Fast Monocular Visual Compass for a Computationally Limited Robot

Peter Anderson and Bernhard Hengst

School of Computer Science and Engineering
University of New South Wales

Abstract. This paper introduces an extremely computationally inexpensive method for estimating monocular, feature-based, heading-only visual odometry - a visual compass. The method is shown to reduce the odometric uncertainty of an uncalibrated humanoid robot by 73%, while remaining robust to the presence of independently moving objects. High efficiency is achieved by exploiting the planar motion assumption in both the feature extraction process and in the pose estimation problem. On the relatively low powered Intel Atom processor this visual compass takes only 6.5ms per camera frame and was used effectively to assist localisation in the UNSW Standard Platform League entry in RoboCup 2012.

1 INTRODUCTION

Bipedal robots need to walk on a variety of floor surfaces that can cause them to slip to varying degrees while walking. They can also be bumped or impeded by undetected obstacles. Accurate navigation therefore requires an autonomous robot to estimate its own odometry, rather than assuming that motion commands will be executed perfectly.

If a robot is fitted with one or more cameras, visual odometry algorithms can be used to estimate the relative motion of the robot between subsequent camera images. However, these methods are typically computationally expensive for resource constrained robots operating in real time. To overcome this problem, this paper exploits the planar motion assumption in both the image feature extraction process, and in the calculation of relative pose estimates. The result is an extremely computationally inexpensive method for estimating heading-only visual odometry using a monocular camera and 1-dimensional (1D) SURF features.

This technique was developed and used in the RoboCup Standard Platform League (SPL) robot soccer competition, and implemented on an Aldebaran Nao v4 humanoid robot. The Nao is equipped with a 1.6 GHz Intel Atom processor and two 30 fps cameras. These cameras have minimal field of view overlap, necessitating the use of monocular visual odometry methods. As the Nao is not fitted with a vertical-axis gyroscope, heading odometry can only be estimated using visual methods.

On the soccer field, the combination of fast motion and collisions with other robots exacerbates the difficulties of biped navigation, making robot soccer an excellent test domain for this technique. However, the technique itself in no way relies on the characteristics of the robot soccer application domain.

Although the use of heading-only visual odometry has some limitations, practical experience suggests that when a bipedal robot manoeuvres, the greatest odometric inaccuracy is observed in the robot's heading, which can change very quickly when the robot slips, rather than in the forward or sideways components of the robot's motion. The visual heading odometry is therefore used in conjunction with the odometry generated by assuming perfect execution of motion commands (which we will refer to as command odometry). Results suggest that the combination of this technique with command odometry results in a dramatic improvement in the accuracy of odometry information provided to localisation filters. A further benefit of the visual odometry module is the ability to detect collisions and initiate appropriate avoidance behaviour.

The remainder of this paper is organised as follows: Section 2 outlines related work, Section 2.1 provides a brief introduction to 1D SURF image features, Section 3 describes calculation of heading odometry using 1D SURF features, and Section 4 presents experimental results.

2 BACKGROUND

If robots are fitted with multiple cameras with overlapping fields of view, stereo visual odometry algorithms can be used to recover the relative 6 degree of freedom (DoF) motion of the robot. This is typically achieved by measuring the 3D position of detected image features in each frame using triangulation [1]. A robot fitted with a single camera must use monocular visual odometry methods, which cannot solve the general 6 DoF relative motion estimation problem. Generally, in this case only heading information can be accurately obtained as the effect of translation is small compared to that of rotation, while the absolute scale of motion is unobservable.

To simplify the 6 DoF visual odometry problem for monocular cameras, a number of previous papers have used constraints on the motion of the agent. Approaches that use the planar motion assumption, such as [2], [3], typically rely on the detection of repeatable local image features that can be tracked over multiple frames. These features are then used to recover the relative robot pose using epipolar geometry; the assumption of planar motion on a flat surface makes relative pose estimation simpler and more efficient. Even greater efficiencies in pose estimation have been demonstrated when the planar motion assumption is coupled with the nonholonomic constraint of a wheeled vehicle [4] [5].

