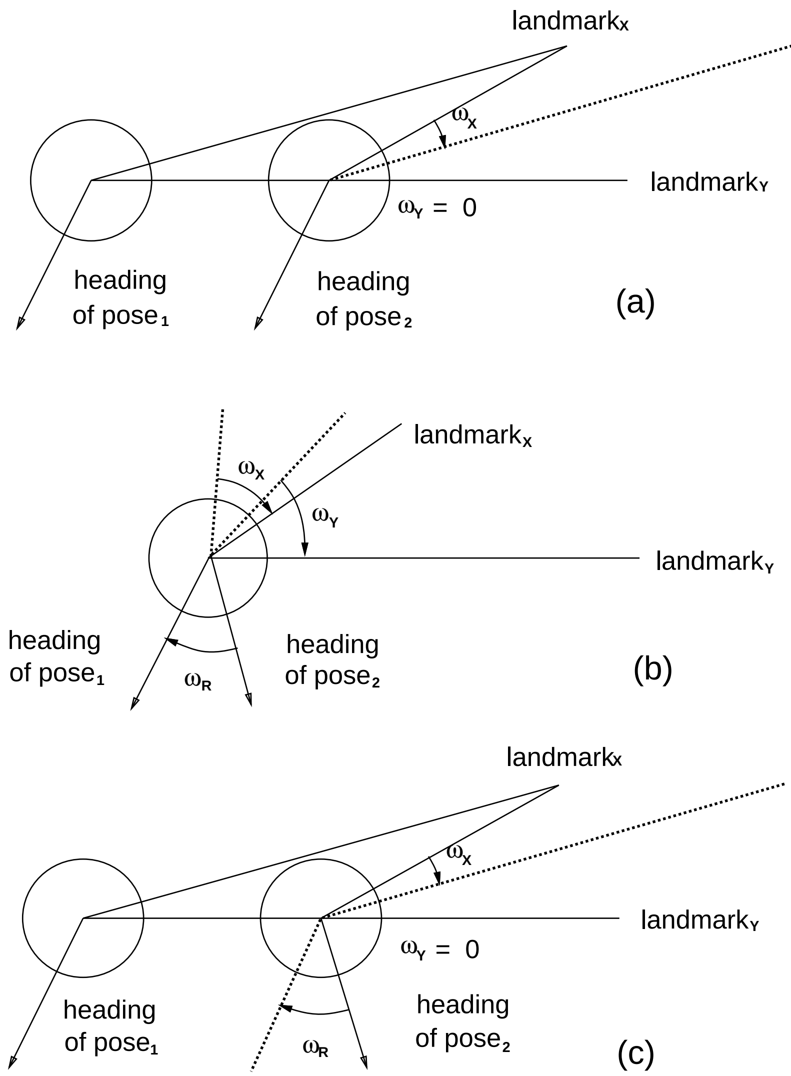

Figure 12: Relationship between the relative difference between two poses (before and after movement) and dispacements of the landmarks between two location signatures. Here ω_X and ω_Y are the corresponding angular motion velocities of landmark X and Y, respectively, ω_R is the angular motion velocity of the pose heading. (a) Two poses with a pure translation (b) Two poses with a pure rotation. Hence, $\omega_R = \omega_X = \omega_Y$ (c) Two poses with a combined motion; the angular displacements are the same as in (a) except that here there is a rotation ω_R added, which must be decomposed

