

Instantaneous Robot Self-Localisation and Motion Estimation with Omnidirectional Vision

Libor Spacek and Christopher Burbridge

*Department of Computer Science
University of Essex, Wivenhoe Park, Colchester, CO4 3SQ, UK*

You can obtain the May 2007 publication of this paper by JRAS (Elsevier) on doi:10.1016/j.robot.2007.05.009

Abstract

This paper presents two related methods for autonomous visual guidance of robots: localisation by trilateration, and inter-frame motion estimation. Both methods use co-axial omnidirectional stereopsis (omnistereo), which returns the range r to objects or guiding points detected in the images. The trilateration method achieves self-localisation using r from the three nearest objects at known positions.

The inter-frame motion estimation is more general, being able to use any features in an unknown environment. The guiding points are detected automatically on the basis of their perceptual significance and thus they need not have either special markings or be placed at known locations.

The inter-frame motion estimation does not require previous motion history, making it well suited for detecting acceleration (in 20^{th} of a second) and thus supporting dynamic models of robot's motion which will gain in importance when autonomous robots achieve useful speeds.

An initial estimate of the robot's rotation ω (the visual compass) is obtained from the angular optic flow in an omnidirectional image. A new non-iterative optic flow method has been developed for this purpose. Adding ω to all observed (robot relative) bearings θ gives true bearings towards objects (relative to a fixed coordinate frame).

The rotation ω and the r, θ coordinates obtained at two frames for a single fixed point at unknown location are sufficient to estimate the translation of the robot. However, a large number of guiding points are typically detected and matched in most real images. Each such point provides a solution for the robot's translation. The solutions are combined by a robust clustering algorithm *Clumat* that reduces rotation and translation errors.

Simulator experiments are included for all the presented methods. Real images obtained from *ScitosG5* autonomously moving robot were used to test the inter-frame rotation and to show that the presented vision methods are applicable to real images in real robotics scenarios.

Key words: self-localisation, motion estimation, omnidirectional vision, omnistereo, omniflow

1. Introduction

Amongst all robot sensors available, vision provides the most information, at the greatest range and often with the greatest accuracy.

Navigation and environment sensing are essential

for autonomous mobile robots. The ability to quickly estimate position in an environment is often crucial. Omnidirectional vision offers detailed information about the entire surroundings and as such is ideally suited for use in robot localisation.

Omnidirectional vision sensors have been con-

structured in many different ways. Tan et al. (2004) use a pyramid of mirrors, and point multiple cameras at the pyramid. This configuration offers high resolution and the possibility of a single view point but is not isotropic and the registration and the physical arrangement of the cameras can be difficult. Rotating cameras and mirrors were used by Kang and Szeliski (1997) and Ishiguro et al. (1992). However, difficulties were encountered with the registration and the motion delay of the camera. Wide angle and fish eye lenses have been used by Swaminathan and Nayar (2000) and Shah and Aggarwal (1997). Satisfactory methods of autocalibration and rectification of the distortions of the true fish eye lenses have only recently been developed by Micusik and Pajdla (2006).

Catadioptric omnidirectional sensors use a mirror and a camera. The mirror is rotationally symmetrical and the camera points at the mirror along its rotational axis. One major advantage that this design has over some of the other omnidirectional sensors is that there is no need to wait for a moving camera or to synchronise multiple cameras. This means better suitability to dynamic environments and mobile robotics.

There are several types of mirror that can be used. We use the conical mirror. This has the benefit of producing no radial distortion or loss of radial resolution, as is produced by hyperbolic mirrors. However, conical mirrors produce multiple effective viewpoints, which was seen until recently as problematic. Spacek (2005) shows that a single effective viewpoint is not actually required for a correct perspective projection and image unwarping.

We use a simulator capable of accurately modelling omnidirectional vision using ray-tracing techniques (Burbridge and Spacek, 2006). This allows us to accurately compare our results to the ground truth. We created a simulated robot containing two vertically aligned catadioptric omnidirectional sensors. The ray tracing methods reconstruct the images, as seen through the mirrors, including their inherent pixellation and resolution errors.

We first deploy omnistereo for range finding. Then we use the distance to the objects in a simple landmark based localisation method, using triangulation from the three closest objects at known positions.

We then develop an optic flow method to estimate the rotation of the robot and, using it, we show how to calculate the motion of the robot between two frames from only a single arbitrary fixed point at an unknown location.