The planar motion assumption has also been used to develop monocular methods to determine the scale of motion [6]. These methods use the surface context approach developed by [7] to split a single image into three geometric regions: the ground, sky, and other vertical regions. Using this technique, scale can be determined based on the motion of features on the ground plane.

A unifying feature of these previous works is that although the planar motion constraint has been applied to the pose estimation problem, it has not been applied to the feature detection and extraction process. This process is computationally expensive, for example, using the OpenCV implementation, [8] demonstrated that SIFT and SURF feature extraction took on the order of 50 - 100 ms on a desktop PC for relatively small images. As a result, the most efficient pose estimation processes such as [4] have found that the overall frame-rate is limited by the feature extraction process, in this case using SIFT, Harris, and KLT image features. The computational cost of these feature extraction methods is prohibitively expensive for resource constrained robots such as the Nao.

Although impressive results have been reported using visual odometry in many application areas, the use of visual odometry in dynamic environments containing many independently moving objects remains challenging. Typical approaches to feature-based relative motion estimation are sensitive to wrong feature matches or feature matches on moving objects, even with the use of RANSAC based outlier rejection schemes [9]. To overcome these issues in dynamic environments, authors such as [9] have used image patch classification to improve rejection of independently moving objects such as other cars.

2.1 1D SURF FEATURES

1D SURF is an optimised feature detector designed to exploit the planar motion constraint. We implemented the algorithm developed by authors in [10] for fast mobile robot scene recognition. It consists of a modified one dimensional variant of the SURF [11] algorithm. As shown in Figure 1, the algorithm processes a single row of grey-scale pixels captured from a 30 pixel horizontal band at the robot's camera level (the robot's horizon). The horizon band is chosen for feature extraction because, for a robot moving on a planar surface, the identified features cannot rotate or move vertically. For a humanoid robot, the position of the horizon in the image is determined by reading the robot's limb position sensors and calculating the forward kinematic chain from the foot to the camera, or by using an IMU.

As shown in [10], the use of a 1D horizon image and other optimisations dramatically reduces the computational expense of SURF feature extraction, exploits the planar nature of the robot's movement, and still provides acceptable repeatability of the features. Consistent with the original SURF algorithm, the extracted features are robust to lighting changes, scale changes, and small changes in viewing angle or to the scene itself. On a 2.4GHz Core 2 Duo laptop 1D SURF runs more than one thousand times faster than SURF, achieving submillisecond performance. This makes the method suitable for visual navigation of resource constrained mobile robots; on the Nao v4 we find the mean extraction time of 1D SURF features is 2 ms.

Fig. 1. Image captured by the Nao robot showing superimposed 30 pixel horizon band in red, and the extracted grey-scale horizon pixels used for 1D SURF feature extraction at the top of the image.

3 RELATIVE POSE ESTIMATION

The 1D SURF visual odometry algorithm is of the monocular, feature-based, heading-only variety. Although it is possible to estimate the scale of camera motion using a monocular system as in [6], 1D SURF features are found only on the robot's horizon, not on the ground plane, and therefore do not lend themselves to these techniques. The error in the heading component of the robot's command odometry has a much greater effect on localisation accuracy than the errors in the forward and sideways components.

The soccer field is a dynamic environment with multiple independently moving objects in the form of other robots, referees, and spectators in the background, as illustrated in Figure 2. To help prevent the movement of other robots on the field from influencing the visual odometry, in this application domain a visual robot detection system is used to discard features that are part of other robots. This system uses region-building techniques on a colour classified image to detect other robots. It is similar in spirit to the feature classification approach used by [9]. However, the system is also robust to the movement of undetected robots and referees in front of the camera, as described further below.