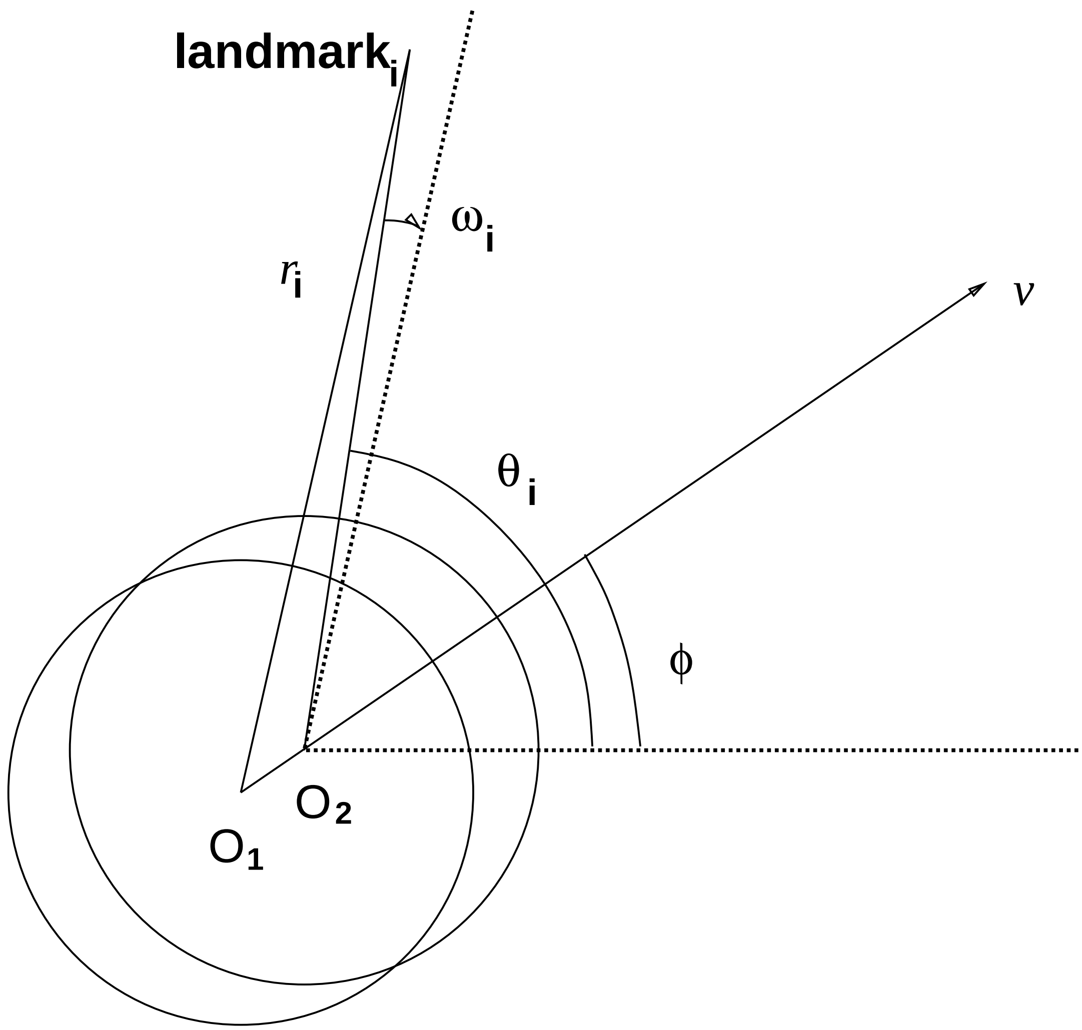

Figure 13: Relationship between the moving velocity of the robot and the angular moving velocity of a landmark. Note, to be visually clear, the relative sizes have been exaggerated. Since the moving step is much less than the environmental distance, θ_i and r_i are assumed to be the same for both poses O_1 and O_2

