Customized Sensing for Robot Swarms

D. Jud, J. Alonso Mora, J. Rehder, R. Siegwart and P. Beardsley

Abstract This paper describes a novel and compact design for an omni-directional stereo camera. A key goal of the work is to investigate the use of rapid prototyping to make the mirrors for the device, by 3D printing the mirror shape and chroming the surface. The target application is in robot swarms, and we discuss how the ability to create a customized omni-camera enables sensing to become an integrated part of system design, avoiding the constraints that arise when using commercial sensors.

1 Introduction

Motivation Our previous works with ground [1] and aerial [2] robots represent two applications of robot swarms in entertainment where localization was performed with an external overhead tracking system (Fig. 1). The motivation of this work on readily customized sensing is to develop an on-board localization solution for a robot swarm (Fig. 2) that

(a) has a reusable core technology,
(b) is customizable for different robot platforms and
(c) is low-cost.

Related work External optical tracking systems[1] provide millimeter accuracy localization but have high cost, require a fixed infrastructure, and are constrained to line-of-sight to the robot swarm. Indoor wireless solutions include UWB[2] and RFID [3]. These solutions typically encounter interference in wirelessly active environments and multi-path issues in multi-robot setups. DGPS is restricted to outdoor use only

---

[1] www.vicon.com, www.optitrack.com.
[2] www.ubisense.com.

D. Jud (✉) · J. Alonso Mora · J. Rehder · R. Siegwart
Autonomous Systems Lab, ETH Zurich, Zurich, Switzerland
e-mail: djud@student.ethz.ch

J. Alonso Mora · J. Rehder · P. Beardsley
Disney Research, Zurich, Switzerland

Fig. 1 Currently used setup
with an overhead camera
tracking a robot swarm [1]

Fig. 2 Robotic swarm of a
central unit supporting the
omnidirectional camera used
to localize a satellite relative
to the central unit
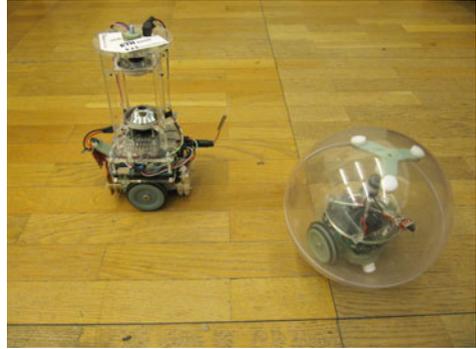
and might also be blocked by buildings. On-board localization can be used for local-
ization relative to landmarks and neighbouring robots [4], or for relative localization
only to enable collision avoidance. Although on-board localization imposes a cost
per unit, an issue for swarms of tens to hundreds of robots, the cost of sensors and
on-board computation is rapidly decreasing.

Different designs for omnidirectional stereo cameras already exist. The standard
approach is to vertically align two omnidirectional sensors [5]. Other solutions only
use one sensor, but still have stereo imaging. One approach uses a double lobed
mirror [6] with a single camera for two different view points. A concave lens in
between the mirror and the camera also enables stereo imaging [7]. Both of these
two single camera omnidirectional stereo sensors have a very short baseline, but still
a large overall size.

Problem statement The first goal of this paper is to develop a custom stereo omnidi-
rectional camera (hereafter omni-cam) for on-board localization. The second goal is
to demonstrate that rapid prototyping can be used to create mirrors specially designed
for the application in hand. The approach is motivated by the Maker Movement and
recent work on rapid prototyping for robots [8]. The contributions of this paper are:

- a novel and compact design of a stereo omni-cam.
- a demonstration that rapid prototyping can be used to make mirrors that are usable
  for computer vision.
- a method for calibrating the omni-cam setup, including the mirror shape.

2 Technical Approach

Stereo omni-cam The stereo omni-cam design has two cameras with opposing mirrors giving a stereo omnidirectional view, as shown in Fig. 4a. It is more compact than existing designs [5], because each camera is embedded inside one of the mirrors, with an aperture for viewing the opposing mirror. This enables the sensor to have any desired baseline. Apart from providing a stereo field-of-view (FOV) in the horizontal direction with the outer rim, the mirrors also have a smaller inner rim to provide a monocular field of view of the floor and ceiling in the vicinity of the robot, used for localizing the camera robot in its environment. The stereo view can be used to recover the position of any object in sight, i.e. our swarm robots. The different fields of view are shown in Fig. 3. Figure 4b shows the physical prototype with two synchronized Matrix Vision BlueFOX-MLC cameras running at a maximum of 5.8 Hz with a resolution of 2592 by 1944 pixels. The synchronized image acquisition is guaranteed by linking the cameras together by wire and only triggering the top camera whenever the bottom camera does so.

