Crowdsourcing as a Methodology to Obtain Large and Varied Robotic Data Sets

Guido de Croon [1,2], Paul K. Gerke[1,3], and Ida Sprinkhuizen-Kuyper[3,4] *

May 31, 2015

Abstract

For autonomous robots to operate successfully in unknown environments, their computer vision algorithms need to generalize over many different environments. However, due to practical considerations robotic vision experiments are typically limited to a single robot and a few (laboratory) environments. We propose crowdsourcing as a methodology for gathering large and varied robotic data sets. We evaluate the methodology by performing the first crowdsourcing experiment involving actual robots. In particular, we have made a space-game called 'Astro Drone' for a toy quad rotor, the Parrot AR drone. Nine months after the game's release, there are 14,628 downloads and 840 contributions, consisting of visual features and drone state estimates. Data mining shows the methodology's potential, providing insights such as the relation between the number of visual features and obstacle distances.

1 Introduction

Vision is a key sense for achieving robot autonomy. Robotic vision research has largely focused on distance estimation, since it is so crucial to basic

*[1] Advanced Concepts Team, European Space Agency, Noordwijk, the Netherlands, guido.de.croon+astrodrone@gmail.com
[2] Micro Air Vehicle laboratory, Control and Simulation, Faculty of Aerospace Engineering, TU Delft, the Netherlands.
[3] Radboud University Nijmegen, Department of Artificial Intelligence, the Netherlands.
[4] Radboud University Nijmegen, Donders Institute for Brain, Cognition and Behaviour, the Netherlands.

autonomous navigation capabilities. Some well-understood visual distance cues, such as stereo vision and optic flow, have a mathematical foundation in projective geometry [9]. As long as there is sufficient visual texture so that corresponding points between images can be found, these cues generalize to any environment - hence their use on planetary landers and rovers such as *Curiosity* [15, 12, 27]. Other visual distance cues, based on the appearances of objects, require training [26, 20, 19]. In order to ensure generic results with such data-driven cues, a large and varied robotic data set is necessary. However, due to practical reasons even large robotic data sets typically only involve a few robots, environments, and light conditions [11, 21, 13, 17, 1, 23].

One way to gather larger and more varied data sets is to create a 'robot internet', as proposed by the RoboEarth project [28, 24]. The idea behind this project is that autonomous robots functioning in different places can gather and exchange knowledge via a cloud robotics infrastructure. The robots are envisaged to share maps or procedures for executing common tasks. This idea may enable significantly accelerated progress in robot learning. However, no tests have been performed yet with large numbers of robots. Indeed, major obstacles include the number of autonomous robots currently functioning in the real world, and how to arrange their connection to the developed robot internet.

A different way to perform robotic data gathering is to employ *crowdsourcing* (cf. [2]). Existing crowdsourcing studies in the field of robotics focus on data-mining human-human interactions in order to teach robots social skills [5], designing swarming nanoparticles for medical applications [10], or investigating the way humans can best control swarms of robots [4]. Also due to the limited availability of real robots [5], all of these studies involve virtual worlds.

Instead, here we evaluate for the first time crowdsourcing as a methodology to gather large and varied data sets from real robots. The aim of the methodology is to gather data from many robots operating in many different environments. Specifically, we focus on a toy quad rotor named the Parrot AR drone. The crowdsourcing methodology presented here is made possible by the following unique combination of properties of the AR drone:

1. It has many sensors onboard. In particular, its frontal and bottom cameras are of interest for robotic vision research. In addition, it can hover autonomously, canceling drift with the help of optic flow as determined by a downward pointing camera.

2. The enhanced autonomy properties make it possible to play augmented reality games. A fun game can provide an effective incentive for people to participate in the experiment.

3. Parrot has made the control library open source. This allows the creation of games by third parties.

4. Over 500,000 drones have been sold over the last three years, making it a widely available robot [18].

5. The AR drone is piloted with a smart device such as an iPhone. Since smart devices can connect to the internet, scientific data can be sent to the experimenter.