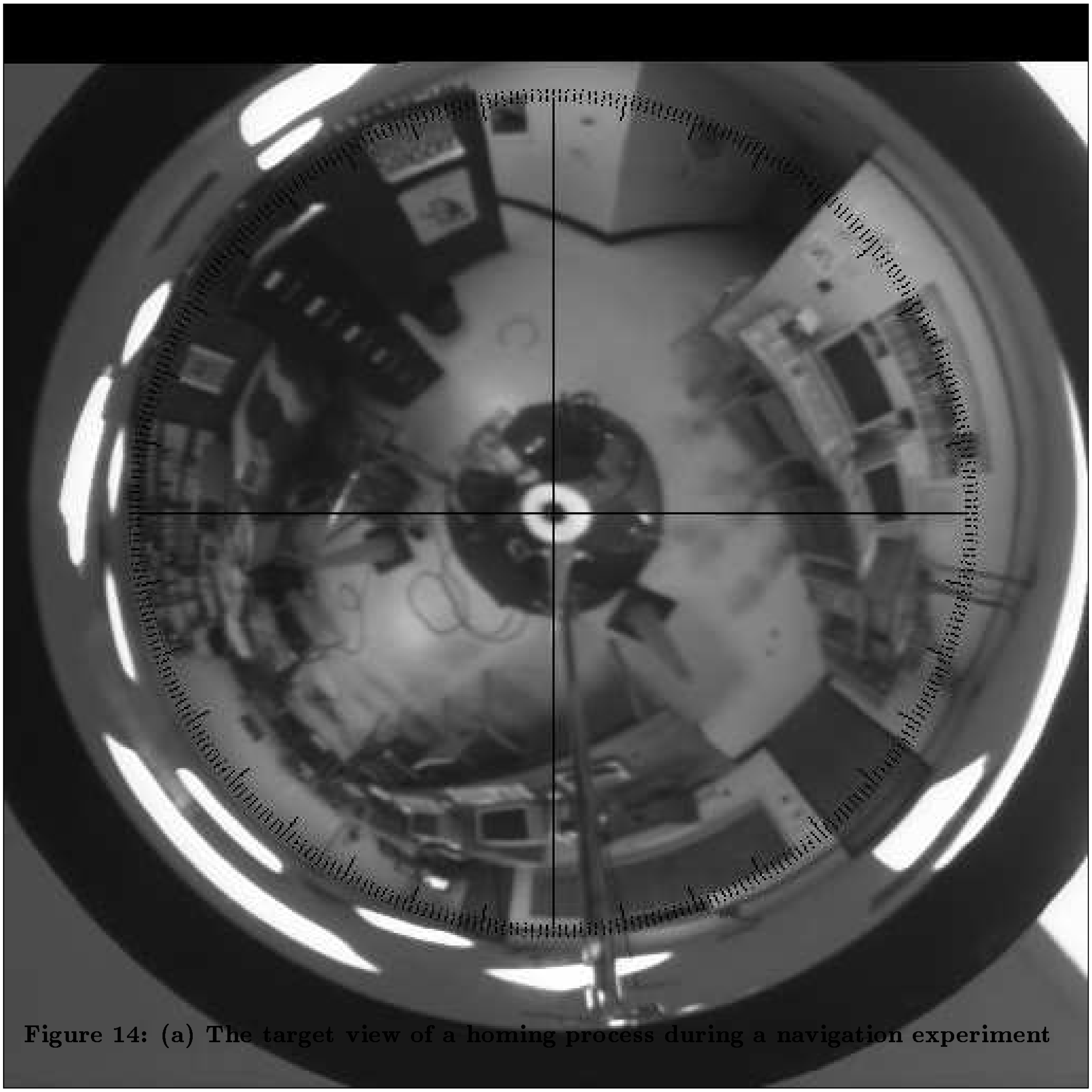

Figure 14: (a) The target view of a homing process during a navigation experiment