To estimate the relative heading motion between two subsequent camera frames, the horizon features in each image are matched using nearest neighbour matching with a distance ratio cutoff as outlined in [10]. For each corresponding feature pair, the horizontal displacement in pixels of the feature between frames is calculated. Using this data, only a single parameter needs to be estimated: the robot heading change between the two frames. Since this can be estimated using only one feature correspondence, it is not necessary to use RANSAC for robust model estimation. Similarly to [4], histogram voting can be used, which is more efficient than RANSAC.

Using the histogram voting approach, the robot's heading change between two frames is estimated by the mode of all feature displacements in a feature

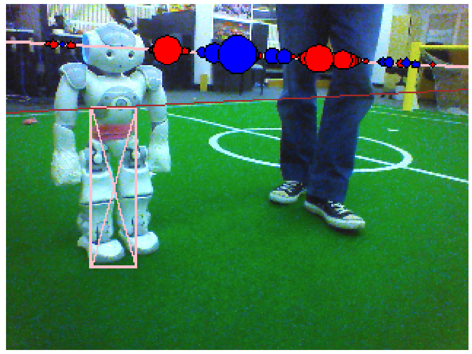

Fig. 2. A typical camera frame captured by the Nao during a soccer match could include both other robots and the referee. In this case the robot is detected and it's horizon features (indicated as blue and red blobs) are discarded. The referee cannot be detected, constituting an independently moving object.

displacement histogram, as illustrated in Figure 3. Knowing the resolution and horizontal field of view of the camera, it is trivial to convert the robot's heading change from pixels to degrees or radians, and to adjust for the movement of the robot's neck joint between frames.

Provided the stationary background is always the largest contributor of features in the image, the histogram mode will remain unaffected by the introduction of independently moving objects. If there are many or large moving objects in the frame, and the identification of the static background is uncertain, the distribution of feature displacements will be strongly multi-modal. Multi-modality enables this scenario to be easily detected, in which localisation filters can fall back to using command odometry only. In contrast, when the visual odometry is reliable, the distribution of feature displacements will be approximately uni-modal, and localisation filters can use visual odometry in preference to command odometry for heading. Using this approach makes the system relatively robust to the movement of undetected robots and referees, as shown in Figure 3. At all times the forward, sideway, and turn components of odometry used for localisation are generated by command odometry.

On the soccer field, robots are frequently bumped or impeded by other robots and obstacles that are not visible. It is advantageous to detect these collisions to prevent robot falls. This can be done by differencing visual heading odometry with command odometry. When an exponential moving average of this quantity breaches certain positive and negative bounds, it indicates that the robot is slipping with a rotation to the left or right respectively. In the robot soccer domain, we have found that reducing the stiffness of the Nao's arms at this point is sufficient to avoid a significant number of falls.

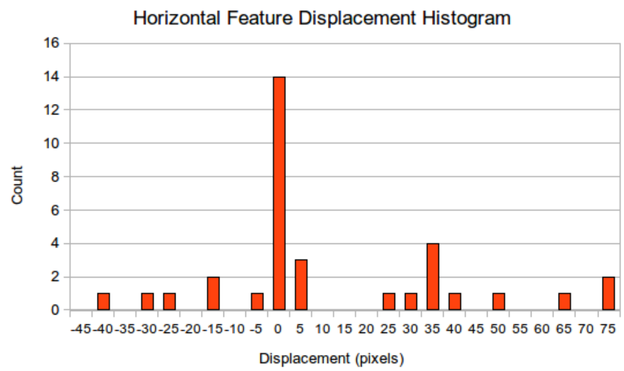

Fig. 3. The distribution of feature displacements over subsequent camera frames indicates two modes: one representing the viewing robot's heading change, measured against static background features, and the other representing the independent motion of the referee. In this case, the larger mode can be easily identified and visual heading odometry determined using this mode. If the two modes are similar in size the algorithm is able to fail gracefully by reverting to command odometry.