We use the robotic crowdsourcing methodology to test a hypothesis on distance estimation with Speeded-Up Robust Features (SURF) [3].

The remainder of the article is structured as follows. In Section 2, we explain the robotic crowdsourcing methodology. Subsequently, we report on the results in Section 3, discussing both the data gathered with the methodology and the results concerning our specific hypothesis. We conclude in Section 4.

2 Methodology

The idea to perform robotic crowdsourcing is based on the Parrot AR drone's unique properties mentioned in the introduction. The *crowd* consists of people owning a Parrot AR drone. This drone cannot only be used for flying, but also for (1) making videos, and (2) playing Augmented Reality (AR) games. The drone's videos, combined with any available state information, are of interest to robotics research on, e.g., obstacle detection. However, asking drone owners to fly many times towards different obstacles does not sound very rewarding. Instead, playing a fun AR game may be a sufficient *incentive* for the crowd to participate in the experiment. A major challenge is to align the setup of the game with the setup of the scientific experiment to be performed.

In the remainder of this section, we will explain the robotic crowdsourcing methodology, which is illustrated in the context of our specific experiment in Figure 1. The details of the figure will become clear in the following subsections.

2.1 The Experiment

The motivation for the experiment comes from the task of visual obstacle detection. In order to detect obstacles, SURF features can be used [3]. The image locations of such features are commonly tracked over time in order to