Robot platform The sensor is mounted on a mobile platform, Fig. 4c, which is deployed inside an acrylic sphere with a gravity-driven self-righting mechanism to maintain upright orientation, similar to the work of [9]. Figure 4d shows two sphere
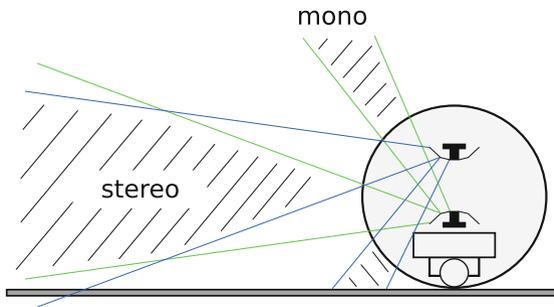

Fig. 3 The omni-cam has a large stereo view in the horizontal plane, but also monocular views of the ground and ceiling in the vicinity of the camera
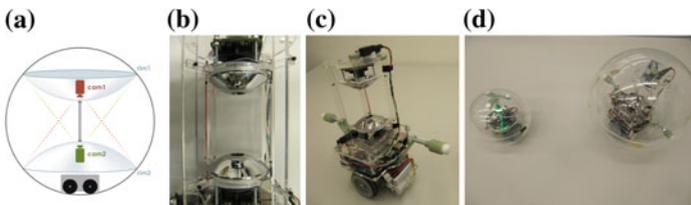

Fig. 4 **a** Schematic of the omni-cam design. There are two cameras with opposing mirrors giving a 360° stereo view. Each camera is embedded inside a mirror for compactness, with an aperture for viewing the opposing mirror. **b** Physical prototype. **c** Omni-cam mounted on robot platform. **d** Two robots inside acrylic spheres, the larger one containing the omni-cam
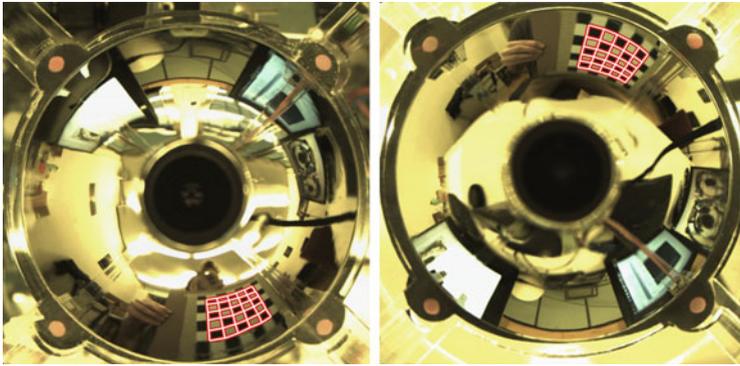
Fig. 5 Representative pair of images taken by the *top* (on the *left*) and *bottom* (on the *right*) cameras of the robot. Checkerboards were used for calibration and the images show example detections overlaid in *red*

robots, the larger one (diameter 250 mm) contains the omni-cam while the smaller one (diameter 160 mm) is a satellite robot without on-board sensing. The swarm consist of multiple satellite robots that are tracked by a single camera robot with the omni-cam. Coloured LEDs on the satellite robots are used to infer the position of the satellites relative to the camera robot. The images are processed on-board using an Odroid-XU Quad Core board. The motors and the belonging electronics are taken from an E-Puck robot [10]. All the other parts are specifically designed for the purpose of driving inside a sphere and are rapid-prototyped. The form of the wheels is made to match the shape of the sphere for maximum grip. On top is a spring loaded slider built from Teflon to keep the robot well positioned inside the sphere. The most important characteristics is the height of the center of gravity. It has to be below the center of the sphere to ensure stability. The lower the center of gravity, the better the robot behaves during dynamic manoeuvres.