3.1 Reducing drift

Computing visual odometry by integrating over all adjacent frames of a video sequence leads to an accumulation of frame-to-frame motion errors, or drift. To minimise this drift, many visual odometry techniques make periodic optimisations over a number of local frames, known as sliding window bundle adjustments. In this paper, adjustments to the estimated robot trajectory are made at each step by choosing to potentially discard some frames from the image sequence. This allows the system to remain robust in the presence of single frames corrupted by horizon location error, blur, or feature occlusion.

To implement this adjustment, whenever a new frame is obtained the heading change between the new frame and each of the three previous frames is calculated. The current heading odometry is then calculated relative to the 'best' of these three prior frames. The notion of the 'best' prior frame takes into account two factors. The first is the confidence level of the heading change estimate between the prior frame and the new frame. The second is the reliability of the prior frame's own odometry estimate (itself a recursive function of confidence levels). More formally, at time t the robot's heading odometry is calculated relative to prior frame at time $t - b_t$, with $b_t \in \{1, 2, 3\}$ given by:

$$b_t = \underset{k \in \{1,2,3\}}{\arg\max} \{ \min\{ reliability_{t-k}, confidence_{t-k,t} \} \}$$

where the *reliability* of the odometry at time t is determined recursively by the *reliability* of the best prior frame odometry and the $confidence$ of the heading change estimate between $t - b_t$ and t:

$$reliability_t = \min\{reliability_{t-b_t}, confidence_{t-b_t,t}\}$$

$$reliability_0 = \infty$$

The measure of the confidence of the heading change estimate between two frames could be calculated in several different ways. It should always reflect higher confidence when the distribution of feature displacements is more uni-modal, and lower confidence when the distribution is more multi-modal (indicating difficulty in resolving independently moving objects from the stationary background).

Our approach was to calculate confidence based on the difference in the count of the first and second modes of the feature displacement histogram. Overall, the choice to consider three previous frames in the odometry calculation represents a trade off between computational cost and drift reduction. If the robot changes heading quickly there is little to be gained from increasing the size of this sliding window.

4 RESULTS

In order to evaluate the performance and robustness of 1D SURF visual odometry in comparison to naive command odometry, a quantitative benchmark test is required. In this paper the University of Michigan Benchmark test (UMBmark) is used [12]. In addition, tests are undertaken that include obstacle collisions that disrupt the natural motion of the Nao, and repeated observations of independently moving objects at close range.

By way of background, UMBmark is a procedure for quantifying the odometric accuracy of a mobile robot. In the test, the robot is pre-programmed to move in a bi-directional square path, in both the clockwise and anti-clockwise directions, and the accuracy of the return position is assessed. Although the test was designed for assessing wheel odometry error in differential-drive robots, several results of the paper are also relevant for bipedal robots. In particular, the paper illustrates that a uni-directional square path test is unsuitable for evaluating robot odometric accuracy due to the possibility of compensating turn and forward motion errors. Using the bi-directional square path test, however, these errors are revealed when the robot is run in the opposite direction.

To assess the performance of visual odometry on the Nao, the UMBmark square path procedure was repeated five times in the clockwise direction, and five times in the counter-clockwise direction. In recognition of the greater inaccuracy of bipedal robots compared to wheeled robots, the side lengths of the square path were reduced from 4 m to 2 m. Prior to this test, the command odometry was calibrated to provide reasonable performance using the same settings across five robots, but it was not calibrated to suit this particular robot.

The test was conducted on the SPL field, and the robot was controlled to ensure that in each test it walked as near to a perfect square path as possible, at a speed of approximately 15 to 20 cm/s, including turning on the spot at each corner, and returning to the starting position. The position of the robot calculated using command odometry dead-reckoning was then compared with the position of the robot using visual odometry dead-reckoning.