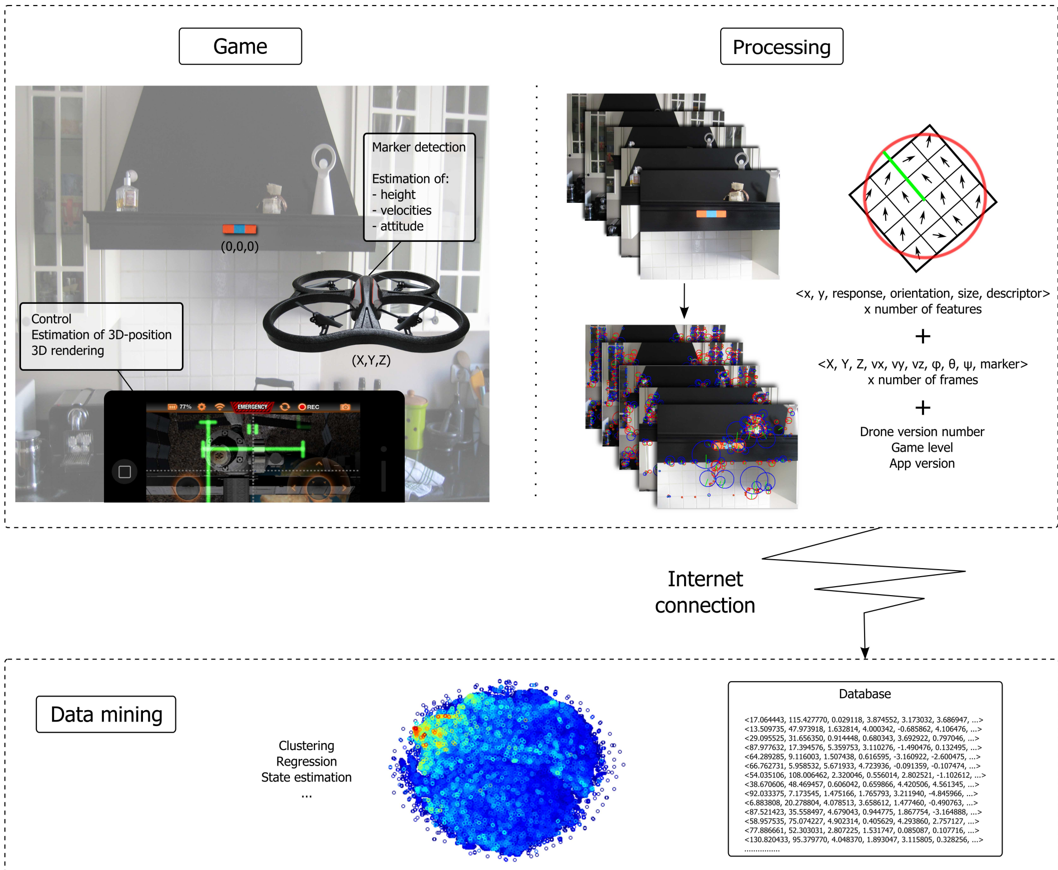

Figure 1: Methodology of robotic crowdsourcing, illustrated in the context of our particular experiment. The game played with the AR drone is shown on the top left, together with the computational processes running on the drone and smart device during the game. During post-game processing (top-right), SURF features are extracted from five images that were stored during the game. If the player agrees, the features are combined with the drone's state estimates and sent to a database (bottom). The data can be analyzed with respect to a specific hypothesis or with various data mining techniques. The center bottom plot shows a Barnes-Hut-SNE clustering [14] of the feature descriptors, colored according to the 'response' strength of the features.

perform visual odometry [16, 8] or Simultaneous Localization And Mapping (SLAM) [6, 25, 29, 22]. Here our interest is in using simple statistics of SURF features as an appearance cue for the distance to an obstacle. In particular, our hypothesis is that the number of features in an image decreases with a decreasing distance to an obstacle. There are two reasons underlying this hypothesis. First, as a robot approaches an obstacle, fewer and fewer objects are in view. Since features are often associated with parts of objects, we expect the number of features to decrease. Second, the robot's movements induce motion blur. When moving at the same speed, there is more motion blur when the robot is close to an object than when it is far away. Since blur smoothens out the texture in the image, it also reduces the number of features in the image.

Figure 2 shows the relation between the number of SURF features and the distance to an obstacle, as gathered by the authors in TU Delft's faculty building of Aerospace Engineering. The robotic crowdsourcing experiment serves to verify the hypothesis on a more varied data set.

Please note that the choice for SURF features allows different types of post-experiment analysis. This analysis does not have to be limited to the verification of this single hypothesis.

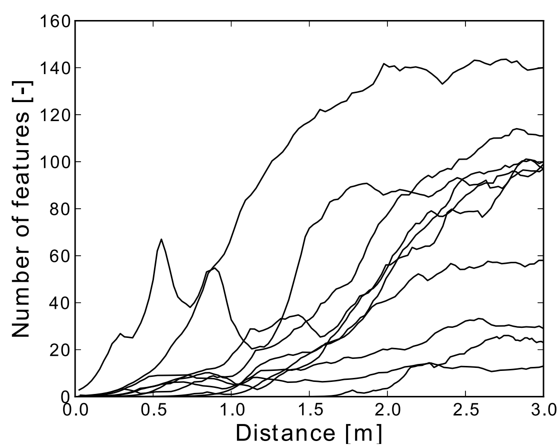

Figure 2: Number of features over time for 10 different image sequences made with an AR drone 2 in a laboratory setting. All sequences start at 3m from an obstacle and end close to the obstacle.

2.2 The Game

The main driver for the game concept is that it should align with the experiment. In this case, the player should fly the drone toward an obstacle with the camera facing forward. Since the main envisaged incentive for players is to have fun, we had to think of a way to make this flight maneuver fun in augmented reality. The nature of the European Space Agency provided an ideal setting: there are many exciting space missions that can be coupled to different flight maneuvers of interest to robotic experiments.

The concept chosen for the first level of our game was that of docking to the International Space Station (ISS). This is a very delicate maneuver, which requires swiftness and accuracy. A screenshot of the game can be seen in the top left part of Figure 1. The relation between the real world and augmented reality is made with the help of an orange-blue marker that is provided with the Parrot AR drone.