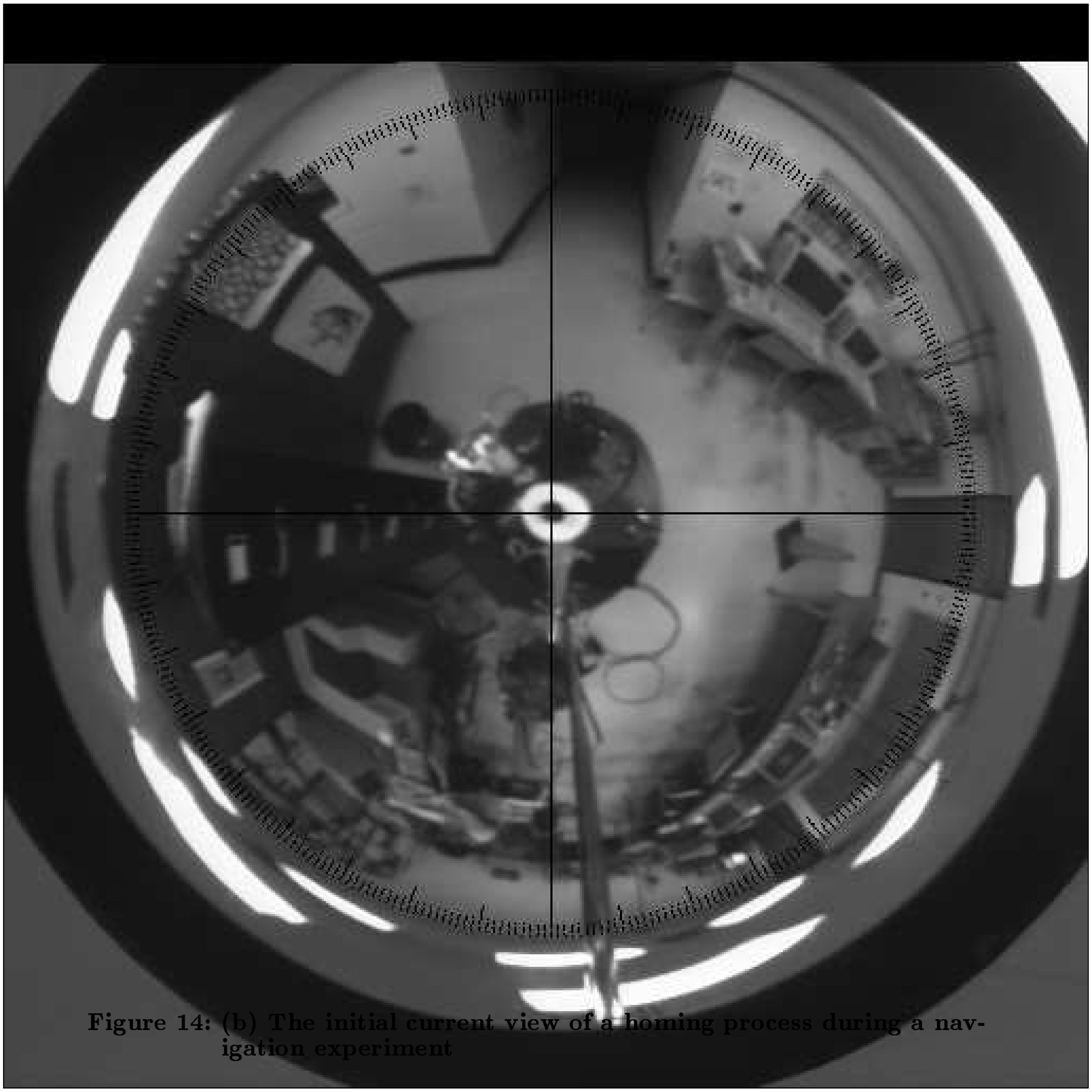

Figure 14: (b) The initial current view of a homing process during a navigation experiment