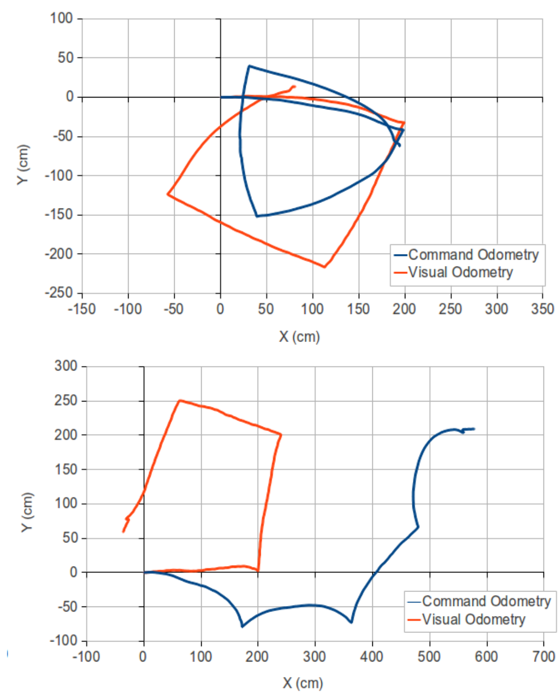

Fig. 4. Clockwise (top) and Anti-clockwise (Bottom) odometry track of an uncalibrated Nao robot walking in a 2m x 2m square path.

The estimated odometry tracks for the first trial in each direction are shown in Figures 4, illustrating a substantial deviation from the true square path in the case of the command odometry, and a much smaller deviation when the 1D SURF visual odometry was used. It is evident from these diagrams that the robot used for the test has a systematic left turn bias. In order to walk around the square field, continuous right turn corrections were required, which can be observed in the paths generated by odometry dead reckoning. The use of visual odometry has compensated for a significant proportion, but not all, of this systematic bias.

Figure 5 illustrates the final positioning error using both odometry methods at the end of five trials in each direction. Using the centre of gravity approach outlined in [12], the command odometry has a measure of odometric accuracy for systematic errors of 306 cm, relative to the visual odometry approach with

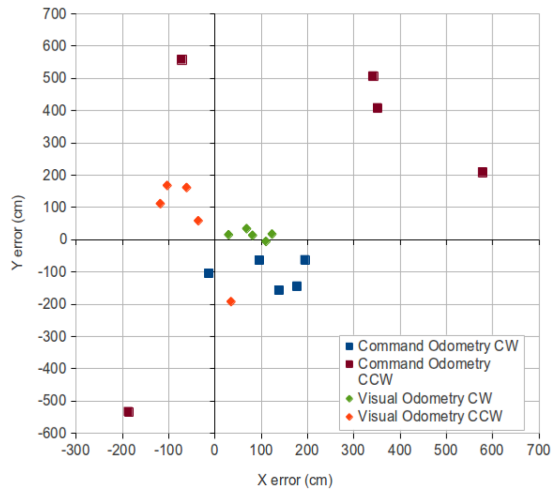

Fig. 5. UMBmark results for an uncalibrated Nao robot walking in a 2 m x 2 m square path in both clockwise (CW) and counter-clockwise (CCW) directions. Results indicate a significant increase in accuracy using visual odometry relative to naive walk-engine generated odometry.

an accuracy of 84 cm; a reduction of 73%. These results suggest that the 1D SURF visual odometry technique can compensate for a significant proportion of the systematic odometry error of an uncalibrated robot. The standard deviation for the walk-engine return positions is 550 cm, relative to 160 cm for the visual odometry return positions. This indicates that the visual odometry technique also compensates for non-systematic odometry errors.