In the game the marker represents the docking gate. As shown in the top left of Figure 1, it is detected onboard the drone. The detections are used by the state estimation onboard the smart device for rendering the 3D-space around the ISS. A detailed discussion of the 3D state estimation of the drone (X, Y, Z) relative to the marker position $(0, 0, 0)$ falls outside the scope of this conference article. However, please note that this part is crucial to the game's success. For the game experience, not only accuracy is of importance, but also responsiveness. This latter property was difficult to obtain, with the state estimation data that is transmitted from the drone to the smart device sometimes lagging behind the robot's movements for almost 0.5s. The solution here was to incorporate control actions into the 3D state estimation filter: if the player tilts the smart device to move the drone, the simulated spacecraft already starts moving before receiving feedback from the drone.

SURF features are not extracted during the flight. The reason for this is that such an extraction on computationally limited smart devices would lead to an extra computational burden and hence large delays in the gameplay. Therefore, images are stored on the device. Since our interest is in the statistics of SURF features while approaching an obstacle, images are only stored during flight if the drone has sufficient forward velocity ($\geq$ 0.15 m/s). Five subsequent images are then stored on the smart device with a 0.25 s interval. The time interval represents a trade-off between having sufficient motion in between images and being able to track features over the images. The stored images are accompanied by the drone's state estimates at the same times.

After playing the game, the player receives a score and can go to the main menu of the game. In that main menu the player can choose to go to the highscore table. If the player does so, a question pops up whether the player wants to contribute to the crowdsourcing experiment. An explanation is given of the experiment and data, with a link to the game's web site. If the player agrees to participate in the experiment, the score is added to the highscore table. Simultaneously, the SURF features are extracted from the images (top right in Figure 1). These features are combined with the state estimates and meta-information. This data is then sent over the internet to a database. The gathered data can subsequently be analyzed with data mining techniques such as clustering or regression (bottom part of Figure 1).

Please note that players can play the (free) game without having to participate in the experiment, keeping the participation completely optional. Only if the player wants to figure in the highscore table, participation is necessary.

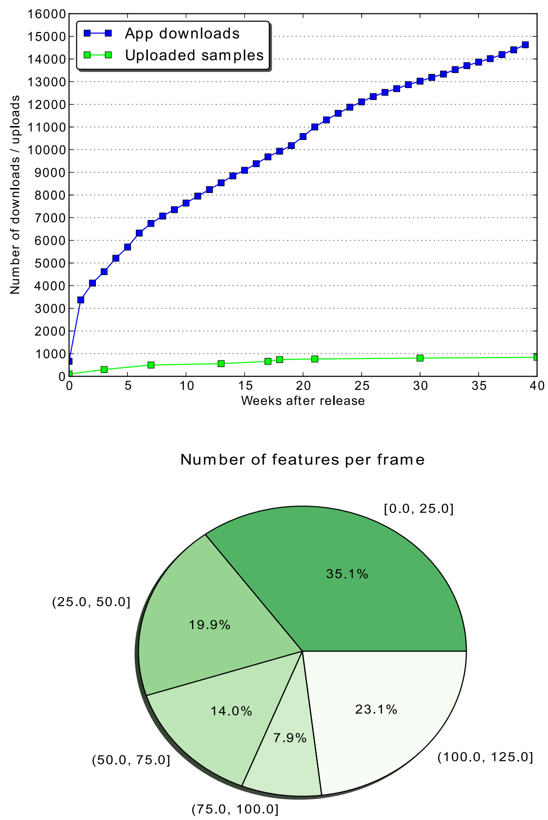

Figure 3: **Left:** Download statistics. Number of downloaded apps (blue) and uploaded samples (green) over time in weeks after the release of the game on March 15, 2013. **Right:** Histogram of the number of features per image in the gathered data set.

2.3 Market Analysis

In 2011, at the origin of this project, there were only iOS apps available for the AR drone. Almost no information was available on the popularity of these apps, since app download numbers are not publicly available in the iTunes store. Although a thorough market analysis was out of reach, we did notice that the number of augmented reality games for the drone was rather limited. A possible reason for this was that Parrot did open up a Standard Development Kit (SDK) for the drone's 'control library', but not the source code for the games that they made. As a result, most third-party apps focused on new ways of controlling the drone or on games with very few visual elements. No third-party app involved a 3D augmented reality. We figured that creating a free augmented reality game for the drone would hence be a nice addition to the games that can be played with the Parrot AR drone. Because of the limited number of games, we assumed the game would appear if someone searches for "AR drone" in the iTunes store.