Mirror design Each mirror is created by making a 3D print[3] of the desired shape, sanding, polishing the print and chroming it using a standard chroming service. The vapour deposition method only applies a thin chrome coat and does not cover up any irregularities of the surface, emphasising the importance of polishing out the layer steps from printing. Our mirror is designed using a second order polynomial function for the monocular rim and a hyperbola for the main curved part, see Fig. 6. This shape constitutes a non-central optical system (the majority of application specific mirror designs can be assumed to have this property). Figure 5 shows example camera images for the top and bottom mirrors.

Calibration Calculating the reprojection error is the crucial part in the optimization. Most commercially available omnidirectional cameras are central cameras as in Fig. 7, which means that all the outgoing rays from the mirror intersect in a virtual viewpoint. That point can be used to calculate the mirror intersection point. However

---

[3]We used the Objet Eden 350V, 16 micron layer resolution.

Fig. 6 Section view of the
mirror CAD model showing
the different functions
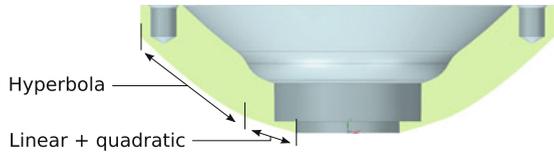resulting in different stereo
and mono views

Hyperbola

Linear + quadratic

Fig. 7 Central camera
system with a single
viewpoint in the center of the
mirror
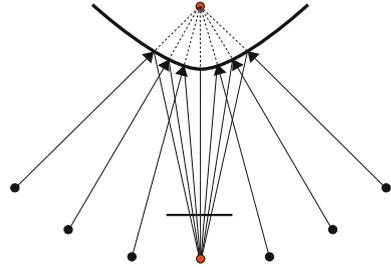
Fig. 8 Iteration process for
the reprojection in
non-central camera systems

Image plane

Focal point
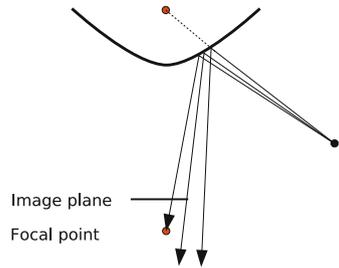
our mirrors can have arbitrary shapes which are non-central and do not have such a closed form solution for the reprojection. Figure 8 shows the iterative procedure that is implemented to find the intersection point. First, an initial guess is calculated by assuming the camera system to be central. The intersection point is then iteratively found by minimizing the distance between the incident ray and the camera center.

The shape of the final mirror differs from the input CAD model due to printer resolution and effects of sanding. We investigate three calibration approaches to determine the final shape

1. fitting a discrete number of hyperbolas to the surface and interpolating

$$z(r, \phi) = b(\phi) - \frac{b(\phi)}{a(\phi)} \sqrt{(r - 10\,\text{mm})^2 + a(\phi)^2} \qquad (1)$$

The hyperbola parameters a and b are defined at discrete steps around the mirror, e.g. they would be defined at $\phi = 0$, $\phi = \frac{\pi}{2}$, $\phi = \pi$ and $\phi = \frac{3\pi}{2}$ when interpolating over four hyperbolas. These angle dependent parameters allow a rotationally unsymmetrical model of the mirror.

2. fitting a polynomial to the surface under the assumption of radial symmetry around the mirror's central axis. The cross-section is governed by a polynomial with only even terms

$$z(r) = a_0 r + a_1 r^2 + a_2 r^4 + \cdots + a_k r^{2k} \qquad k \in \mathbb{N} \qquad (2)$$

3. fitting a polynomial to the surface in the radial direction as in Eq. 2 with additional terms for modelling the 2π-periodic asymmetries with Fourier series.

These three different models have completely different strengths and weaknesses. The first approach combines prior knowledge of the hyperbolic mirror surface with a simple model of the rotational asymmetries. Only 16 parameters are needed for both mirrors if four hyperbolas are used in the interpolation. The second approach includes the prior knowledge when initializing the polynomial parameters, but for higher order polynomials, it is less constraining than the hyperbolic model. It is simply a polynomial in radial direction and cannot model any rotational asymmetries. Although there might be the drawback of needing a high order polynomial for a sufficiently good fit. The third approach is the most flexible of them all. It uses the same polynomial in radial direction as the second approach, but also includes an angular model for the rotational asymmetries based on Fourier series. Again the same problem might occur that there is a high order of degree needed for the polynomial and the Fourier series.