In the next test, the robot performed five trials of a simple out-and-back manoeuvre along a 2 m long straight line. On both legs of the path, a block of wood was placed in front of one shoulder of the robot to cause a collision. Figure 6 illustrates an odometry track for a trial during which the robot experienced a gentle collision on the way out, and a more significant collision on the way back to the starting position that resulted in an uncommanded turn. Over five trials of this test, the command odometry had a measure of odometric accuracy of 140 cm with standard deviation of 102 cm, versus the combined visual odometry approach with an accuracy of 43 cm and standard deviation of 53 cm.

The final test of 1D SURF visual odometry was designed to illustrate the performance of the system with multiple visible moving objects. For this test, the Nao was programmed to maintain position in the centre of the SPL field while walking on the spot, while a number of people wearing blue jeans (simulating referees as in the RoboCup soccer application domain) were instructed to walk in front of the Nao at a range of around 50 cm. The first five people walked in the same direction, from right to left. Figure 8 illustrates that at this range the

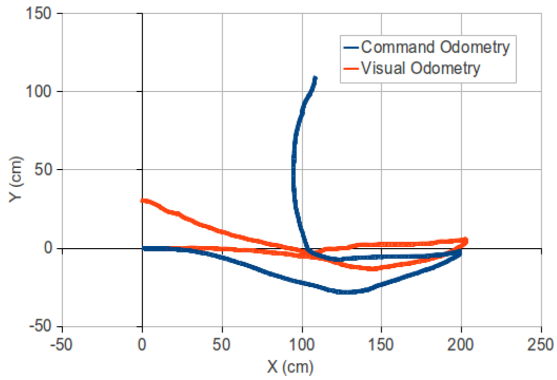

Fig. 6. Odometry track from an uncalibrated Nao on a single 2 m out and back trial with collisions in both directions.

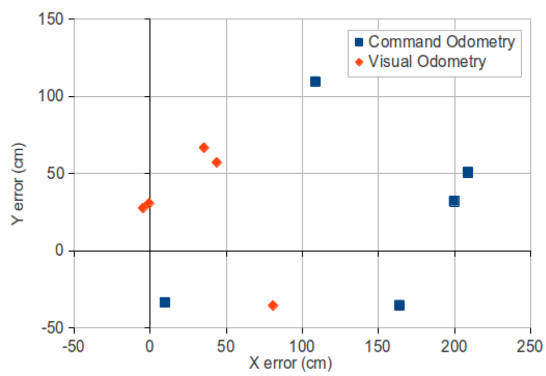

Fig. 7. Final positioning error over five out and back trials with collisions. Results again indicate a significant increase in accuracy using visual odometry relative to naive command odometry.

referees jeans will typically fill approximately half of the robot's horizontal field of view.

During this test, the robot's visual odometry was monitored to see if the movement of the objects in front of the camera could trigger a false positive heading change. As illustrated in Figure 9, during this process there is no apparent drift in the robot's visual heading odometry in over one thousand frames (approximately 33 seconds). The oscillating pattern of the heading is attributed to the robot's constant adjustments to maintain position in the centre of the field. During these experiments, the mean execution time of the visual odometry module on the Nao v4 (including the execution time required to extract 1D SURF features from the camera image) was 6.5 ms. This equates to a theoretical maximum frame rate of around 150 fps (although in practise the Nao camera is

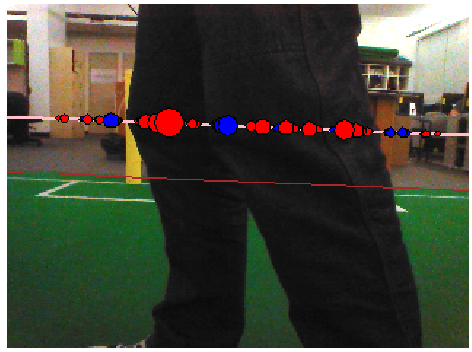

Fig. 8. Moving object tests as they appear to the Nao robot. Approximately half of the robot's horizontal field of view was obscured by the moving object. Although the person's jeans appear very dark, features are still detected in this area as indicated by the red and blue blobs.

limited to 30 fps and the remaining computational resources are used for other tasks).

