IMAGE-BASED HOMING

Jiawei Hong   Xiaonan Tan
Brian Pinette   Richard Weiss
Edward M. Riseman

COINS Technical Report 91-15

February 1991

Abstract

We describe a system that allows a robot to acquire a model of its environment and to use this model to navigate. Our system maps the environment as a set of snapshots of the world taken at target locations. The robot uses an image-based local homing algorithm to navigate between neighboring target locations. Interesting and novel features of our approach are an imaging system that acquires a compact, 360° representation of the environment and an image-based, qualitative homing algorithm that allows the robot to navigate without explicitly inferring three-dimensional structure from the image. We describe the results of an experiment in a typical indoor environment and argue that image-based navigation is a feasible alternative to approaches using three-dimensional models.

---

1. Introduction

Mobile robot navigation has proved to be a complex task, even in a task domain where the robot is given a detailed three-dimensional model of its environment (Fennema et al [2]). Providing a robot with such a model is itself a significant, time-consuming task: a survey of many of the natural and cultural objects in the robot's environment and their spatial relationships to each other is required. If done sparsely this might amount to the extraction of key landmarks that would allow proper navigation relative to the prominent features of the visible environment. In the limit, this would involve determining contour maps and full three-dimensional solid models of all prominent objects. In either case, acquiring accurate geometric information is difficult and expensive.

Because it is difficult to acquire world models for navigation, it becomes an obvious goal to have a mobile robot explore the environment itself in order to extract information sufficient for effective navigation (e.g., Davis [1] or Yeap [8]). There are many possible ways this might be done, but if the goal is navigation — instead of full three-dimensional surface reconstruction — then the environmental features that are most prominent and visible (i.e., landmarks) will provide the key information for locating the robot vehicle and determining an appropriate path to the goal. Recent efforts involving navigation with landmarks include Fennema et al [2], Kumar and Hanson [5], and Zheng and Tsuji [9].

In this paper we develop a navigation strategy built on the technique of 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. The robot is capable of finding its way only to these target locations, but not to any arbitrary place in its environment. In contrast, such tasks as "Go down Elm St. until you come to a big white house with a poplar tree in front" or "Move three meters north" are not homing tasks: they require the robot to move to unfamiliar locations.

We use a novel and powerful imaging system to project a full 360° view of the world into a single image and then condense this view into a compact, one-dimensional *location signature*. A location signature retains enough information about the landmarks seen from its target location to allow homing. In image-based local homing, the differences between the signature of a robot's current location and the signature of a target location are used to compute incremental movements that take the robot closer to the target location. We call our technique "local" homing because the robot's current location must be close enough to the target location that it falls within its "capture region" for homing. If the robot's current location is too far from the target location, the homing algorithm will fail because there will be too much distortion in images of the prominent landmarks common to both location signatures.

We acquire a model of the world by running the robot along a desired route and having the system extract location signatures for a sequence of target locations on the route. After acquiring this model, the robot can navigate the route by successively homing to each of its target points. Thus we have reduced the problem of navigating a large-scale space to a problem of navigating a sequence of small-scale spaces.

The remainder of this paper will provide the details of matching landmark features and computing local homing movements for navigation. We will also summarize the results of indoor experiments that acquire environmental information in a training phase and then use this information for successful navigation.

2. Image Processing

We have implemented the homing algorithm on a Denning DRV-1 mobile robot. The imaging system (Figure 1), which is mounted to the front of the robot's body, comprises a spherical mirror mounted above a video camera. The video camera points up at the bottom of the spherical mirror and sees a 360° "hemispherical" image of the world (e.g., Figure 2). Since the body of the robot does not rotate, the orientation of the perceptual frame of reference in these images changes very little as the robot translates through the world. This imaging technique is similar to the conical mirror and laser striping system of Jarvis and Byrne [4] and the passive conical-mirror imaging system of Yagi and Kawato [7], but different from other methods that rotate a horizontal camera to acquire a panoramic view (e.g, Zheng and Tsuji [9] and Suzuki and Arimoto [6]).