Also, in 2011 it was difficult to estimate the number of drone owners playing augmented reality games. Although the concrete number of $500,000$ sold drones was mentioned recently [18], it is not clear what portion of these drones have been bought by research labs (less likely to play games) or individuals (more likely to play games), what portion of these drones are still in one piece and still used, how many users look for new games when they do use their drone, etc. The number of active drone users with an interest in augmented reality games is likely much lower than $500,000$. Although this makes the potential crowd much smaller than for other scientific crowdsourcing studies not using real robots [5, 10, 4], we expected the crowd to be quite interested by a new AR game.

2.4 Planning Game Release

The following steps were set up to prepare the game for release. First, we had various colleagues and students play the game and then fill in a questionnaire on the game experience. This survey provided valuable feedback, since naive players play the game and operate the drone differently than the game's programmers. Second, we prepared all marketing materials. A video was made by the European Space Agency, a news item for ESA's central web site was written, a press release was prepared, and some messages were prepared for various forums. Third, we submitted the app to the iTunes store, selecting the option to release the app into the store manually. After one iteration, which required us to add some extra information on the connectivity with the drone, the app was accepted and ready for release.

3 Results

The game, called "Astro Drone", has been released by the European Space Agency as a free app in the iTunes store[1] on March 15, 2013. The game release drew quite some attention, with news items appearing in online media ranging from BBC Technology to "Die Zeit", news papers such as the Dutch "Metro", and - very important for reaching our crowd - on social media such as Parrot's facebook page.

We determine the results based on two sources: (1) the information available from the ESA database containing the uploads of the players, and (2) the information available in the iTunes store. The information sent to the ESA database has been described in the previous section. The information from the iTunes store consists of: the number of downloads over time, the number of downloads per country, and the number of downloads per device. It is not possible to see whether people downloading the app actually play the game. We even expect some people without access to an AR drone to download the game, despite the clear message on this in the game's description.

The left plot in Figure 3 shows the number of app downloads over time (blue line) and the number of 'samples' transmitted by participating players (green line). A sample consists of the SURF features extracted from five images with corresponding state estimates. Since its release the game has been downloaded $14,628$ times in many countries such as the United States, Turkey, and Russia, resulting in 840 contributions (on December 15, 2013). The percentage of samples with respect to downloads is hence 5.74%. For privacy reasons, we do not keep track of how many samples are contributed by the different players. Therefore, we cannot relate the number of samples exactly to the number of players / robots. However, given the current limited extent of the game, we expect most players to make a single contribution, if any. A few players may be quite enthusiastic and contribute a large number of samples. This reasoning suggests that hundreds of robots have been involved in the data gathering.

In this section, we first investigate the gathered data (Subsection 3.1) and then verify whether the hypothesis on a decreasing number of features is correct (Subsection 3.2).

3.1 Data Investigation

The 840 contributed samples together lead to a database of $226,937$ SURF features. At `http://www.bene-guido.eu/` the Astro Drone data set and

[1]For information on the game and a link to its place in the iTunes (and as of May 2014 Google Play) store, please visit the support site: `http://www.astrodrone.org/`

analysis scripts are available for inspection or use by other researchers.

