MOCAP4ROS2: An Open Source Framework for Motion Capture Systems in Robotics

Francisco Martín
francisco.rico@urjc.es
Intelligent Robotics Lab, Universidad Rey Juan Carlos
Fuenlabrada, Spain

José Miguel Guerrero
josemiguel.guerrero@urjc.es
Intelligent Robotics Lab, Universidad Rey Juan Carlos
Fuenlabrada, Spain

A. Alberto García
alberto.gomezjacinto@urjc.es
Intelligent Robotics Lab, Universidad Rey Juan Carlos
Fuenlabrada, Spain

Francisco J. Rodríguez
fjrodl@unileon.es
Robotics group, Escuela de Ingenierías Industrial, Informática y Aeroespacial, Universidad de León
León, Spain

Vicente Matellán
vicente.matellan@unileon.es
Robotics group, Escuela de Ingenierías Industrial, Informática y Aeroespacial, Universidad de León
León, Spain

ABSTRACT

Motion Capture systems are crucial in many fields, and Mobile Robotics is one of them. This paper describes an Open Source robotic framework to standardize the use of motion capture systems called MOCAP4ROS2. This framework features a layered architecture that allows building applications that use Motion Capture systems regardless of the specific system model/vendor. The challenges are technical and social: on the one hand, resolving synchronization and representation issues; on the other hand, involving the community to reach a consensus on the necessary interfaces. MOCAP4ROS2 has been implemented in ROS2 and already has drivers (we understand a driver for MOCA4ROS2 as a ROS2 node that publish the MOCAP system information) for today's main commercial systems.

KEYWORDS

Open Source Robotics, Motion Capture Systems, Open Frameworks.

ACM Reference Format:
Francisco Martín, José Miguel Guerrero, A. Alberto García, Francisco J. Rodríguez, and Vicente Matellán. 2022. MOCAP4ROS2: An Open Source Framework for Motion Capture Systems in Robotics. In *The 18th International Symposium on Open Collaboration (OpenSym 2022), September 7–9, 2022, Madrid, Spain.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3555051.3555076

1 MOTIVATION

Motion Capture (MOCAP) Systems play an essential role in Mobile Robotics [4]. When evaluating location algorithms [1], they provide ground truth or replace location modules to reduce complexity when evaluating navigation algorithms[6]. In the case of aerial robots, many of the advances in flight control in individual robots and fleets are based on an accurate and real-time motion capture system [5]. These are just a couple of examples, but, in general, obtaining the reliable position of elements in the environment provides much valuable information for robots that have to move, interact, and function in real space.

ROS [2] is a *de facto* standard in Robotics. It is an Open Source robot programming framework that provides libraries, tools, and infrastructure. ROS applications are built as computing networks of nodes (programs) that collaborate through different communication paradigms (fundamentally publication/subscription) to carry out a task in a distributed environment.

One of its key aspects in ROS success is standardization: A consensus has been reached on the formats of the exchanged messages. For example, the nodes that provide information about a laser (laser drivers) always publish the same format. This approach allows drivers, tools, and programs to connect at deployment. Another critical aspect is the community: thousands of developers produce software packages in a federal development model in which the Open Source Robotics Foundation provides tools, documentation, and infrastructure to create the packages that are part of the distributions.

Recently, ROS2[3] is a refactor of ROS, providing multiplatform features, security, real-time, and a maturity[1] demanded by an industry willing to incorporate ROS in its developments.

In this paper, we propose MOCAP4ROS2[2], a framework built on top of ROS2, providing drivers, libraries, and tools for configuring and using MOCAP systems in Robotics. Although there are already MOCAP system drivers, each has its tools, formats, and procedures, and users of a specific MOCAP cannot benefit from the developments of a different one.

In this sense, MOCAP4ROS2 represents an unprecedented step in the standardization of the software that uses motion capture systems. We have not been able to find either in the literature or in any repository initiatives, software, or documents that have tried to standardize the aforementioned motion capture system. This field

OpenSym 2022, September 7–9, 2022, Madrid, Spain

ACM ISBN 978-1-4503-9845-9/22/09…$15.00
https://doi.org/10.1145/3555051.3555076

[1] https://docs.ros.org/en/dashing/index.html
[2] https://github.com/MOCAP4ROS2-Project

was dominated by system manufacturers, who seem unwilling to lose their commercial niche in applications that exploit their systems. For this reason, it is not possible to compare MOCAP4ROS2 with similar initiatives or systems.