Figure 2 shows the world as the robot sees it. The annulus of tick marks in Figure 2 contains the robot's horizon line. The horizon line is projected into the hemispherical image as a circle. As the robot moves horizontally across the ground plane, landmarks that project to points inside this horizon circle can potentially move to any location inside the circle; landmarks that project to points outside the horizon circle can potentially move to any location outside the circle. Landmarks that project to points on the horizon circle, however, remain on this circle as the robot moves. By sampling the image on the horizon line, we take advantage of a projective invariance that allows us to reduce the complexity of the landmark matching problem from two dimensions to one dimension.

The robot extracts a one-dimensional, circular location signature by sampling the hemispherical image along the horizon circle at angular intervals $\Delta\theta$; in our experiments, we sampled at 1° intervals. Each sample is a radial average of the image near the horizon circle; in our experiments, we average over 5 radial pixels. We can formally express the re-
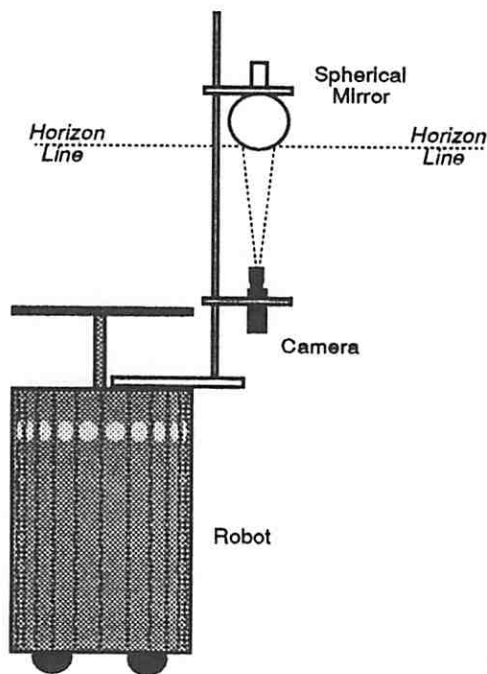
Figure 1: Robot with imaging system.

lation between the one-dimensional location signature V and the two-dimensional image I as

$$V_i = \sum_{j=-2}^{2} I(i\Delta\theta, r_h + j\Delta r),$$

where V_i is the ith intensity value of a one-dimensional location signature V, $I(\theta, r)$ is the intensity of the hemispherical image at polar coordinates (θ, r), $\Delta\theta$ is the angular sampling interval, Δr is the radial sampling interval and r_h is the radius of the horizon circle. Thus we compress a 512 by 512-byte image into a 360-byte location signature. This efficient representation of images is of major importance in the simple yet effective development of the homing algorithm presented in later sections. Figure 3 shows a graph of a typical location signature.