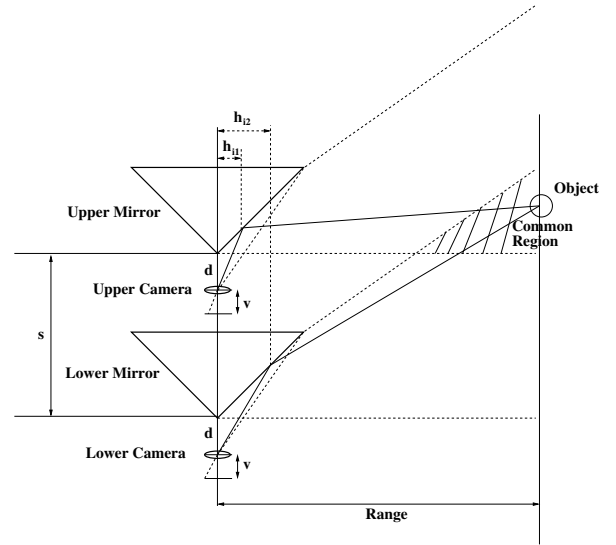


Fig. 1. Diagram showing the vertically aligned coaxial omnistereo configuration.

Finally, we demonstrate the extensibility of the developed methods to real robotics with experiments that use a real robot set up in the same way as the original simulated robot.

2. Omnistereo Range Finding

The range to objects r can be calculated using a pair of omnidirectional images in much the same way as in classical forward looking stereo vision. The formula for calculating the range (Spacek, 2005) is:

$$r = \frac{vs}{h_{i1} - h_{i2}} - d, \quad (1)$$

where d is the distance from the camera lens to the tip of the mirror, s is the distance between the mirrors, v is the distance of the image behind the lens of the camera and $h_{i1} - h_{i2}$ is the radial disparity between the imaged object positions in the two images. See figure 1. The distance to an object can only be computed if the object lies in the common region. The common region for our mirrors has the vertical field of view of 45° and the horizontal field of view of 360° , which compares very favourably with classical stereopsis.

To calibrate v and convert between pixels and mm, v is calculated as in (Spacek, 2005):

$$v = \left(\frac{d}{R} + 1\right)r_m, \quad (2)$$

where R is the radius of the mirror, and r_m is the radius of the mirror in the image.

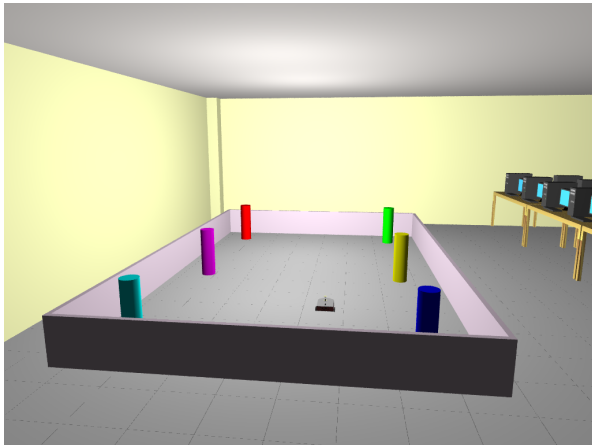


Fig. 2. The virtual lab. Six coloured pillars are positioned around the arena at known locations for landmark based robot localisation.

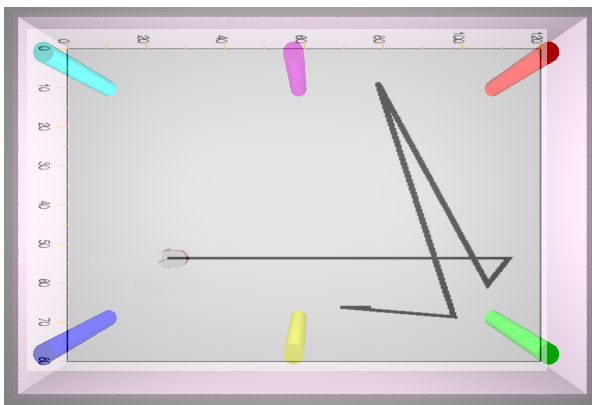


Fig. 3. The trajectory of the robot overlayed onto an overhead image of the arena. Measurements are in simulator units, where one unit is 50 mm.

These formulae apply for conical mirrors with 90° angle at their apex. Other mirrors do not produce a cylindrical projection. However, similar formulae can be obtained for all circularly symmetrical mirrors. The registration issues connected with aligning the line of sight to the mirror axis are discussed in (Spacek, 2005) and in (Geyer and Daniilidis, 2002).

2.1. Experiments

In our initial experiments we used the simulated arena of 6 m by 4 m as shown in figure 2. In all experiments reported in this paper the robots were configured for vertically aligned omnidirectional stereo as described here and shown in figure 1.