The main objective being pursued is to create a community of users and developers around this open-source framework so that the efforts of some can be used by others, providing common standards that guide development. Some projects tried to afford this problem in the previous version of ROS1, and they were focused on how to run different MOCAP Systems, but not on standardization[3].

2 FRAMEWORK DESCRIPTION

MOCAP4ROS2 provides a layered architecture (Figure 1). Each layer is responsible for one aspect of using a MOCAP system:

- **Drivers Layer**: In this layer are the drivers of the MOCAP system. Not only are systems based on cameras and markers taken into account, but also those based on inertial sensors (IMUs) or any other movement or position capture system. For each type of MOCAP, a standard format of information that the drivers must produce will be defined and subject to debate in the MOCAP4ROS2 community.
- **Composer Layer**: This layer contains nodes common to each type of MOCAP and combines their information into a common axis of geometric references, using the standard TF system that ROS already uses to model inter/intra robot geometric relationships. For example, IMU-based systems require additional information, such as fixed relationships in a set of sensors attached, so that angular information can be translated into geometric positions.
- **Application Layer**: This layer contains programs that access the lower layers to control, visualize, and collect data or any application.
  The philosophy of MOCAP4ROS2 is that any application developed in this framework, distributed in its repositories, can be used by users with other types or models of MOCAP systems, following a federated Open Source development scheme. There are several challenges that MOCAP4ROS2 must address, such as timestamp synchronization between systems, the geometric relationship between systems running at the same time, or start-up and shutdown procedures.

MOCAP4ROS2 can merge the information coming from several MOCAP systems in the same reference axes so that the information coming from them and the sensory information from the robot can be spatially reasoned.

3 RESULTS

In the MOCAP4ROS2 repository, different packages include:

- Drivers for Vicon[4] and Optitrack[5] systems, as well as Technaid[6] IMUs.

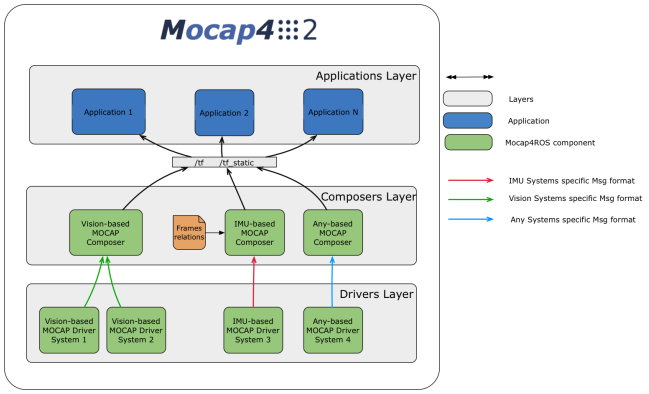

Figure 1: MOCAP4ROS2 Layered Architecture.

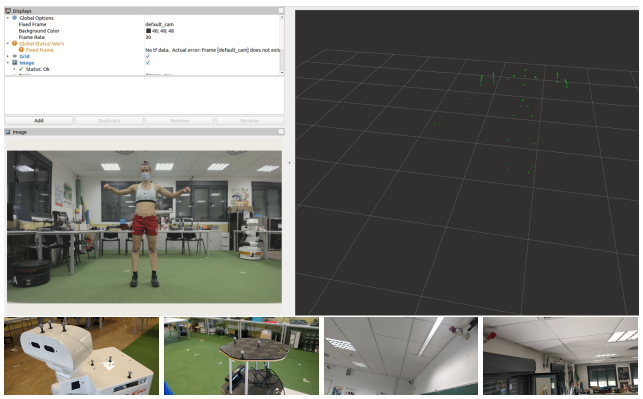

Figure 2: MOCAP4ROS2 running with sensors attached to a human, using the visualization RViz tool (up row). Setup for mobile robots and global setup (bottom row)

- A control application[7] that allows controlling the starting and stopping of MOCAP systems and verifying the synchronization of their clocks.
- Mechanisms to bridge to use MOCAP systems implemented in ROS1.