Let us now examine how prominent world features, i.e., landmarks, are selected from the location signature. We call these features *characteristic points*. Characteristic points of signatures are found in three steps. On the first step the location signature gets segmented into regions of monotonically increasing or decreasing intensity. On the second step the point of maximum instantaneous intensity change (i.e., zero-crossing) in each segment is
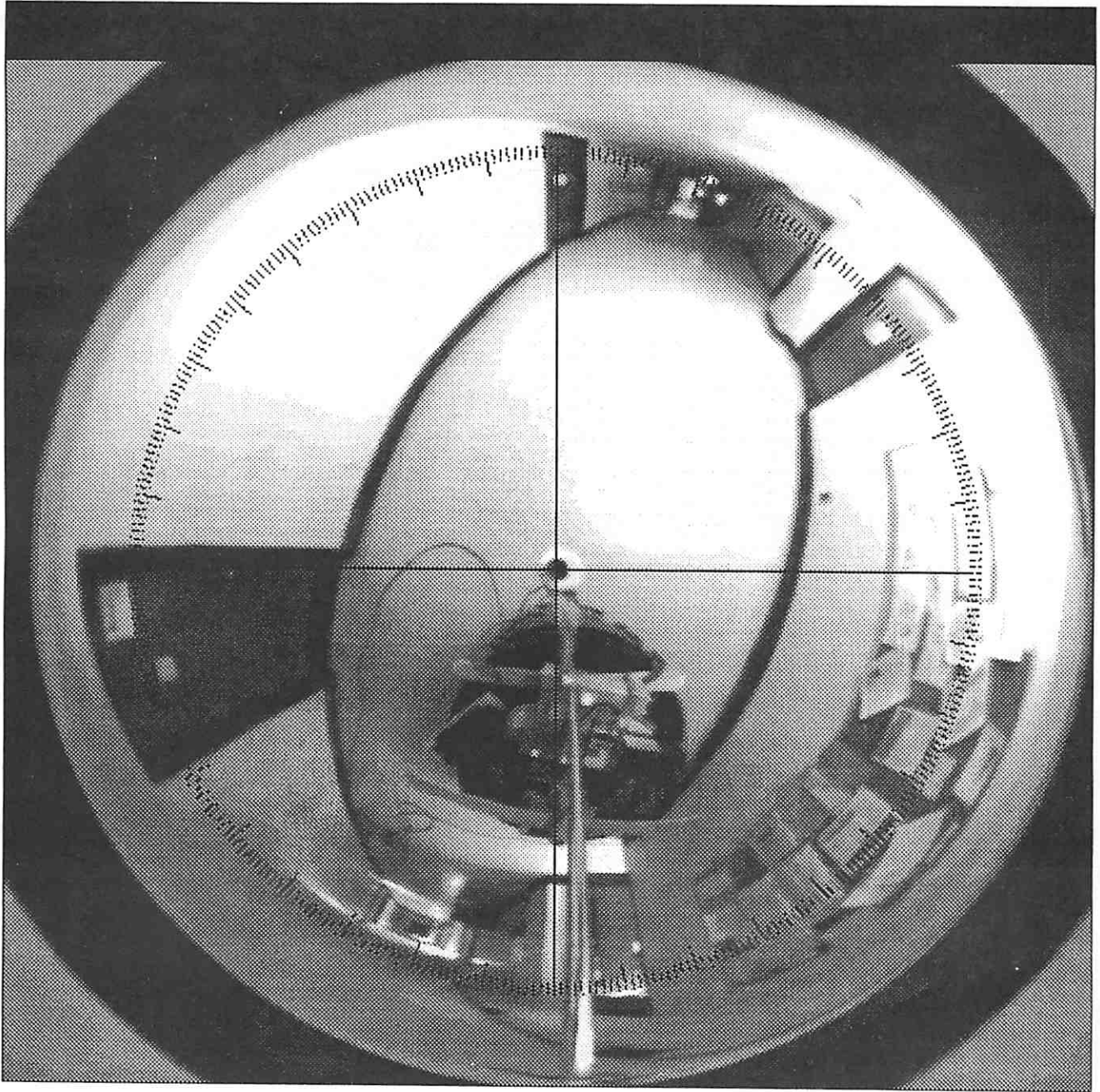
4

Figure 2: Hemispherical view of corridor. The superimposed circle of tick marks shows the portion of the image sampled to create a location signature. The horizon circle of robot lies within this annulus.

found. Such a point is accepted as a potential characteristic point if it represents a large enough instantaneous intensity change or if the total intensity change across its segment is large enough. On the third step these potential characteristic points are ranked, and the top fifteen are selected as those image features that represent the most prominent landmarks. The rank of a potential characteristic point i is given by $S_i \Delta_i$, where Δ_i is the total intensity change across its segment and S_i is its *sparseness*. The sparseness of a characteristic point is the distance between the potential characteristic points on its immediate right and left.

3. Matching Location Signatures

The goal of the matching step is to find a set of correspondences between the characteristic points in the signature V of the current location and the points in the signature V^T of the target location. Since our correlation function computes the *difference* between two signatures, the best set of correspondences is the one that minimizes the sum of a set of correlation values ρ_{ij}. In matching a point i in V against a point j in V^T, we are actually matching the values in windows centered around those points. Matching is performed with a normalized correlation function that uses the mean μ of the intensity in a window and an approximation σ to the standard deviation of the intensity in a window to normalize the matching. The normalization compensates for illumination changes that might occur between the time a location signature is acquired and the time it is used as a target. We assume that the difference between a target signature V^T for a location and a current signature V for the same location can be expressed as some affine transformation of the brightness profile, i.e.,

$$V^T = s(V - k)$$

for some constant bias k and scale factor s. In other words, the current signature for a location can be transformed into the target signature for the same location by removing some constant bias and then multiplying every intensity value by some scale factor.