The robot was programmed with a random wandering, obstacle avoiding behaviour. This was run

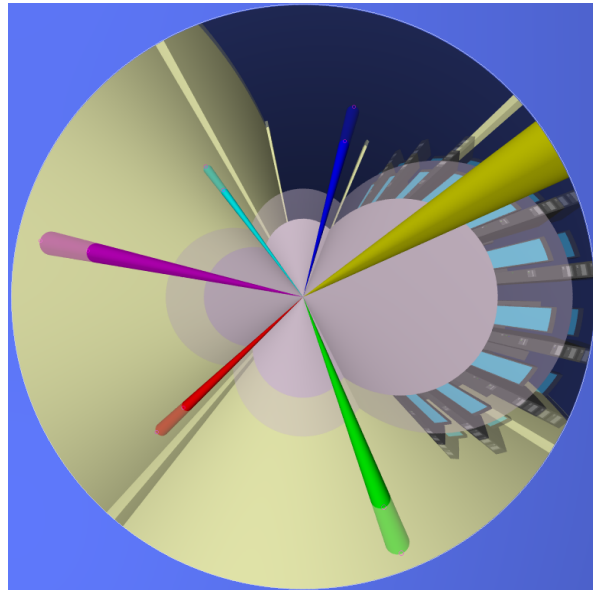


Fig. 4. The image of the upper mirror superimposed on the lower. The small circular markers show the points matched by the omnistereo algorithm.

for 450 frames and all images were saved alongside the actual robot trajectory. These frames were then processed to calculate the distance to the pillars and the estimated position of the robot in the arena.

In order to achieve obstacle avoidance a map of the arena was programmed into the robot motion controller. The controller drives the robot forwards, checking the map until it is at a pillar or a wall. It then randomly turns either left or right for a random duration and continues. The trajectory of the robot is shown in figure 3.

The radial disparity between upper and lower mirror can be seen in figure 4. The algorithm for measuring the disparity in these early experiments relies on colour matching along radial lines. Of course, for real scenes, more sophisticated correspondence matching is usually required.

The range to each of the six coloured pillars in each video frame was calculated by measuring the disparity of the top of the pillar between the corresponding upper and lower images, and applying equations 1 and 2 above. The table 1 shows the results. Graph of distance against frame number for the yellow pillar is presented in figure 5. When it is not possible to calculate the distance to an object, for example if it is too close or occluded, this is recorded on the graph as zero value. It is apparent from the graph that as the distance of the object increases, the absolute accuracy of the computed dis-

Table 1

A summary of the range results.

Pillar	Max. error	Mean error
Red	136.6 mm	21.95 mm
Green	124.8 mm	25.42 mm
Blue	256.9 mm	51.97 mm
Cyan	250.1 mm	62.93 mm
Magenta	110.5 mm	23.17 mm
Yellow	75.28 mm	20.14 mm
mean	159.03 mm	34.26 mm
s	76.05 mm	18.37 mm

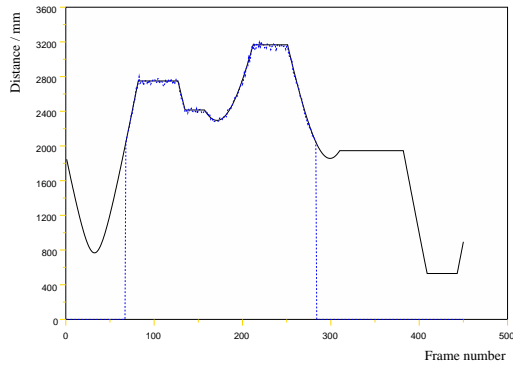


Fig. 5. Graph of actual distance to yellow pillar (solid line), and computed distance (dotted line).

tance decreases. This is also shown by the mean error for the cyan and yellow pillars being far greater than that of the others, these two pillars being the furthest from the robot. Also, as the top of the pillar gets closer to the centre of the mirror, the horizontal (angular) image resolution decreases.

The conclusion of this experiment is that the mean errors in the measured range to all objects were $0.86 \pm 0.46\%$ of the smaller arena dimension.

3. Self-Localisation using Trilateration

The calculated range of three nearest objects at known locations is used to fix the robot's position at the point of intersection of three circles around the objects, see figure 6. Three well separated objects are required for a unique solution. Since the ranges of objects contain some errors, a triangle of uncertainty will form. The centre of this triangle is recorded as the robot's estimated position.