The MOCAP4ROS2 developer community is still tiny, as seen in each repository, although the first versions of the drivers have already been provided by developers external to the project.

MOCAP4ROS2 is used within the Eurobench project in its Biosensors facility. This project evaluates exoskeletons and humanoids in different testbeds, such as sensorized doors, configurable stairs and ramps, and irregular terrain. These benchmarks use ROS and are geometrically modeled using the ROS TF system. Using MOCAP4ROS2 allows them to relate the elements of the testbed to the robot and thus obtain complete experimental data.

4 CONCLUSION AND IMPACT

MOCAP4ROS2 is an Open Source framework from MOCAP systems for ROS2. It has drivers for the most common commercial camera-based MOCAP systems and supports other types of systems. It

[3]https://github.com/tuw-cpsg/tuw-cpsg.github.io
[4]https://github.com/MOCAP4ROS2-Project/mocap4ros2_vicon
[5]https://github.com/MOCAP4ROS2-Project/mocap4ros2_optitrack
[6]https://github.com/MOCAP4ROS2-Project/mocap4ros2_technaid
[7]https://github.com/MOCAP4ROS2-Project/device_control

has tools for controlling an environment where multiple MOCAP systems are integrated and are fully functional (Figure 2).

MOCAP4ROS2 standards have been defined for the generation of MOCAP system information, fostering the standardization of this type of system in an Open Source model, where the community makes decisions after a technical debate[8].

However, the upcoming challenges that MOCAP4ROS2 will face in this standardization process involve heterogeneous APIs that produce data that, although mostly overlapping, have some distinctive components. Our objective is to unify the information they produce, reducing the particularities to a minimum. In this way, the applications created within MOCAP4ROS2 will be able to work with any MOCAP vendor.

We are convinced that this framework will facilitate this type of system in Robotics. We rely on this federated development model based on Open Source to guarantee the future of this framework beyond the period in which it was funded.

ACKNOWLEDGMENTS

MOCAP4ROS2 is a FTP funded by EU ROSIN, which has received funding from the European Union‘s Horizon 2020 research and innovation programme under grant agreement No 732287.

REFERENCES

[1] Dudzik, S.: Application of the Motion Capture System to Estimate the Accuracy of a Wheeled Mobile Robot Localization. Energies. vol. 13 (2020). doi: 10.3390/en13236437

[2] Quigley, M., Gerkey, B., Conley, K., Faust, J. Foote, T., Leibs, J., Berger, E., Wheeler, R., Ng, A.: ROS: an open-source Robot Operating System. In: Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA). Workshop on open source software, vol. 3, p. 5 (2009)

[3] Macenski, S., Foote,T., Gerkey,B., Lalancette,C., Woodall, W.: Robot Operating System 2: Design, architecture, and uses in the wild, Science Robotics vol. 7, pp. 1-12 (2022). doi: 10.1126/scirobotics.abm6074

[4] Field, M., Stirling,D., Naghdy, F., and Pan, Z.: Motion capture in robotics review. In: 2009 IEEE International Conference on Control and Automation. pp. 1697-1702 (2009). doi: 10.1109/ICCA.2009.5410185

[5] Gargioni, G., Peterson, M., Persons, J., Schroeder, K., Black, J.: A Full Distributed Multipurpose Autonomous Flight System Using 3D Position Tracking and ROS. International Conference on Unmanned Aircraft Systems (ICUAS). pp. 1458-1466 (2019). doi: 10.1109/ICUAS.2019.8798163

[6] Engel, J., Sturm, J., Cremers, D. : Scale-aware navigation of a low-cost quadrocopter with a monunocular camera. Robotics and Autonomous Systems, vol. 62, pp. 1646-1656 (2014). doi: 10.1016/j.robot.2014.03.012

[7] Torricelli, D., Pons, J.L.: EUROBENCH: Preparing Robots for the Real World, In: Carrozza M., Micera S., Pons J. (eds). Wearable Robotics: Challenges and Trends. WeRob 2018. Biosystems & Biorobotics, vol. 22, pp. 375-378. Springer (2018). doi: 10.1007/978-3-030-01887-0_72

[8] https://discourse.ros.org/t/mocap4ros2-motion-capture-systems-in-ros2/12308