Stereo calibration is carried out using checkerboards. The camera intrinsics are calibrated beforehand using a standard algorithm from the OpenCV library.[4] An initial estimate of the mirror shape and the configuration of the cameras plus mirrors is taken from the CAD design. Figure 9 shows the configuration with the coordinate transformations being optimized in the calibration. With these initial parameters, an initial estimate of 3D pose for a set of calibration patterns (see examples of detected patterns in Fig. 5) is made from triangulating the corners of the checkerboards. For all approaches above, the mirror shape and the configuration is then recovered by optimizing over the shape parameters and coordinate transformations with the objective function being the discrepancy between the recorded checkerboard positions and the reprojection of the 3D reconstructions of the checkerboards. The optimization routine is a nonlinear least-squares solver.

3 Results

Two datasets were captured, each containing 50 images of checkerboard patterns. Checkerboard detection was done using the package in [11]. The first dataset of patterns is used to compute the mirror shape based on the method in Sect. 2. The second dataset is used to evaluate the calibration, doing a stereo reconstruction of the

---

[4]www.opencv.org.

Fig. 9 Image of the omni cam showing the different coordinate frames and the according coordinate transformations as they are optimized in the calibration procedure
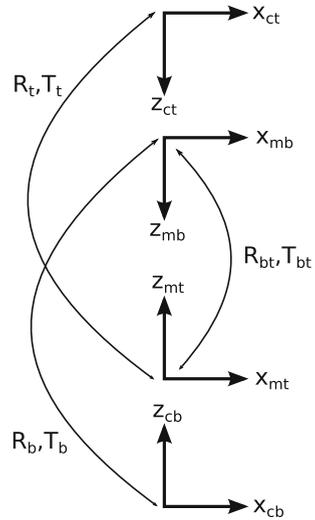

Table 1 Reprojection errors resulting from evaluating the calibration with the test dataset

(a) Reprojection error in pixels when mirror shape is modelled by discrete hyperbolas, equally-spaced with angle ϕ, with interpolation				(b) Reprojection error in pixels when mirror shape is modelled with a polynomial. Term k is the highest order of the polynomial expression in Eq. 2				
$\phi = 2\pi$	$\phi = \pi$	$\phi = \pi/2$	$\phi = \pi/4$	$k = 2$	$k = 4$	$k = 6$	$k = 8$	$k = 10$
2.72	2.66	2.2	2.36	3.93	3.37	3.25	3.23	3.21

calibration patterns, reprojecting to the image plane, and measuring the reprojection error between checkerboard detections and the projected 3D reconstructions.

Table 1a shows the results when the mirror shape is modelled using discrete hyperbolas, equally-spaced around the radial axis, with interpolation. The number of fitted hyperbolas is 1 (assumption of radial symmetry), 2, 4, and 8. The best result is obtained for 4 hyperbolas, while 8 hyperbolas is over-fitting. Table 1b shows the results when the mirror shape is modelled by a polynomial with assumption of radial symmetry, as in Eq. 2. The reprojection error decreases when the order of the polynomial expression increases. The best polynomial result (3.21) is not smaller than the lowest reprojection error with interpolated hyperbolas (2.2). This enforces the importance of a mirror model that includes the rotational asymmetries.

The third approach discussed in the calibration section is a mirror model that has the same polynomial in radial direction as Eq. 2 but also models the asymmetries with a Fourier series. A low order Fourier series did not make any difference to the result, whereas a higher order Fourier series renders the convergence to meaningful parameters difficult, because the cost surface exhibits multiple local minima. Consequently the experiments section shows data gathered with the interpolated hyperbolas
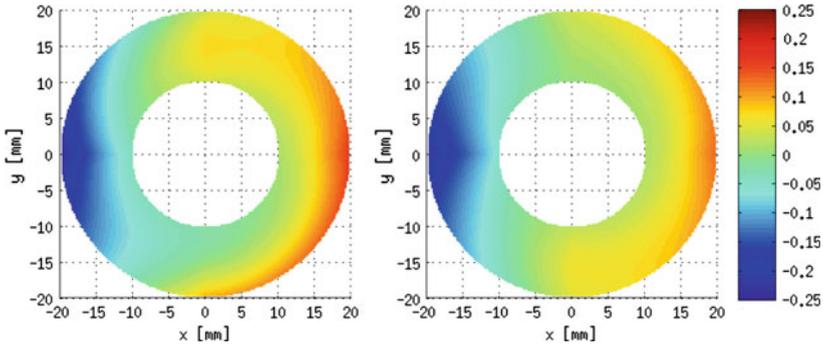
Fig. 10 Deviation of computed mirror shape from an averaged rotationally symmetric model for the *bottom* (on the *left*) and *top* (on the *right*) mirrors, for the case that the shape is modelled using four hyperbolas and interpolation. The height difference is displayed from −0.25 mm (*blue*) to +0.25 mm (*red*)

as mirror model. Figure 10 shows how the computed mirror shape diverges from an averaged rotationally symmetric representation.