The graph in figure 7 shows the error in the robot's

Robot situated in between intersections

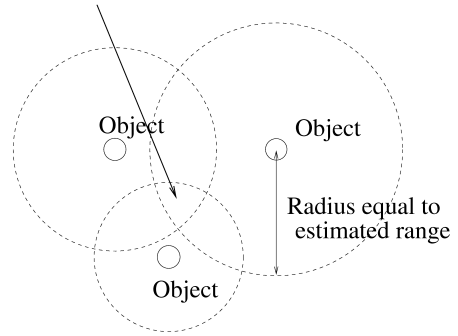


Fig. 6. Circles are drawn around three objects. The robot lies within the triangle of uncertainty.

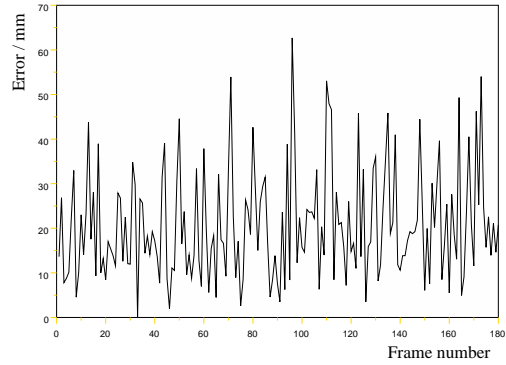


Fig. 7. Graph of the error in the predicted position of the robot.

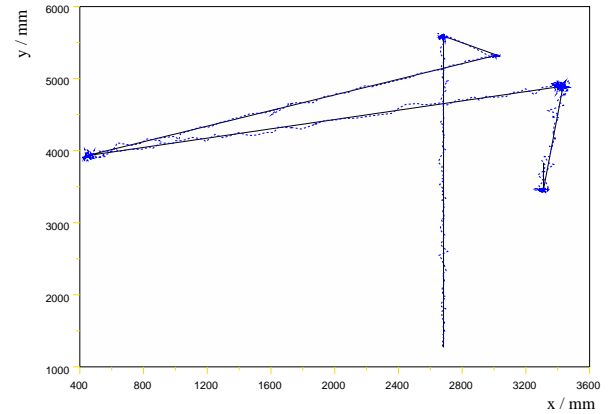


Fig. 8. Graph of the actual robot trajectory and the computed robot trajectory.

localisation for the first 180 frames. The actual trajectory and the calculated trajectory of the robot are shown in figure 8.

The accuracy of the trilateration depends entirely on the accuracy of the measured range. Therefore

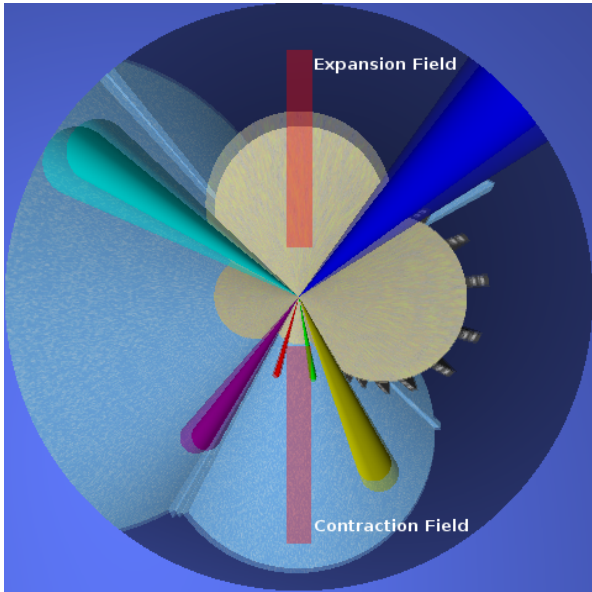


Fig. 9. The image sectors used to calculate the optic flow shown on two superimposed successive frames.

the errors are again of the order of 1% of the objects' range. By autonomously choosing the nearest objects, and therefore the most accurate measurements, the robot was able to calculate its location with the maximum error of 130 mm, and the mean error of 29.97 mm, or 0.75% of the smaller of the arena dimensions.

4. Optic Flow

The optic flow is calculated directly (non iteratively) in the regions shown in figure 9 by calculating the flows in the directions of the edge gradients of two adjacent edge pixels in the image and combining them to calculate the true flow.

Let the image function be $f(x, y)$. Carry out edge finding with a first derivative edge finder capable of returning normalised image function gradients $\mathbf{g} = \nabla f(x, y)$ at all points of significant curvature and significant edge contrast (Spacek, 1986).