We find it more convenient to express the affine transform as

$$s^T(V^T - k^T) = s(V - k).$$

This is equivalent to the previous expression for an affine transform, except that two scale factors, s^T and s, and two biases, k^T and k, must be determined. We estimate the constant biases to be the average intensity in a window and the scale factors to be reciprocals of the

mean absolute standard deviation (which uses the L_1 norm instead of the L_2 norm). That is,

$$k = \mu = \frac{\sum_\ell w_\ell V_{i+\ell}}{\sum_\ell w_\ell} \qquad k^T = \mu^T = \frac{\sum_\ell w_\ell V_{i+\ell}^T}{\sum_\ell w_\ell}$$

$$s = \frac{1}{\sigma} = \frac{\sum_\ell w_\ell}{\sum_\ell w_\ell |V_{i+\ell} - \mu|} \qquad s^T = \frac{1}{\sigma^T} = \frac{\sum_\ell w_\ell}{\sum_\ell w_\ell |V_{i+\ell}^T - \mu^T|},$$

where $V_{i+\ell}$ is the image intensity of the current location signature at index $i + \ell$ (i.e., in the window centered at index i in V), ℓ is an index into a fixed-size window, and w_ℓ is a harmonic weighting function, defined as $w_\ell = 1/|2 + \ell|$, that gives more weight to the center of the window than to the edges. In our system, the window into the location signature is 13 pixels wide. Note that this transformation compensates only for variations in brightness; it does not compensate for spherical distortions induced in the images of landmarks as the robot moves.
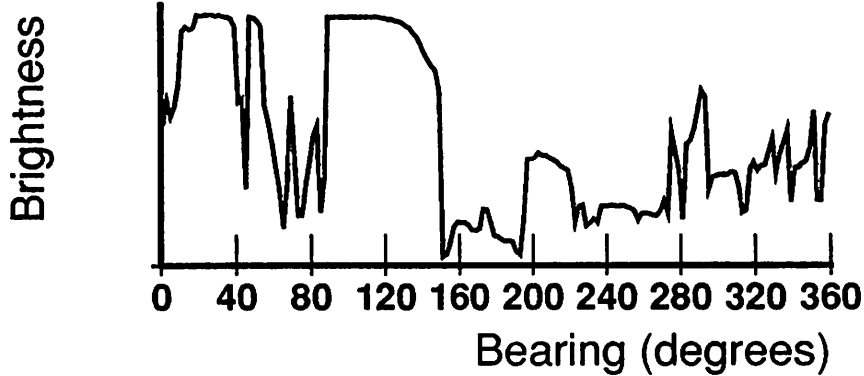

Figure 3: A typical location signature.

We match the characteristic points in the current signature V against the image points (not necessarily characteristic points) in the target signature V^T. The ideal form of our matching function would

$$\rho_{i,j} = \sum_\ell w_\ell |(V_{i+\ell} - \mu) - \frac{\sigma}{\sigma^T}(V_{j+\ell}^T - \mu^T)|,$$

where $\rho_{i,j}$ is the match value at current location signature index i and target location signature index j. We have refined this ideal matching function in several ways.

First, we have changed the shape of the matching function by multiplying it by the ratio $\frac{k_{high}}{k_{low} + \mu^T}$. The constants k_{high} and k_{low} are chosen to give better dynamic range of matching values. The effect of this refinement is to de-emphasize large differences between the target and current images when the average intensity in the target image is large.

Next, we remove any (unintentional) rotation ψ of the robot's body that may have occurred since the time the target signature was acquired. We call this rotation the *deviation*. The algorithm assumes that this deviation will be small. In practice, it is estimated to be less than 3°.

The initial estimate ψ_{est} of the deviation is computed by

$$\psi_{est} = \frac{\sum_i \phi_i S_i \rho_{i,j_{best}} \sin^2(\theta_i)}{\sum_i S_i \rho_{i,j_{best}} \sin^2(\theta_i)},$$

where S_i is the sparseness of the ith characteristic point, ϕ_i is its offset, θ_i is its the angular bearing, and $\rho_{i,j_{best}}$ is the best match value found for the ith characteristic point. Offset ϕ_i is computed by $\theta_i^T - \theta_i + \psi_{est}'$, where θ_i^T is the bearing of the landmark in the target signature and ψ_{est}' is the most recent estimate of the deviation (initially 0). The initial estimate of the target direction ω^0 is also made (using the algorithm described in the following section). Once these initial estimates are found, the set of deviations in the range $[\psi^0 - k_\psi, \psi^0 + k_\psi]$ and the set of directions in the range $[\omega^0 - k_\omega, \omega^0 + k_\omega]$ are searched for a deviation and a direction that minimizes $\sum_i \rho_{i,j_{best}}$. The parameters k_ψ and k_ω are empirically determined. In our system k_ω is a constant 56° but k_ψ may become as large as 180°.

