Monocular ORB-SLAM on a Humanoid Robot for Localization Purposes

Daniel Ginn(✉), Alexandre Mendes, Stephan Chalup, and Jake Fountain

School of Electrical Engineering and Computing, Faculty of Engineering and Built Environment, University of Newcastle, Callaghan, NSW, Australia
{daniel.ginn,jake.fountain}@uon.edu.au
{alexandre.mendes,stephan.chalup}@newcastle.edu.au

Abstract. This work presents an application of ORB-SLAM in an iGus bipedal humanoid robotic platform. The method was adapted from its original implementation into the framework used by the NUbots robotic soccer team and used for localization purposes. The paper presents a description of the challenges to implement the adaptation, as well as several tests where the method's performance is analyzed to determine its suitability for further development and use on medium sized humanoid robots.

To conduct the tests, we determined the robot's real location using a high-accuracy, camera-based infrared tracking system. Two experiments were performed to estimate the robustness of the method to the vibration and constant camera wobbling inherent to a bipedal walk and its ability to deal with the kidnapped robot problem.

The tests indicate that ORB-SLAM is suitable for implementation into a medium sized humanoid robot in situations comparable to a robotic soccer environment, and requires relatively low computational resources, leaving enough CPU power for other tasks. Additionally, since ORB-SLAM is robust to the difficulties associated with humanoid motion, we conclude that it provides a good SLAM algorithm to enhance with features specific to the humanoid robotic platform.

Keywords: ORB-SLAM · Humanoid · Visual odometry
Visual SLAM

1 Introduction

One of the primary enabling capabilities of any autonomous mobile robotics platform is the ability to keep track of its location. Various methods have been utilized to achieve this, including odometry sensors, inertial measurement units (IMU) like accelerometers and gyroscopes, and SLAM (Simultaneous Localization and Mapping) techniques using cameras and Lidar. In the last decade, visual odometry and visual SLAM techniques have become increasingly capable of being run in real-time on mobile robotics platforms, with ORB-SLAM [6]

T. Mitrovic et al. (Eds.): AI 2018, LNAI 11320, pp. 77–82, 2018.
https://doi.org/10.1007/978-3-030-03991-2_8

widely considered state-of-the-art. The application of these SLAM techniques to various mobile robotics platforms has focused mainly on ground-based wheeled platforms, flying quadcopters and hand-held cameras. These platforms are relatively stable when in motion. However, considerably less work has been done on humanoid robots. Humanoid robots are bipedal which is a significantly more unstable method of locomotion, considerably reducing the accuracy of odometry measurements. This work focuses on the RoboCup soccer competition and thus the humanoid platform used supports only the strictly humanoid binocular cameras, which is one of the requirements of the humanoid league.

In this paper, we report on and discuss the suitability of using ORB-SLAM on a medium sized humanoid robot to provide visual odometry. This paper will only investigate monocular ORB-SLAM, due to the computational limitations of the robot, as all processing is done on-board. To the best of the authors knowledge, there has been no feasibility study on the use of the state-of-the-art monocular ORB-SLAM on humanoids.

Only one other 2018 RoboCup humanoid team (NimbRo[1]) mentioned using a visual odometry system. They report testing two state-of-the-art visual odometry (VO) techniques called SVO [3] and DSO [1]. They found that these techniques failed over longer periods of time and under rapid movement. We believe that a full visual SLAM system which provides loop closure, map building and relocalisation will be able to succeed in the same circumstances.

The remainder of this paper is organized as follows: Sect. 2 gives a brief overview of related work and concepts; Sect. 3 presents the humanoid robot and experiment design used in this paper; Sect. 4 presents the results of ORB-SLAM's performance on a humanoid robot; Sect. 5 provides a discussion on the advantages and disadvantages of ORB-SLAM; and Sect. 6 presents our conclusions.

2 Background

2.1 Related Work

The majority of works that implement SLAM onto humanoids use Lidar or RGB-D sensors. This choice is often made due to the superior accuracy of these sensors. Both however have their drawbacks, with Lidar sensors being quite expensive, and RGB-D sensors having a fairly limited range. Cameras in contrast are very cheap, and are usually already necessary for other vision processing tasks. Among the studies that implement passive visual SLAM onto a humanoid, Oriolo et al. [7] used odometry and foot pressure sensors to provide the state prediction for an EKF (Extended Kalman Filter), and PTAM and IMU data to provide the measurement update.

Scona et al. [8] used ElasticFusion [10] (an originally RGB-D camera SLAM method) on a 1.8 m tall humanoid robot, and addressed the issue of what happens when a robot faces its camera at a featureless area such as a wall. Odometry

[1] https://www.robocuphumanoid.org/qualification/2018/AdultSize/NimbRo/tdp.pdf.

and IMU data was used to provide a motion prior that estimates where the tracked features have moved to since the last frame. The odometry and IMU data was then fused with the results of the SLAM algorithm. ElasticFusion is a SLAM technique that tracks pixel intensities as opposed to tracking features like ORB-SLAM and PTAM. As mentioned in Sect. 1, the RoboCup team NimbRo reported trialing DSO and SVO, however they found the lack of long term reliability of these purely visual odometry techniques leads to unreliable results or complete loss of tracking.