The Horn and Schunk (1981) optic flow, equation 3, is derived from the assumption that the grey level values of imaged points do not change as the points move across an image:

$$f(x, y, t) = f(x + dx, y + dy, t + dt)$$

Then using only the first order terms of the Taylor series gives:

$$-\frac{\partial f}{\partial t} \simeq \mathbf{v} \cdot \mathbf{g} \quad (3)$$

where $\mathbf{v} = (v_x, v_y) = (\frac{dx}{dt}, \frac{dy}{dt})$ is the optic flow vector to be found and \cdot denotes the scalar product of two vectors. Dividing both sides of equation 3 by $|\mathbf{g}|$ (the magnitude of \mathbf{g}), we obtain the magnitude v_g of the flow in the direction of the unit vector $\hat{\mathbf{g}}$:

$$v_g = \mathbf{v} \cdot \hat{\mathbf{g}} \simeq -\frac{\frac{\partial f}{\partial t}}{|\mathbf{g}|}. \quad (4)$$

Let two edge elements (edgels) with different orientations have unit gradient vectors $\hat{\mathbf{u}} = (u_x, u_y)$, $\hat{\mathbf{w}} = (w_x, w_y)$ respectively. Such edgels are typically found on opposite sides of a point \mathbf{c} of high curvature. Substituting these known unit gradients for $\hat{\mathbf{g}}$ in equation 4, we get:

$$v_u = \mathbf{v} \cdot \hat{\mathbf{u}}, \quad v_w = \mathbf{v} \cdot \hat{\mathbf{w}}, \quad (5)$$

where v_u and v_w are known constants, obtained by evaluating the right hand side of equation 4. Applying $\hat{\mathbf{u}}$ and $\hat{\mathbf{w}}$ at \mathbf{c} , we can thus estimate the optic flow vector \mathbf{v} at \mathbf{c} by solving equations 5 directly for v_x and v_y , provided that

$$d = u_x w_y - u_y w_x \neq 0$$

$$v_x = \frac{v_u w_y - v_w u_y}{d}, \quad v_y = \frac{v_w u_x - v_u w_x}{d} \quad (6)$$

5. Inter-Frame Robot Rotation

The idea of using omnidirectional images to detect the rotation (visual compass) and possibly also the translation of a robot has been investigated by Binding and Labrosse (2006) in the context of appearance based vision, whereby images are typically added up in the radial direction to obtain a simple overall measure of luminance. We take a different approach which concentrates on the positions and image velocities of fine image features. Our approach is in principle capable of greater accuracy and thus can be applied to inter-frame rotations. Omnidirectional vision again supports this method well, as the guiding points remain in view.

In order to create a more generally applicable localisation than the trilateration, the rotation of the robot needs to be separated from its translation. This can be achieved by using the optic flow, where the v_θ component of the flow vectors gives the approximate rotation.

The robot's rotation ω is estimated by averaging the angular flow components over the front and back regions of the omnidirectional image:

$$\omega = \left(\frac{v_\theta}{r} \right),$$

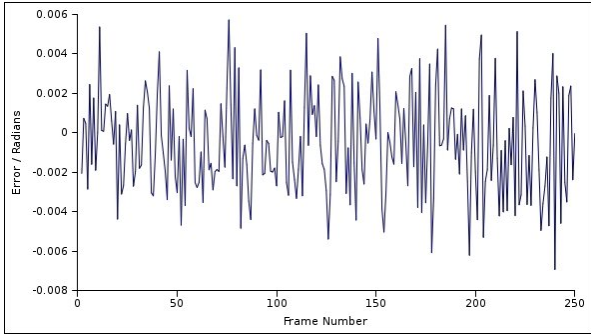


Fig. 10. The errors in estimated inter-frame rotation of the simulated robot.

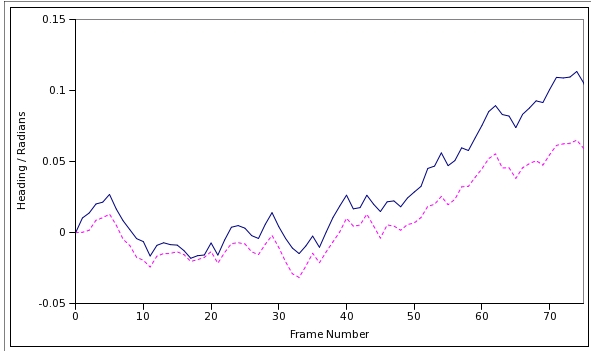


Fig. 11. The cumulative error in the simulated robot heading.

where the v_θ component of the optic flow vectors is found by converting \mathbf{v} to polar coordinates centred on the mirror. The reason for leaving out the side regions is that they are more affected by the translatory flows. The translatory flows on the sides appear as strong local v_θ flows with opposite signs. However, they cannot be simply cancelled out because they also depend on the distance of the various viewed objects.