For each new value of deviation ψ_{est} and direction ω_{est}, the robot tries matching the ith characteristic point in the current signature view V to each point in the interval between $\psi_{est} + \theta_i - k_{front} \sin(\theta_i - \omega_{est})$ and $\psi_{est} + \theta_i + k_{back} \sin(\theta_i - \omega_{est})$ in model signature view V^T, where $\theta_i - \omega_{est}$ is the bearing of ith characteristic point relative to the current estimate ω_{est} of the direction towards home. The choice of k_{front} and k_{back} helps determine how far towards its front and its back (where its front is in the estimated direction of home, ω) that the robot searches for matches to characteristic points. In our system $k_{front} = 0°$ and $k_{back} \approx 8°$.

The actual matching function, reshaped and with the deviation removed, is given by

$$\rho_{i,j} = \sum_\ell \frac{k_{high}}{k_{low} + \mu^T} w_\ell |(V_{i+\ell} - \mu) - \frac{\sigma}{\sigma^T}(V_{j+\ell-\psi}^T - \mu^T)|,$$

where the mean and reciprocal of the standard deviation in the target window are now given by

$$\mu^T = \frac{\sum_\ell w_\ell V_{i+\ell+\psi}^T}{\sum_\ell w_\ell}$$

$$\frac{1}{\sigma^T} = \frac{\sum_\ell w_\ell}{\sum_\ell w_\ell |V_{i+\ell+\psi}^T - \mu^T|}.$$

The values μ and $\frac{1}{\sigma}$ remain the same as before.

4. Computing a Movement

The difference between the bearing of a characteristic point in the signature for the robot's current location and bearing of the point it matches in the target signature is the *offset* ϕ of the characteristic point. The offsets for the characteristic points allow the algorithm to compute an incremental local homing movement. Suppose the robot is at some current position C and its goal is target position T, as shown in Figure 4. What would the robot see if it were to move directly towards target location T? The spherical mirror induces very simple image displacements of landmarks that project onto the horizon line. Every characteristic point on the invariant horizon circle slides along the horizon circle away from the robot's direction of motion in the direction shown by the curved arrows in Figure 4. Thus the robot should move in a direction that will cause the characteristic points to slide to the bearing they have at the target location.

Let us consider landmark ℓ_X in Figure 5. The offset between A in the current location signature and its matching point ℓ_X^T in the target location signature is ϕ_X. If the robot were at the target location, this offset would be 0. Our strategy is to move the robot in in a way that most quickly decreases this offset — a direction ω_X perpendicular to the bearing θ_X of characteristic point X in the current location signature (Figure 5). By similar reasoning, the fastest way to reduce the offsets ϕ_Y and ϕ_Z of characteristic points ℓ_Y and ℓ_Z is to move in directions ω_Y and ω_Z, respectively. Figure 5 shows how the homing vectors ω_Y and ω_Z (shown as thin dotted lines) add to the homing vector ω_X (shown as a thick solid line) to form the final homing vector $\omega_{est.}$ (shown as a thick dotted line). The final direction vector $\omega_{est.}$, a sum of individual homing vectors, points in the approximate direction of the target location T; the true direction towards T is ω_{actual} (shown as a thick dotted line).

This final direction $\omega_{est.}$ is computed by

$$\text{atan}\left(\frac{\sum_i -\text{sgn}(\phi_i - \psi_{est})\sin(90° + \theta_i + \psi_{est})}{\sum_i -\text{sgn}(\phi_i - \psi_{est})\cos(90° + \theta_i + \psi_{est})}\right)$$

where ϕ_i is the offset of the ith characteristic point, θ_i is its bearing, and ψ_{est} is the estimated deviation. We refine this estimate of the homing direction in three ways.