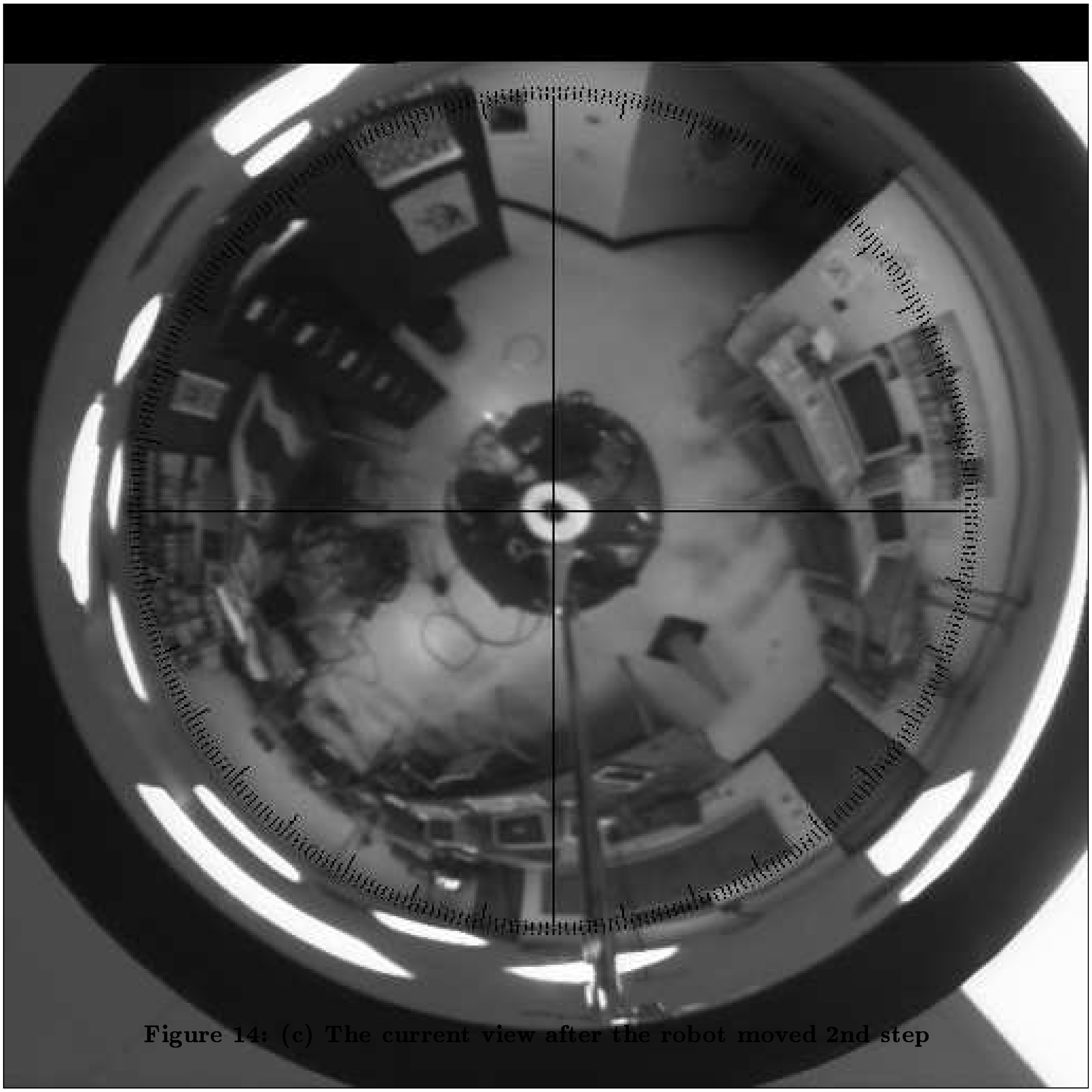

Figure 14: (c) The current view after the robot moved 2nd step

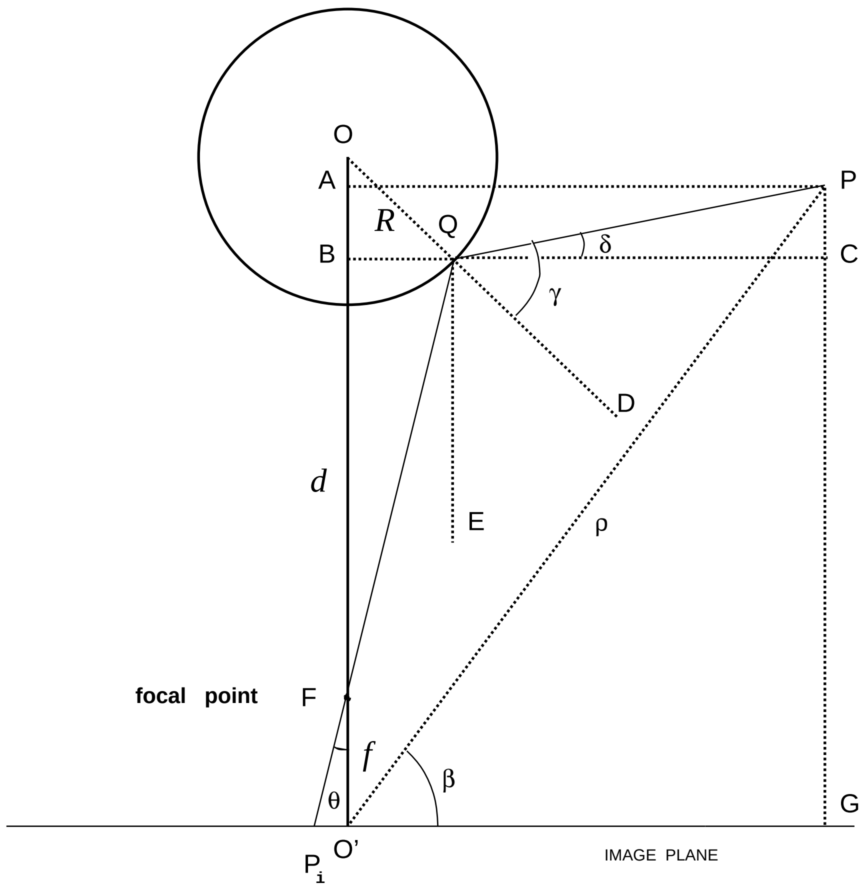

Figure 15: Derivations of the equations in Sec. 3.

Table 1 – Matching Result between the initial current view and the target view
The average displacement: $29.9°$

Target C.F. Azimuth Angle	Current C.F. Azimuth Angle	Matching Correct?
100.0	128.0	y
47.5	70.0	y
78.0	103.0	y
93.5	124.0	y
6.0	32.5	y
104.5	132.5	y
305.5	344.5	y
116.5	147.0	y
182.0	221.0	n

Table 2 – Matching Result after 1st Rotation
The average displacement: $8.7°$

Target C.F. Azimuth Angle	Current C.F. Azimuth Angle	Matching Correct?
291.0	303.5	y
310.5	320.5	y
326.5	333.5	y
305.5	316.5	y
335.0	341.5	y
333.0	338.5	y

Table 3 – Matching Result after 2nd Rotation
The average displacement: $-3.2°$

Target C.F. Azimuth Angle	Current C.F. Azimuth Angle	Matching Correct?