5.1. Experiments

250 frames were generated at 20 frames per second, with the robot having random velocities for the left and right wheels. The rotational velocity was restricted to 20° per second, and the translational velocity was restricted to 50 cm/s . A graph of the error in the estimation of the rotation of the robot is shown in figure 10, and a graph showing the cumulative error of the robot heading based only on the visual compass is shown in figure 11. It is apparent that the optic flow method tends to systematically underestimate the robot rotation. Nevertheless, the Pearson product-moment correlation coefficient be-

tween the actual change in angle of the robot and the estimated change in angle over 250 frames was found to be 0.94, showing a very strong correlation.

6. Inter-Frame Robot Translation

Having estimated the rotation of the robot using optic flow, the translation can now be calculated as follows:

- At time step t , choose a guiding point in the lower catadioptric image.
- Calculate the range r_0 and the bearing angle θ_0 to the point by finding its location in the upper mirror and using the coaxial omnistereo as described earlier.
- At time step $t + 1$ calculate the new range to the point, r_1 and the new bearing angle θ_1 . See figure 12.
- Calculate the rotation of the robot ω between frames t and $t + 1$.
- Compute the translation (t_x, t_y) of the robot from equations 7 and 8:

$$t_x = x_0 - \cos(\omega) x_1 - \sin(\omega) y_1 \quad (7)$$

$$t_y = y_0 + \sin(\omega) x_1 - \cos(\omega) y_1 \quad (8)$$

These equations assume rotation about the z axis (vertical axis) only. The full 3D motion, using homogeneous coordinates, is given by:

$$\mathbf{p}_1 = \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \mathbf{p}_0, \quad (9)$$

where the column vectors \mathbf{p}_0 and \mathbf{p}_1 are the positions of the same guidance point in the two cartesian frames and r_{11} to r_{33} are the elements of a 3D rotation matrix. Solving this more general case necessitates estimating the rotations about each of the three axis (Euler's angles).

7. Clustering with Clumat

Although only one point is required in order to estimate the translation of the robot, by using multiple points, multiple solution of the translation can be found. These solutions are combined in order to acquire a more accurate estimate of both the translation and the rotation of the robot.

We developed a clustering algorithm *Clumat*, underpinned by the mathematical model that gives it

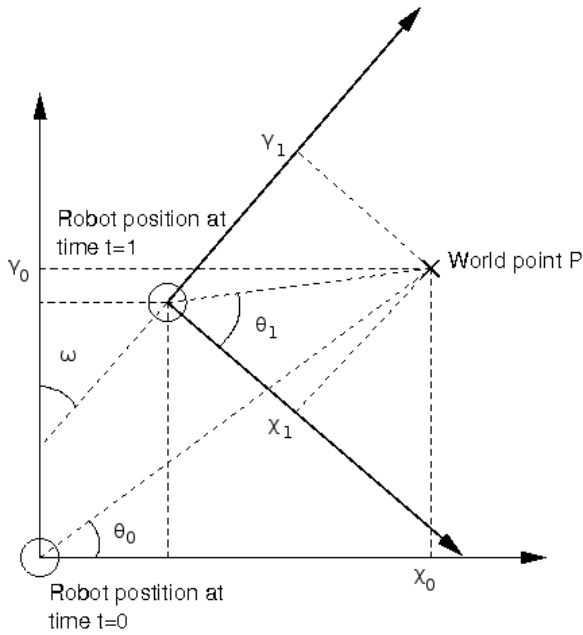


Fig. 12. The translation and rotation of the robot between two frames, $t=0$ and $t=1$.

fast convergence. The novel aspect of *Clumat* is that unlike ordinary k-means clustering or random sampling consensus (Ransac) algorithms, *Clumat* does not rely only on the statistical properties of the candidate solutions. The outliers are removed automatically as in ordinary k-means clustering but this is followed at each iteration by using the mathematical model, in this case equations 7 and 8, to repopulate the solutions space. In particular, having rejected some translatory outliers, we recompute the mean rotation of the robot by solving equations 7 and 8 for ω , and then recompute the individual translations for the remaining points. This is iterated until no more outliers beyond certain distance from the mean remain. The final mean rotation and translation are returned as the results.

In order to test *Clumat*, only the simple coloured pillars are again used as the guiding points to calculate the range. Crucially, we no longer need to know their locations in order to track the robot. In real environment applications any object points can be used as long as they can be reliably located in two frames.

In this simulation experiment, there are a maximum of six estimates for (t_x, t_y) . In real environments there will be several hundred, offering much better clustering opportunities for *Clumat*.