Monocular or Binocular ORB-SLAM has been implemented on other platforms such as Micro-aerial Vehicles (MAVs) and image datasets from wheeled ground vehicles, but not on humanoids to the best of our knowledge. Using a ground station, Garcia et al. [4] combined LSD-SLAM [2] (which is another featureless, pixel tracking SLAM method) and ORB-SLAM (feature tracking based) in a complementary way, along with IMU data to provide pose and map data which could then be used by the ground station to provide path planning commands to the MAV. Song et al. [9] collected binocular ORB-SLAM data, along with IMU, GPS, and barometric data, which was then processed offline. For ground based vehicles, Mur-Artal et al. [6] who are the creators of ORB-SLAM, used wheeled ground based vehicle datasets of cars and smaller indoor wheeled robots, as well as quadrotors to benchmark their results against other SLAM algorithms.

2.2 Porting ORB-SLAM

ORB-SLAM, which is available for open source download[2], relies on OpenCV, as well as two third party libraries which come included in the download. The first is DBoW2 which is a Bags of Words library, and the second is g2o which handles the bundle adjustments and optimizations.

The NUbots team uses a framework called NUClear [5], which required some reorganizing of ORB-SLAM. The source code was mostly able to stay untouched, except that the threading had to be modified to work in NUClear, as it manages its own threading. The two third party libraries needed to be compiled separately and included into NUClear's libraries.

3 Methodology

3.1 NUbots iGus Humanoid Robot

The iGus humanoid robot used in this paper is a modified version of the NimbRo robot[3]. It is 90 cm tall and contains a Point Grey Flea3-U3-13E4 Global shutter cameras with fisheye lenses, an IMU, and an Intel NUC7i7BNH (core i7-7567U) 3.5 GHz processor. The current foot configuration does not include pressure sensors, so the odometry data is purely based off servo measurements. It is worth mentioning that the swaying motion that occurs when walking can potentially assist monocular depth perception.

[2] https://github.com/raulmur/ORB_SLAM2.

[3] http://nimbro.net/Humanoid/robots.html.

3.2 Data Collection

The data collection focused on recording the keyframe trajectory produced by ORB-SLAM, timing data, and truth data from an infrared camera based Motion Capture system set up in the lab. In the first experiment, the iGus walked in a 3 m by 2 m rectangular path, performing a loop closure once the rectangle was complete. In the second experiment, the iGus walked forwards for 2 m, then was picked up by the robot handler and moved rapidly back to a position a little to the left of the starting position (also received a 360° rotation), where it walks forward a little distance. This procedure is to simulate handling of the robot during a soccer match and tests the ability of ORB-SLAM to handle the kidnapped robot problem where a robot is lifted and moved to an unknown new location.

4 Results

With our walk engine running, ORB-SLAM ran on the iGus at an average frame rate of 20 frames/second (standard deviation of 1.7) before an initial map had been created, and an average of 26 frames/second (standard deviation of 5.2) afterwards (see Fig. 1). When the keyframe trajectory data is compared to the truth data, ORB-SLAM tracks the movement of the robot with a level of accuracy which is acceptable for a robot soccer application (see Fig. 2). In the kidnapped robot experiment (see Fig. 3), ORB-SLAM was able to realize it had been placed down in a familiar location. While ORB-SLAM was not able to track the trajectory of the carried segment, as soon as it was put down, it was able to recognize its location and resume tracking.

5 Discussion

The results demonstrated that a medium sized humanoid robot with a NUC7i7 processor is capable of running the current state-of-the-art ORB-SLAM in real

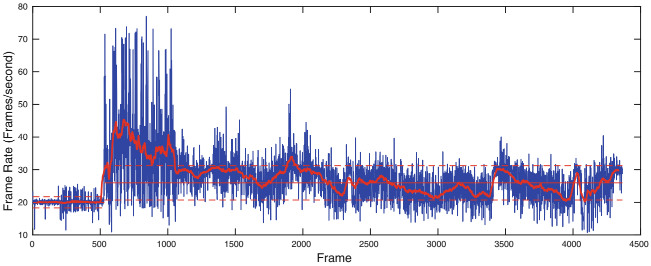

Fig. 1. Frame rate data from the first experiment (see Fig. 2). The blue line is the instantaneous frame rate of each frame, the bold red line is a 50 frame moving average, and the horizontal solid red lines bounded by dashed red lines show the average frame rate before and after map initialization, along with their standard deviations. (Color figure online)

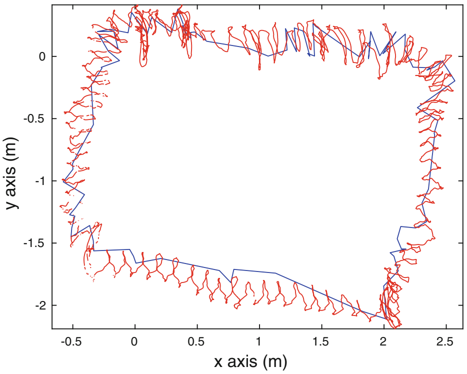

Fig. 2. First experiment trajectory as reported by ORB-SLAM keyframes in blue, and the motion capture system in red. The path walked is a 3 m by 2 m rectangle with a loop closure at the end. Notice the sway in the red trajectory, which represents the lateral movement of the robot's head as it walks. (Color figure online)