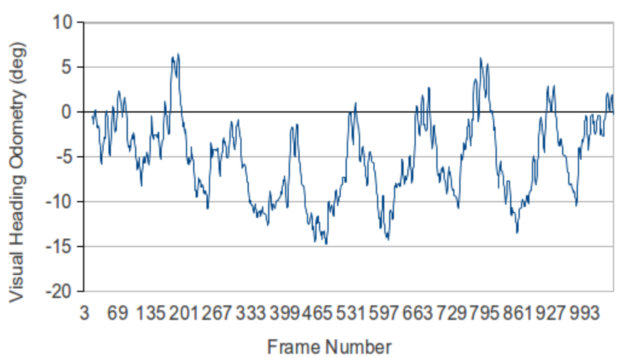

Fig. 9. Visual heading odometry with the Nao walking on the spot during the moving object test. There is no evidence of large spikes which would indicate false positive heading changes, or heading drift, during this 33 second period. An oscillating pattern can be seen which reflects the robot's actual movement adjustments to maintain position while walking on the spot.

The visual compass was used by the UNSW team in the 2012 Standard Platform League with great effect to help localisation accuracy [13].

5 CONCLUSIONS

We have presented a robust and extremely computationally inexpensive method for estimating monocular, feature-based, heading-only visual odometry using 1D SURF features. Further work is required to investigate the feasibility of full planar visual odometry (including both heading and translation) using the same features.

References

1. Scaramuzza, D., Fraundorfer, F.: Visual odometry: Part I - the first 30 years and fundamentals. IEEE Robotics and Automation Magazine **18** (2011)
2. D. Nister, O.N., Bergen, J.: Visual odometry for ground vehicle applications. Journal of Field Robotics **23** (2006)
3. Liang, B., Pears, N.: Visual navigation using planar homographies. IEEE International Conference on Robotics and Automation (ICRA2002) (2002) 205–210
4. D. Scaramuzza, F.F., Siegwart, R.: Real-time monocular visual odometry for on-road vehicles with 1-point ransac. Proceedings of IEEE International Conference on Robotics and Automation (ICRA) (2009)
5. Scaramuzza, D.: 1-point-RANSAC structure from motion for vehicle-mounted cameras by exploiting non-holonomic constraints. International Journal of Computer Vision **95** (2011) 75–85
6. Sunglok Choi, Ji Hoon Joung, W.Y., Cho, J.I.: What does ground tell us? monocular visual odometry under planar motion constraint. 11th International Conference on Control, Automation and Systems (ICCAS) (2011) 1480–1485
7. D. Hoiem, A.A.E., Hebert, M.: Recovering surface layout from an image. International Journal of Computer Vision **75** (2007)
8. Juan, L., Gwun, O.: A comparison of SIFT, PCA-SIFT and SURF. International Journal of Image Processing (IJIP) **3**(4) (2009) 143–152
9. Bernd Kitt, F.M., Stiller, C.: Moving on to dynamic environments: Visual odometry using feature classification. IEEE International Conference on Intelligent Robots and Systems (IROS) (2010)
10. Anderson, P., Yusmanthia, Y., Hengst, B., Sowmya, A.: Robot localisation using natural landmarks. Proceedings of the 2012 RoboCup International Symposium (2012)
11. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (SURF). Computer Vision and Image Understanding **110**(3) (2008) 346–359
12. Borenstein, J., Feng, L.: UMBmark: A benchmark test for measuring odometry errors in mobile robots. SPIE Conference on Mobile Robots (1995)
13. Harris, S., Anderson, P., Teh, B., Hunter, Y., Liu, R., Hengst, B., Roy, R., Li, S., Chatfield, C.: Robocup standard platform league - rUNSWift 2012 innovations. In: Proceedings of the 2012 Australasian Conference on Robotics and Automation (ACRA 2012). (2012)