The motion estimation algorithm was tested on

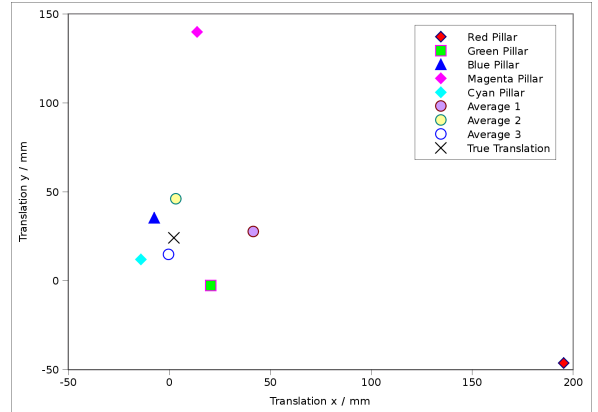


Fig. 13. The translations suggested by each pillar and the mean translation after two iterations of clustering.

each pair of frames generated for the previous rotation tests. The error was defined as the magnitude (the Euclidian norm) of the difference between the robot's true translation vector and the estimated translation vector. Figure 13 shows the inter-frame translations suggested by the pillars for one pair of frames and the mean translation after outliers were eliminated. The final translation error (9 mm) represents a significant improvement over the errors before the clustering, which are typically twice as large. Given more points, the accuracy should improve further.

8. Experiments with a Real Robot

To extend this work from simulation to real-world robotics, we mounted two 90° mirrors on a Scitos G5 mobile robot in the same arrangement as in our earlier simulation. This is shown in figure 14, and an example of the omnidirectional images produced is shown in figure 15. This robot was then used to perform rotation estimation experiments in the same way as in the previous simulated rotation estimation experiments, this time using the real Robot Arena at the University of Essex. The results for estimated robot heading compared to actual heading are similar to the results obtained in simulation. Errors in the rotation estimates are shown in figure 16. A graph of actual heading and calculated heading for the real robot experiment is shown in figure 17.

9. Conclusion

The use of omnidirectional stereo vision for range finding is clearly a definite possibility. The results



Fig. 14. DaX, the Scitos G5 mobile robot with two coaxial mounted conical mirrors and two firewire cameras.

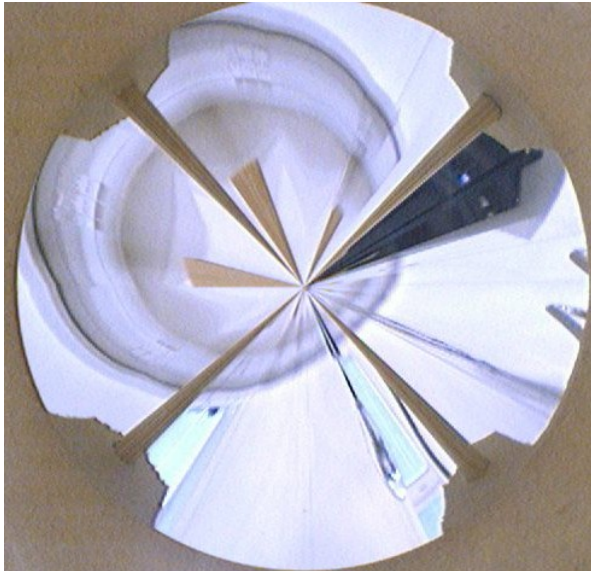


Fig. 15. An example of an omnidirectional image using a conical mirror on our mobile robot.

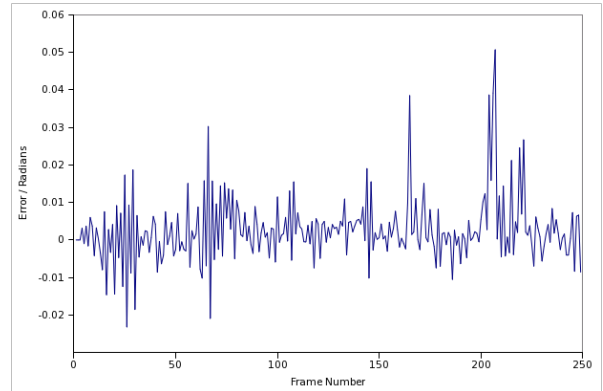


Fig. 16. The errors in estimated inter-frame rotation of the real robot.