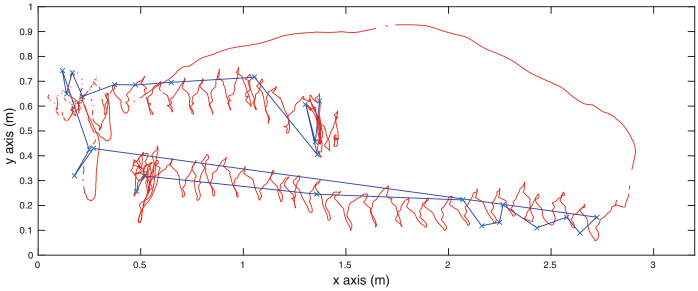

Fig. 3. Second experiment trajectory as reported by ORB-SLAM keyframes in blue, and the motion capture system in red. Occasional missing parts of the red trajectory are due to the robot handler temporarily blocking the view of the motion capture cameras. The robot first walked a straight line (0.5,0.3) to (2.7,0.15) before being picked up by handler and moved rapidly and disorientatingly to a new position (0.2,0.6) which was close to position it had seen before. (Color figure online)

time, was able to handle the swaying motion and could recover from a typical kidnapped robot situation. However several advantages and disadvantages that should be weighed before implementing ORB-SLAM onto a humanoid robot. An average frame rate of 20 fps was achieved before the initial map was created, rising to 26 fps afterwards, leaving plenty of computational resources for other system components to run.

Now that the basic reliability of ORB-SLAM has been observed, the authors intend to address some of the limitations of this visual SLAM method. The maps and trajectory ORB-SLAM produces are not referenced to the real world in any way, so for ORB-SLAM to be useful for localization in a known environment like RoboCup additional known feature extractors like goal detectors would need to be used. Additionally, while ORB-SLAM is resistant to objects moving within its environment, it is unknown how ORB-SLAM degrades when placed in crowded dynamic environments like RoboCup.

6 Conclusion

The objective of this research was to investigate the practicality of implementing the state-of-the-art monocular ORB-SLAM onto a medium sized humanoid robot. To the best of our knowledge, monocular ORB-SLAM has not been implemented on humanoid robots, with its unique locomotion challenges of swaying and jarring movements. We provided an evaluation of ORB-SLAM and detailed the process undertaken to port it onto an iGus humanoid robot intended for the robot soccer competition RoboCup and found that ORB-SLAM was able to run at 26 fps while the robot was walking. ORB-SLAM was able to successfully provide accurate localization to the robot during two experiments that tested for loop closure and relocalization.

References

1. Engel, J., Koltun, V., Cremers, D.: Direct sparse odometry. IEEE Trans. Pattern Anal. Mach. Intell. **40**(3), 611–625 (2018)
2. Engel, J., Schöps, T., Cremers, D.: LSD-SLAM: large-scale direct monocular SLAM. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8690, pp. 834–849. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10605-2_54
3. Forster, C., Zhang, Z., Gassner, M., Werlberger, M., Scaramuzza, D.: SVO: semidirect visual odometry for monocular and multicamera systems. IEEE Trans. Robot. **33**(2), 249–265 (2017)
4. García, S., López, M.E., Barea, R., Bergasa, L.M., Gómez, A., Molinos, E.J.: Indoor SLAM for micro aerial vehicles control using monocular camera and sensor fusion. In: 2016 International Conference on Autonomous Robot Systems and Competitions (ICARSC), pp. 205–210, May 2016
5. Houliston, T., et al.: NUClear: a loosely coupled software architecture for humanoid robot systems. Front. Robot. AI **3**, 20 (2016). https://doi.org/10.3389/frobt.2016.00020
6. Mur-Artal, R., Montiel, J.M.M., Tardós, J.D.: ORB-SLAM: a versatile and accurate monocular SLAM system. IEEE Trans. Robot. **31**(5), 1147–1163 (2015)
7. Oriolo, G., Paolillo, A., Rosa, L., Vendittelli, M.: Humanoid odometric localization integrating kinematic, inertial and visual information. Auton. Robot. **40**(5), 867–879 (2016)
8. Scona, R., Nobili, S., Petillot, Y.R., Fallon, M.: Direct visual SLAM fusing proprioception for a humanoid robot. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1419–1426, September 2017
9. Song, Y., Nuske, S., Scherer, S.: A multi-sensor fusion MAV state estimation from long-range stereo, IMU, GPS and barometric sensors. Sensors **17**(1), 11 (2016)
10. Whelan, T., Salas-Moreno, R.F., Glocker, B., Davison, A.J., Leutenegger, S.: ElasticFusion: real-time dense SLAM and light source estimation. Int. J. Robot. Res. **35**(14), 1697–1716 (2016)