The localization uncertainty [12] of satellite robots relative to the camera was not investigated within this work, but it is a part of future work on the project.

4 Experiments

This section focuses on the performance of localizing the satellite robot relative to the camera. Experiments were conducted using a central static robot equipped with the omni-cam and a satellite robot equipped with a color LED for detection (robots shown in Fig. 4d). The satellite robot was moved in a spiral with a range from 0.2 to 1.6 m from the central robot. Ground truth is obtained from an overhead tracking system. For this experiment, the mirror shape of the omni-cam was modelled using 4 hyperbolas with interpolation.

Figure 11a shows the robot trajectory. Figure 11b shows the radial component of the measured position, where the red line marks the ground truth distance and blue crosses correspond to individual localizations. While uncertainty grows with distance, the mean value is recovered correctly. A bias in the distribution around 1.3 m suggests imperfections which can most likely be attributed to the calibration. Figure 11c shows statistics for the difference in robot bearing angle between the measurement and the ground truth, with 50 % of the measurements within 0.02 rad error and 90 % within 0.04 rad error. The standard deviation of the angular error is 0.9°.
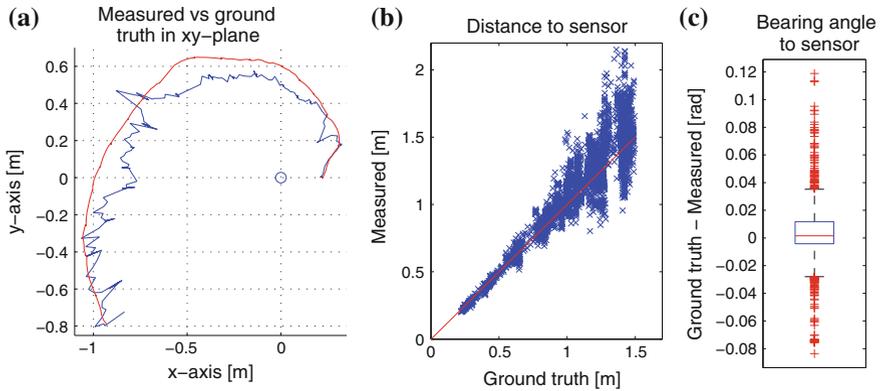
Fig. 11 a Ground-truth (*red*) and measured (*blue*) robot position using raw measurements with no filter on robot motion. **b** The radial distance of the satellite robot from the central robot. **c** The error of the angular measurement in a box plot. Both **b** and **c** are derived from measurements of the sensor

5 Application

The omni-cam is mounted onto a differential drive robot and used to track the satellite robot that is equipped with an LED. The triangulation of a single LED only provides a position and not the orientation of the robot. An extended Kalman filter is used to estimate the heading from the motion and the robots odometry. The measurement update of the Kalman filter is incorporating the position measurement at 5 Hz, whereas the state update is integrating the wheel odometry of the tracked robot at 40 Hz. This setup combined with a controller to drive the satellite robots enables the satellite to drive to waypoints relative to the camera robot. By including the wheel odometry of the camera robot into the state update, it is made possible to drive the camera robot while the satellite robots are moving to waypoints relative to the camera. Figure 12 shows the trajectory of a satellite robot with arrows indicating the estimated orientation along the trajectory.

The plot in Fig. 13 is showing the raw position measurement in x-direction from the omni-cam in blue and the according filter output in red. The increased noise at high distances from the sensor is successfully suppressed by including the odometry.