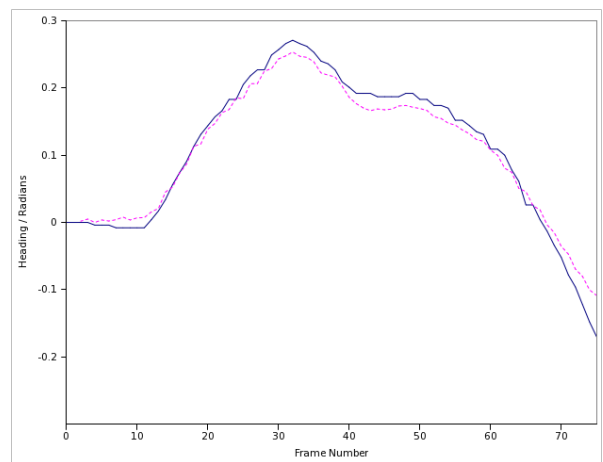


Fig. 17. The cumulative error in the robot heading using the real robot. The ground truth obtained by external VICON tracking system (solid line) versus the computed heading (dotted line).

show that the error in distance averages at about 2 cm for objects about 2 m away, i.e. about 1% of the distance. Different mirrors will cover different ranges. A 120° conical mirror, for example, will show the ground around the robot rather than objects in the distance. The accuracy achieved is generally better than that of sonar sensors. In many situations vision ranging is also more favourable than laser range finders as more information is provided – here the object is identified as well as its distance measured, avoiding problems such as perceptual aliasing.

Omnidirectional vision is ideal for robot localisation as objects do not go out of sight when turning. Using an accurate omnistereoo made fast localisation by simple trilateration possible. The speed of this method is limited only by the processing time – it could run at sub frame rates. It would not be realis-

tic to measure robot's motion occurring within 50 *ms* by GPS, for instance. Figure 7 shows that the error in position remained roughly constant. This result compares favourably to localisation by dead reckoning where the error accumulates with time.

Localisation experiments were performed using a simulator in order to make careful comparisons against the ground truth (the exact true translation and rotation of the robot). The use of the simulator instead of the real scenes made certain aspects of the localisation experiments less demanding, in particular the automatic selection and matching of the guidance points. On the other hand, there were fewer guidance points available, which limited the effectiveness of the *Clumat* clustering and hence the accuracy of the inter-frame translation results.

The trilateration localisation relied on three objects at known locations. Object positions can be fixed automatically by employing a simultaneous localisation and mapping scheme (Slam), such as the one proposed in (Rushant and Spacek, 1998), or others. Nevertheless, this restricting assumption was removed by the subsequent inter-frame rotation and translation methods, which yielded similar results.

All of these methods are equally applicable to longer time intervals and motions. We have chosen the inter-frame case as the most challenging.

In order to address the issue of real robotics, an inter-frame rotation experiment with a real robot and real images was performed, resulting in a close agreement with the equivalent rotation experiments on the simulator. This indicates that the remaining inter-frame translation methods may also be extended to real robots, which is to be our future work.

References

- Binding, D. and Labrosse, F. (2006). Visual local navigation using warped panoramic images. In *2006 Towards Autonomous Robotic Systems (TAROS'06)*, pages 19–26.
- Burbridge, C. and Spacek, L. (2006). Omnidirectional vision simulation and robot localisation. In *2006 Towards Autonomous Robotic Systems (TAROS'06)*, pages 37–44.
- Geyer, G. and Daniilidis, K. (2002). Paracatadioptric camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):1–10.
- Horn, B. K. P. and Schunk, B. G. (1981). Determining optical flow. *Artificial Intelligence*, 17:185–203.
- Ishiguro, H., Yamamoto, M., and Tsuji, S. (1992). Omni-directional stereo. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(2):257–262.
- Kang, S. B. and Szeliski, R. (1997). 3-d scene data recovery using omnidirectional multibaseline stereo. *Int. J. Comput. Vision*, 25(2):167–183.
- Micusik, B. and Pajdla, T. (2006). Structure from motion with wide circular field of view cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(7):1135–1149.
- Rushant, K. and Spacek, L. (1998). An autonomous vehicle navigation system using panoramic vision techniques. In *Proceedings of the International Symposium on Intelligent Robotics Systems, ISIRS*, pages 275–282. Tata McGraw-Hill.
- Shah, S. and Aggarwal, J. K. (1997). Mobile robot navigation and scene modeling using stereo fish-eye lens system. *Mach. Vision Appl.*, 10(4):159–173.
- Spacek, L. (1986). Edge detection and motion detection. *Image and Vision Computing*, 4(1):43–56.
- Spacek, L. (2005). A catadioptric sensor with multiple viewpoints. *Robotics and Autonomous Systems*, 51(1):3–15.
- Swaminathan, R. and Nayar, S. K. (2000). Non-metric calibration of wide-angle lenses and poly-cameras. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(10):1172–1178.
- Tan, K.-H., Hua, H., and Ahuja, N. (2004). Multiview panoramic cameras using mirror pyramids. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(7):941–946.