In order to assess the quality of the data, we first look at the following sample characteristics. Concerning the type of drone, 80.2% of the samples was made with the AR drone 2 and 19.8% with the AR drone 1. The AR drone 2 has a better camera quality and larger image size, which is likely to influence the number and type of features extracted. Because in the experiment we want to extract distance cues from visual features, it is important to have a ground-truth distance to an obstacle in the environment. This ground-truth is provided by the marker detection. For this reason we look at the percentage of samples in which the marker is detected. Inspection of the ESA database shows that the marker is detected in 52.0% of the samples. Finally, the data is most interesting if players fly toward objects with some visual texture, i.e., flying only toward white walls is not in the interest of the experiment. Therefore, we look at the number of features per image. The right plot in Figure 3 shows a pie chart of the number of features per image. Although there is a considerable percentage of images with 0 - 25 features (35.1%), there are also many images with many more features. Please remark that since we limited the number of sent features per image to 125, the interval of $(100, 125]$ also represents all images with more than 125 features $(100, -)$ and amounts to 23.1%.

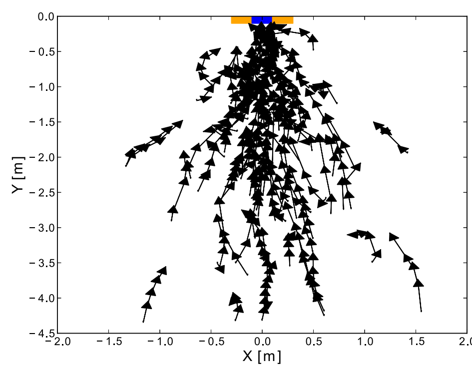

Figure 4: Trajectories for the 92 selected informative samples for AR drone 2.

3.2 Hypothesis Validation

In order to test whether the number of features correlates with the distance to the obstacle on which the marker is placed, we first select a set of the most informative data samples with respect to the ground-truth. The criteria for a sample to be 'informative' are: (1) the marker is detected, (2) the drone

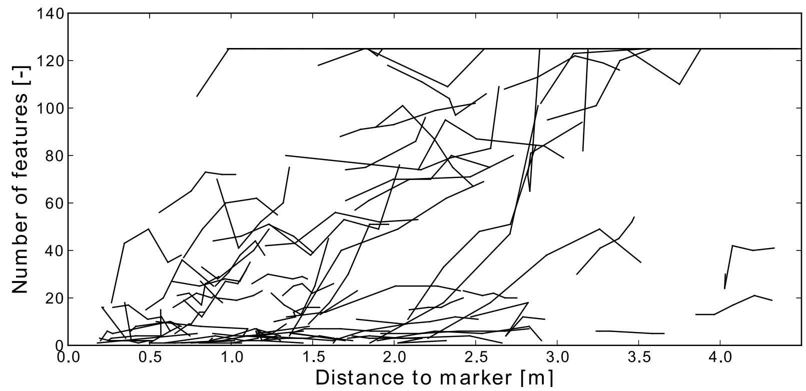

Figure 5: Relation between the number of features and distance to an obstacle as determined with robotic crowdsourcing. Distance to the marker (x-axis) versus the number of features (y-axis). In order to limit the amount of data sent over the internet, the number of features was limited to 125.

moves sufficiently fast, i.e., more than 30cm over the five images, (3) the drone moves toward a region of a meter around the marker position, and (4) the drone does not end up behind the marker position. Applying these criteria results in 110 samples that can be used for the current analysis. Since most of these samples (92) have been made with the AR drone 2, we will focus on that drone version in the analysis. Figure 4 shows the trajectories for the selected samples.

Figure 5 shows the relation between the number of extracted SURF features in the image and the distance to the marker. At smaller distances, fewer features are detected in the image. When a least-squares linear fit of distance versus the number of features is performed for each sample, 66% of the estimated slopes is positive and 14% is equal to 0. The zero slopes mostly occur due to the cut-off at 125 features.

The number of features does not provide a metric distance estimate. However, it can be a useful complement to other visual cues, since it does capture situations in which accurate 3D-reconstruction or optic flow determination can be difficult. In [7] a similar non-metric cue was shown to be a good complement to optic flow for obstacle detection in an obstacle avoidance task.

4 Conclusions

Our main conclusion is that robotic crowdsourcing is a viable methodology for gathering large and varied robotic data sets. The 840 uploaded samples contain $226,937$ gathered SURF features.