In another application, the camera robot can be driven by using a laser pointer. The robot sees the dot on the ground and follows it. This shows the utility of the additional monocular view of the camera. The bottom camera can detect the laser dot even if it is as close as 5 cm to the robot body where there is no stereo view anymore. The green laser pointer was implemented to pull the robot, whereas the red dot is pushing the robot away, as seen in Fig. 14.
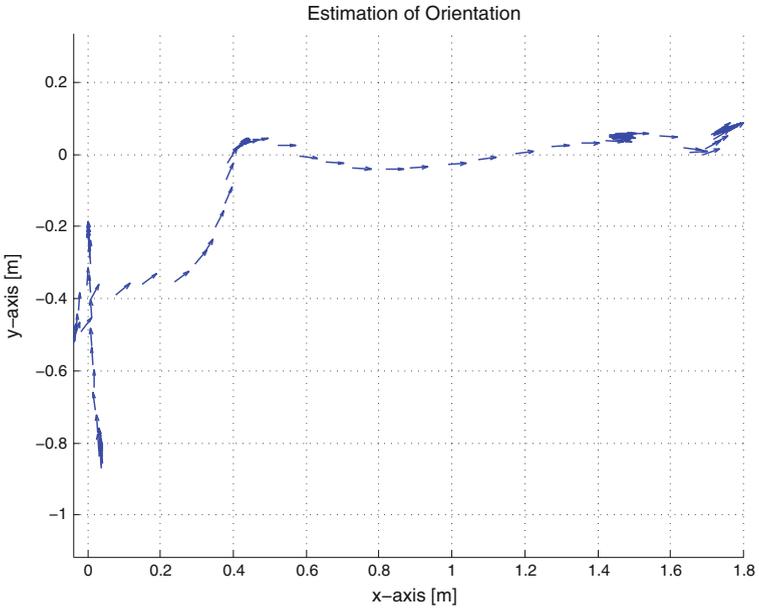
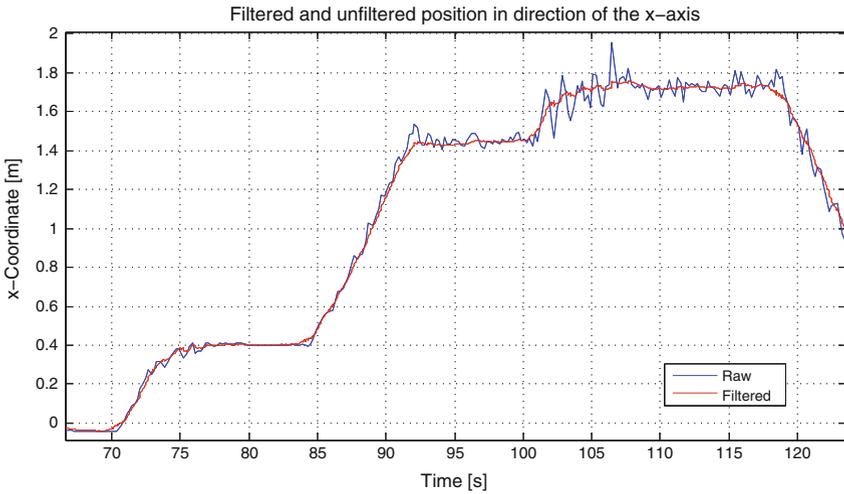**Fig. 12** Plot in the horizontal xy-plane showing the estimated orientation of the robot along a trajectory
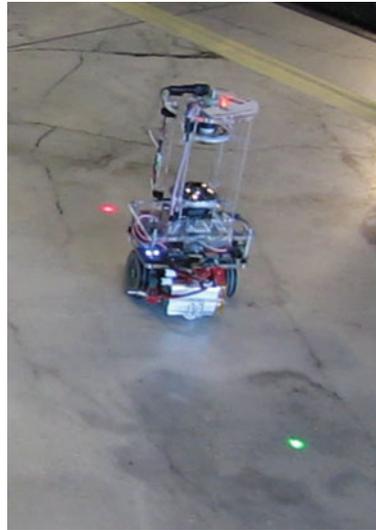

Fig. 13 A plot showing the raw measurement (*blue*) and the according Kalman filter output (*red*) in x-direction

Fig. 14 The camera robot follows the *green laser dot* while simultaneously driving away from the *red laser dot*

6 Discussion

Rapid prototyping for custom mirrors One approach to making a mirror is to mill and polish aluminium. This gives precise and high-quality results. Our focus has been on 3D printing followed by mirroring of the surface to make a custom mirror. The main motivation is ease-of-use, because 3D printing and associated design software are becoming standard hobbyist tools. In addition, the typical plastic from a 3D printer is about half the density of aluminium, making it more suitable for small, low power, ground and aerial robots. Furthermore, metal printing has now appeared at the high-end of the market, and can be expected to filter down to lower-cost solutions.