First, consider this subtlety: If we believe that the best way to get from the target location T to the current location Y is some direction ω^T, then we must believe that the best way to get from C to T is the opposite direction, $\omega^T - 180°$. If the robot were at the target location T and were trying to go to location O, the rôles of V^T and V would be switched: V^T would be the "current signature view" and V would be the "target signature view." In
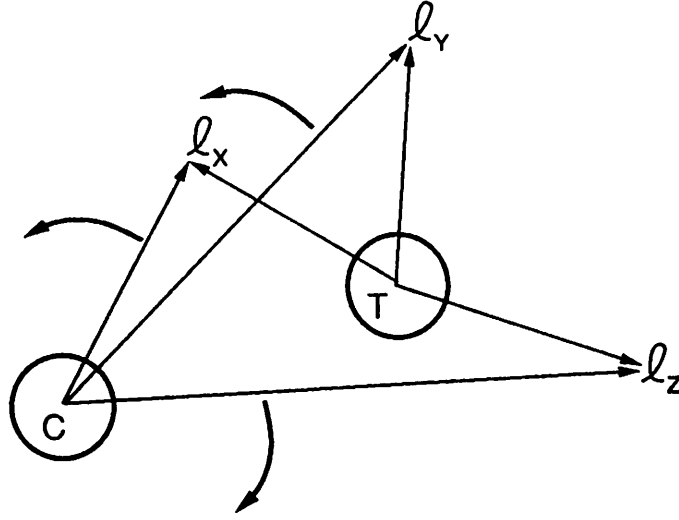
Figure 4: Homing problem with three landmarks.

this case the robot would believe that the fastest way to reduce the angle between some now-current landmark ℓ_X^T and its now-target landmark ℓ_X would be to move in a direction ω_X^T perpendicular to the bearing θ_X^T of landmark ℓ_X^T. In the same way that it originally computed the direction from C to T, the robot would use the formula above to compute a final direction ω^T from T to C. But this implies that the direction from C to T must be $\omega^T - 180°$, which in general is different from ω. Which direction, ω_X or $\omega_X^T - 180°$, should the robot move? We simply average these two directions and move in direction $(\omega_X + \omega_X^T - 180°)/2$. A homing movement ω that includes the contributions from the distribution of the target landmarks is given by

$$\text{atan}\left(\frac{\sum_i -\text{sgn}(\phi_i - \psi_{est})\sin(\phi_i/2 + 90° + \theta_i + \psi_{est})}{\sum_i -\text{sgn}(\phi_i - \psi_{est})\cos(\phi_i/2 + 90° + \theta_i + \psi_{est})}\right)$$

In a further refinement, we weight a characteristic point i by the term $|\phi_i - \psi|S_i/\rho_{i,jbest}$, giving a better estimate of ω:

$$\text{atan}\left(\frac{\sum_i -\frac{(\phi_i - \psi_{est})S_i}{\rho_{i,jbest}}\sin(\phi_i/2 + 90° + \theta_i + \psi_{est})}{\sum_i -\frac{(\phi_i - \psi_{est})S_i}{\rho_{i,jbest}}\cos(\phi_i/2 + 90° + \theta_i + \psi_{est})}\right)$$

As mentioned, we search a range of estimated deviations and homing directions to find a pair of values for which the current image best matches the target image. We use this
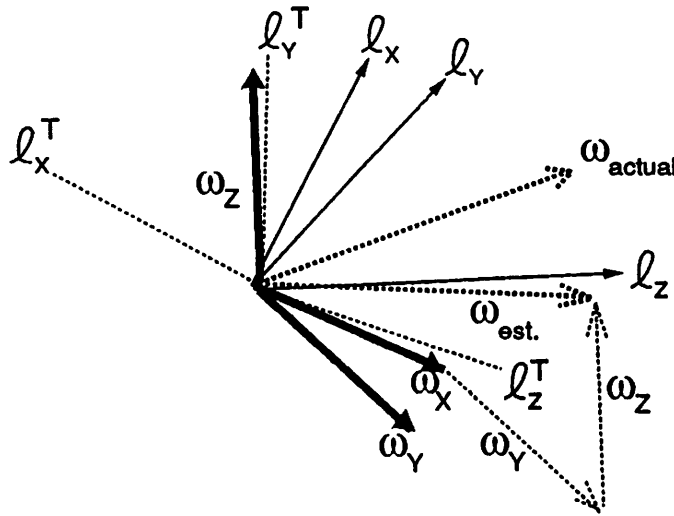
Figure 5: How a robot uses landmarks to estimate homing direction.

best set of correspondences with the above formula to compute an approximate direction ω towards the target location. We then average this value with the estimated homing direction ω_{est} that gave rise to this best set of correspondences to get a final homing direction,

$$\omega_{final} = (\omega + \omega_{est})/2$$

This is the final refinement to the computation of the homing direction.