As a consequence of the methodology, the environments are not known to nor controlled by the researcher. Therefore, it is more difficult to perform experiments requiring a ground-truth. Still, we were able to test our hypothesis that the number of features decreases with the obstacle distance. The informative samples from 92 different approach flights of the AR drone 2 show a clear relationship between the number of extracted features and the distance to an obstacle. This relationship can be used to ameliorate autonomous obstacle avoidance, and shows the potential of robotic crowdsourcing for experiments requiring a ground-truth. Moreover, the gathered data also allows other types of analysis that do not rely on ground truth obstacle distance. For instance, unsupervised feature clustering (see, e.g., the bottom part of Figure 1) can use all 840 uploaded samples.

The creation and release of the Astro Drone app in total required around 18 person-months of work. Hence, one can wonder if the same number of samples could not have been obtained with less effort by manually collecting the data. However, the uniqueness of the data set does not lie in its size, but in its variety: it has been collected by hundreds of different players controlling different robots in different environments and (light) conditions.

Moreover, the effort to make this first release of the game is considerably larger than the creation of additional levels, which can be used to answer new research questions. For example, we intend to make a Mars landing level, in which players have to land safely with a limited amount of fuel. The gathered data will include visual features from the downward pointing camera and sonar. Sonar is likely to provide a more reliable ground-truth measurement than the (forward) marker detection used in the current experiment. A different path to explore is a level in which we analyze the flight behavior of experienced and less experienced players. The data may help in understanding human (learning of) piloting skills under different conditions. An additional advantage of adding a new level is that we can use it to increase the incentive to participate in the experiment, for example by only giving access to the second level in the case of participation.

We expect an increasing number of toy robots with similar properties as the Parrot AR drone to be available to the 'crowd'. Hence, the methodology proposed in this article will allow ever more people to contribute to robot autonomy by playing games at home.

Acknowledgements

We thank L. Summerer, I. Duvaux-Bechon, D. Izzo, A. Le F'loch, M. Sabbatini, N. Menning, J. De Dalmau, and many others for their help in realizing this project within the European Space Agency. Furthermore, we thank L. van der Maaten and P. Haselager for their comments on a draft version of this article.

References

[1] H. Aanaes, A. Lindbjerg-Dahl, and K. Steenstrup-Pedersen. Interesting interest points - a comparative study of interest point performance on a unique data set. *International Journal of Computer Vision*, 2011.

[2] E.E. Arolas and G.F. Ladrn de Guevara. Towards an integrated crowdsourcing definition. *Journal of Information Science*, 38(2):189–200, 2012.

[3] H. Bay, A. Ess, T. Tuytelaars, and L. van Gool. SURF: Speeded up robust features. *Computer Vision and Image Understanding (CVIU)*, 110(3):346–359, 2008.

[4] A. Becker. Crowdsourcing swarm manipulation experiments. http://www.swarmcontrol.net/, 2013.

[5] S. Chernova, N. DePalma, E. Morant, and C. Breazeal. Crowdsourcing human-robot interaction: Application from virtual to physical worlds. In *RO-MAN*, 2011.

[6] A.J. Davison and D.W. Murray. Simultaneous localisation and map-building using active vision. *IEEE PAMI*, 2002.

[7] G.C.H.E. de Croon, E. de Weerdt, C. de Wagter, B.D.W. Remes, and R. Ruijsink. The appearance variation cue for obstacle avoidance. *IEEE Transactions on Robotics*, 28(2):529–534, 2012.

[8] F. Fraundorfer and D. Scaramuzza. Visual odometry: Part II - matching, robustness, and applications. *IEEE Robotics and Automation Magazine*, 19(2), 2012.

[9] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004.

[10] S. Hauert, J. Lo, O. Nachum, and S. Bhatia. Nanodoctor: Crowdsourcing the design of swarming nanoparticles. In *Biomedical Engineering Society, Seattle USA*, 2013.

[11] A. Howard and N. Roy. The robotics data set repository (Radish). `http://radish.sourceforge.net/`, 2003.