Methods to deposit a mirror on a surface include (a) traditional silvering using a silver spray, (b) chrome electroplating, and (c) vacuum deposition of chrome or aluminium. A lower-cost solution is mirror finish spray paint. Our approach was to print the 3D part, sand it by hand, and send it to a chroming service for vacuum deposition chroming. This method is relatively low cost (chroming services are widely available for the auto and home fittings markets) and can be used for large pieces.

The trade-off for this extra flexibility is that 3D printed parts are currently less precise than milled parts due to the coarse resolution of the 3D print, and the sanding and chroming process. We developed calibration methods to estimate the 3D shape, and experimental results demonstrate that the calibration is successful.

Adaptability to different robot platforms A motivation of the paper is to achieve reusable localization technology for use with different types of robot platform in varied settings. Rapid prototyping of omni-cams does not of course achieve this larger goal on its own. But by having greater control of the swarms sensing technology, the goal of a reusable localization system becomes more achievable. We are no longer

limited to commercial omni-cam designs. As described in this paper, it was readily possible to prototype our own novel design of stereo omni-cam. And it is possible to adopt an iterative development cycle—for example to tune the stereo baseline, stereo field-of-view etc.—without excessive budgetary or time cost. Thus sensing becomes a controllable part of the design process, instead of an inflexible component which imposes undesirable constraints on the robot swarm system.

References

1. Alonso-Mora, J., Breitenmoser, A., Rufli, M., Siegwart, R., Beardsley, P.: Image and animation display with multiple robots. Int. J. Robot. Res. **31**(6), 753–773 (2012)
2. Alonso-Mora, J., Schoch, M., Breitenmoser, A., Siegwart, R., Beardsley, P.: Object and animation display with multiple aerial vehicles. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1078–1083 (2012)
3. Zhou, J., Shi, J.: RFID localization algorithms and applicationsa review. In: Journal of Intelligent Manufacturing (2009)
4. Leung, K.Y.K., Barfoot, T.D., Liu, H.H.: Distributed and decentralized cooperative simultaneous localization and mapping for dynamic and sparse robot networks, pp. 3841–3847 (2011)
5. Yi-ping, T., Qing, W., Ming-li, Z., Jun, J., Yi-hua, Z.: Design of vertically aligned binocular omnistereo vision sensor. EURASIP J. Image Video Proc. **2010** (2010)
6. Cabral, E.L., de Souza, J., Hunold, M.C.: Omnidirectional stereo vision with a hyperbolic double lobed mirror. In: Proceedings of the 17th International Conference on Pattern Recognition, ICPR 2004. vol. 1, pp. 1–9. IEEE (2004)
7. Yi, S., Ahuja, N.: An omnidirectional stereo vision system using a single camera. In: 18th International Conference on Pattern Recognition, 2006. ICPR 2006. vol. 4, pp. 861–865. IEEE (2006)
8. Soltero, D.E., Julian, B.J., Onal, C.D., Rus, D.: A lightweight modular 12-dof print-and-fold hexapod. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2013, pp. 1465–1471. IEEE (2013)
9. Kawabata, K., Sato, H., Suzuki, T., Tobe, Y.: A wireless camera node with passive self-righting mechanism for capturing surrounding view. In: Gallegos-Funes, F. (ed.) Vision Sensors and Edge Detection, InTech (2010)
10. Mondada, F., Bonani, M., Raemy, X., Pugh, J., Cianci, C., Klaptocz, A., Magnenat, S., Zufferey, J.C., Floreano, D., Martinoli, A.: The e-puck, a robot designed for education in engineering. In: Proceedings of the 9th Conference on Autonomous Robot Systems and Competitions, vol. 1, pp. 59–65 (2009)
11. Geiger, A., Moosmann, F., Car, O., Schuster, B.: Automatic camera and range sensor calibration using a single shot. In: International Conference on Robotics and Automation (ICRA), St. Paul (2012)
12. Thrun, S., Burgard, W., Fox, D.: Probabilistic Robotics (Intelligent Robotics and Autonomous Agents). The MIT Press (2005)