As well as determining a direction to move, the algorithm must also choose the distance to move. For each target signature, the robot is moved 0.6 feet on the first incremental homing step, 0.4 feet on the second incremental homing step and 0.3 feet on subsequent incremental homing steps. The homing process for a given target location signature stops when the current direction the robot moves, $\omega(\tau)$, differs sufficently from the previous direction it moved, $\omega(\tau - 1)$. When this happens, it is assumed that the robot has overshot the target location. In our system the stopping criterion is $|\omega(\tau) - \omega(\tau - 1)| > 40°$. Of course, many other stopping criteria are possible, such as a threshold on the degree-of-match between current and target location signatures.

5.   Demonstration of the Homing Algorithm

We tested the algorithm by taking a sequence of hemispherical images in a hallway. There were 17 images in all, taken at target locations spaced about 1 foot apart. We placed the

11

robot 1 foot from the first target location; its goal was to home to each target location in sequence. The robot was able to traverse this path successfully and reach the final target location. We discuss the details of the experiment in Hong et al [3].

6. Summary and Discussion

We have argued that image-based landmark navigation is a feasible alternative to navigation approaches that maintain three-dimensional models of the world. This paper describes an approach that divides large-scale navigation tasks into a sequence of small-scale navigation tasks that are solved by local, image-based homing. Our homing algorithm uses compact location signatures acquired by a novel 360° imaging system. In addition, landmark information is acquired in a natural and straightforward way that does not involve acquiring three-dimensional information. We have described our image-based homing algorithm and have demonstrated it on a mobile robot for a typical short-range navigation task.

In future work we will be trying to improve the robustness of the homing algorithm and to extend its range. Ultimately, image-based local homing might be used to create a full-blown navigation system that can autonomously acquire a qualitative spatial map of its environment for robust, goal-oriented navigation.

Acknowledgements

References

[1] Ernest Davis.
    Representing and Acquiring Geographic Knowledge.
    PhD thesis, Yale University, 1984.

[2] Claude Fennema, Allen R. Hanson, Edward Riseman, J. Ross Beveridge and Rakesh Kumar.

Model-directed mobile robot navigation.
To appear in *IEEE Transactions on Systems, Man and Cybernetics.*

[3] Jiawei Hong, Xiaonan Tan, Brian Pinette, Richard Weiss and Edward M. Riseman.
Image-based Navigation Using 360° Views.
In *Image Understanding Workshop 1990*, pages 782-791. Morgan Kaufman, 2929 Campus Drive, San Mateo, California. 1990.

[4] R. A. Jarvis and J. C. Byrne.
An automated guided vehicle with map building and path finding capabilities.
In Robert C. Bolles and Bernard Roth, editors, *4th International Symposium on Robotics Research*, pages 497-504. MIT Press, Cambridge, Massachusetts. 1988.

[5] Rakesh Kumar and Allen R. Hanson.
Robust estimation of camera location and orientation from noisy data having outliers.
In *IEEE Computer Society Workshop on Interpretation of Three-dimensional Scenes*, pages 52-60.
IEEE Computer Society Press. 1989.

[6] Hisashi Suzuki and Suguru Arimoto.
Visual control of autonomous mobile robot based on self-organizing model for pattern learning.
Journal of Robotic Systems 5(5):453-470. 1988.

[7] Yasushi Yagi and Shinjiro Kawato.
Panorama scene analysis with conic projection.
In *IEEE International Workshop on Intelligent Robots and Systems*, pages 181-187. 1990.

[8] Wai K. Yeap.
Towards a computational theory of cognitive maps.
Artificial Intelligence 34:297-360, 1980.

[9] Jiang Yu Zheng and Saburo Tsuji.
Panoramic representations of scenes for route understanding.
In *Tenth International Conference on Pattern Recognition*, pages 161-167. IEEE Computer Society Press. 1990.