100.0	91.0	y
310.5	312.0	y
326.5	325.5	y
93.5	87.0	y
6.0	3.5	n
104.5	95.5	y
305.5	308.0	y
335.0	333.0	y
333.0	330.0	n

Table 4 – Matching Result after 1st Moving Step
The average displacement: $-2.5°$

Target C.F. Azimuth Angle	Current C.F. Azimuth Angle	Matching Correct?
100.0	93.5	y
310.5	310.5	y
78.0	67.5	y
326.5	324.5	y
93.5	87.5	y
256.0	260.5	y
305.5	306.5	y
116.5	113.0	y
335.0	332.0	y
333.0	329.5	n
295.0	297.5	y

Table 5 – Matching Result after 2nd Moving Step
The average displacement: $-0.9°$

Target C.F. Azimuth Angle	Current C.F. Azimuth Angle	Matching Correct?
100.0	100.5	y
291.0	292.0	y
310.5	310.5	y
78.0	77.0	y
326.5	325.5	y
93.5	93.5	y
6.0	2.5	y
104.5	105.0	y
305.5	305.5	y
11.0	7.5	y
38.0	34.0	y
295.0	295.0	y

Table 6 – Results of Nelson and Aloimonos' Algorithm

Motion Flow Data	Estimation of Rotation	Estimation of Translation Direction
Table 1	32°	190°
Table 2	10°	143°