[12] A. Johnson, R. Willson, Y. Cheng, J. Goguen, C. Leger, M. Sanmartin, and L. Matthies. Design through operation of an image-based velocity estimation system for Mars landing. *International Journal of Computer Vision*, 74(3):319–341, 2007.

[13] K.Y.K. Leung, Y. Halpern, T.D. Barfoot, and H.H.T. Liu. The UTIAS multi-robot cooperative localization and mapping dataset. *International Journal of Robotics Research*, 30(8):969–974, 2011.

[14] L.J.P. van der Maaten. Barnes-Hut-SNE. In *International Conference on Learning Representations*, 2013.

[15] L. Matthies, M. Maimone, A. Johnson, Y. Cheng, R. Willson, C. Villalpando, S. Goldberg, A. Huertas, A. Stein, and A. Angelova. Computer vision on Mars. *International Journal of Computer Vision*, 2007.

[16] D. Nister, O. Naroditsky, and J. Bergen. Visual odometry. In *CVPR*, 2004.

[17] G. Pandey, J.R. McBride, and R.M. Eustice. Ford campus vision and LIDAR data set. *International Journal of Robotics Research*, 30(13):1543–1552, 2011.

[18] Parrot. Parrot establishes itself on the civil drones market. http://www.parrot.com/paris-air-show-2013/usa/bg-press-release.pdf, 2013.

[19] S. Ross, N. Melik-Barkhudarov, K.S. Shankar, A. Wendel, D. Dey, J.A. Bagnell, and M. Hebert. Learning monocular reactive UAV control in cluttered natural environments. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2013.

[20] A. Saxena, J. Schulte, and A.Y. Ng. Depth estimation using monocular and stereo cues. In *In IJCAI*, 2007.

[21] M. Smith, I. Baldwin, W. Churchill, R. Paul, and P. Newman. The new college vision and laser data set. *International Journal of Robotics Research*, 28(5):595 – 599, 2009.

[22] H. Strasdat, J.M.M. Montiel, and A.J. Davison. Real-time monocular SLAM: why filter? In *ICRA*, 2010.

[23] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers. A benchmark for the evaluation of RGB-D SLAM systems. In *Proc. of the International Conference on Intelligent Robot Systems (IROS)*, Oct. 2012.

[24] M. Tenorth, U. Klank, D. Pangercic, and M. Beetz. Web-enabled robots. *IEEE Robotics and Automation Magazine*, 18(2), 2011.

[25] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. MIT Press, Cambridge, MA, 2005.

[26] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, K. Lau, C. Oakley, M. Palatucci, V. Pratt, P. Stang, S. Strohband, C. Dupont, L.-E. Jendrossek, C. Koelen, C. Markey, C. Rummel, J. van Niekerk, E. Jensen, P. Alessandrini, G. Bradski, B. Davies, S. Ettinger, A. Kaehler, A. Nefian, and P. Mahoney. Stanley: The robot that won the DARPA grand challenge. *Journal of Field Robotics*, 23(9):661–692, 2006.

[27] F. Valette, F. Ruffier, S. Viollet, and T. Seidl. Biomimetic optic flow sensing applied to a lunar landing scenario. In *IEEE International Conference on Robotics and Automation (ICRA 2010)*, pages 2253 – 2260, 2010.

[28] M. Waibel, M. Beetz, J. Civera, R. DAndrea, J. Elfring, D. Galvez-Lopez, K. Haussermann, R. Janssen, J.M.M. Montiel, A. Perzylo, B. Schiessle, M. Tenorth, O. Zweigle, and R. van de Molengraft. RoboEarth a World Wide Web for robots. *IEEE Robotics and Automation Magazine*, 18(2):69–82, 2011.

[29] B. Williams, M. Cummins, J. Neira, P. Newman, I. Reid, and J. Tardós. A comparison of loop closing techniques in monocular SLAM. *Robotics and Autonomous Systems*, 2009.