



HAL
open science

ICP Localization and Walking Experiments on a TALOS Humanoid Robot

Thibaud Lasgaignes, Isabelle Maroger, Maurice Fallon, Milad Ramezani, Luca Marchionni, Olivier Stasse, Nicolas Mansard, Bruno Watier

► **To cite this version:**

Thibaud Lasgaignes, Isabelle Maroger, Maurice Fallon, Milad Ramezani, Luca Marchionni, et al.. ICP Localization and Walking Experiments on a TALOS Humanoid Robot. 20th International Conference on Advanced Robotics (ICAR 2021), Dec 2021, Ljubljana (en ligne), Slovenia. 10.1109/ICAR53236.2021.9659474 . hal-03482225

HAL Id: hal-03482225

<https://laas.hal.science/hal-03482225v1>

Submitted on 15 Dec 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

ICP Localization and Walking Experiments on a TALOS Humanoid Robot

T. Lasguignes¹, I. Maroger¹, M. Fallon², M. Ramezani², L. Marchionni³, O. Stasse^{1,4}, N. Mansard^{1,4}, B. Watier⁵

Abstract—This system paper describes the integration and the evaluation of an ICP-based localization system on the TALOS humanoid robot. The new generation of flash LiDAR systems, here an Ouster OS1-64, have made it possible to obtain 3D clouds at 10 Hz. Coupled with an Intel RealSense T265 providing visual-inertial odometry it is possible to localize the robot and use this information to generate foot steps in real time to reach specific points. The approach is validated with a Qualisys motion capture system. It is also used to generate real-time walking motion on the TALOS humanoid robot. This paper is an integration paper showing that it is now feasible to accurately guide a humanoid robot in an environment in real time using a LiDAR system.

Index Terms—LiDAR-based localization, TALOS humanoid robot, walking

I. INTRODUCTION

In order to enable humanoid robots to autonomously perform useful behaviors in a semi structured environment, it is necessary to provide them with a way to autonomously localize themselves in a 3D environment. This is necessary to inspect specific important points in a factory, or to execute some manipulation tasks such as those performed during the DARPA Robotics Challenge (DRC). The difficulty is to maintain a sufficient precision for the targeted tasks while being in an environment whose parts are changing due to human activities. A similar problem occurs for autonomous cars navigating in a city. Indeed even if the architectural elements are consistent, the traffic is constantly perturbing the laser measurements. It is therefore important to have a robust and yet efficient way to localize the robot in a large 3D environment and to update the current status of the map.

In this paper, the problem we are interested in is to assess the localization of the humanoid robot TALOS [1] such that it is able to perform a behavior such as manipulating an object or walking over debris. The behaviors can be adaptive through perception, control and planning, and they may have various requirements in terms of localization precision.

Finally, the capability to embed the localization software in the robot to make it autonomous is an important aspect to consider.

M. Fallon, L. Marchionni, O. Stasse and N. Mansard acknowledge the H2020 Memmo project, T. Lasguignes acknowledges the ROB4FAM project. I. Maroger acknowledges the ANR Cobot (ANR-18-CE10-0003).

¹ LAAS-CNRS, Université de Toulouse, France

² Oxford Robotics Institute, Univ. of Oxford, UK

³ PAL-Robotics, Barcelona, Spain

⁴ Artificial and Natural Intelligence Toulouse Institute, France

⁵ LAAS-CNRS, Université Toulouse 3 - Paul Sabatier, Université de Toulouse, France

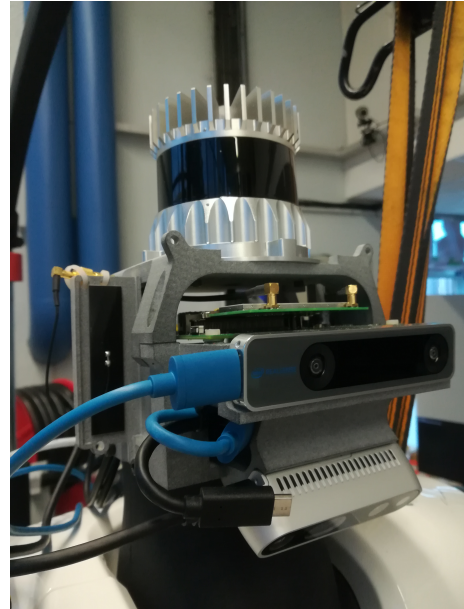


Fig. 1. TALOS equipped with an Ouster OS1-64 LiDAR, an Intel RealSense T265 tracking camera and an Intel RealSense D435i RGB-D camera

A. State of the art

The research community working on the Simultaneous Localization And Mapping (SLAM) problem is very active and consider many instances of the problem. Historically, for humanoid robots, it is based on computer vision and is targeting affordable perception systems and therefore uses monocular camera [2]. Since then, the DRC has popularized multiple sensors systems including stereo camera and LiDAR [3] such as the MultiSense SLB product from Carnegie Robotics, that was recently incorporated in the Walkman humanoid robot from IIT [4].

In [5], a laser-based localization was proposed for a NAO robot in a small multi-level indoor environment, later combined with a monocular camera in [6]. This localization was based on the Monte Carlo Localization technique presented in [7]. Nowadays the general framework of factor graph [8] allows to represent coherently the relationships between the robot state, its control vector and measurements of various sensors. Using this representation, it is possible to utilize a general non-linear solver to solve localization, calibration, state estimation or map building problems. However, sensors have different measurement frequencies and different memory complexities. For instance, in this setting the IMU

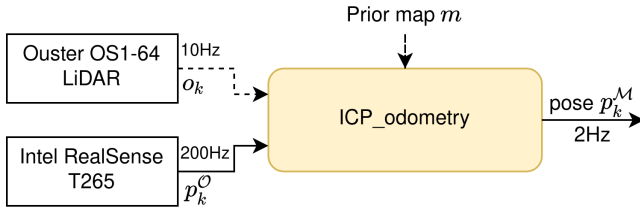


Fig. 2. Block diagram of the localization system, dash-arrows show the flow of point clouds.

provides 6 real accelerations and gyrometer measurements at 1kHz, while the LiDAR provides 65536 range measurements at only 10Hz. They need to be preprocessed in different ways before being incorporated in such representations.

For LiDAR-based measurements, Iterative Closest Point (ICP) and 3D points accumulation maps are efficient methods to propose such pre-processing [9]. Such methods have been recently refined and deployed in the field of legged robots and more precisely for quadruped robots [10]. Indeed, it is now possible to find reliable industrial robots such as ANYmal from ANYbotics or Spot from Boston Dynamics. This paper focuses on the practical implementation of such subsystem and its implementation on a research humanoid robot platform, here the TALOS robot from the company PAL-Robotics. Similar work has been realized using a Dense SLAM system [11], with a HRP-4 humanoid robot and an off-board computer equipped with a GPU. In this work we focus on the localization performance using an onboard computation.

In this work, we want to localize the TALOS humanoid robot in an indoor industrial-like environment. We use a modification of the initial TALOS humanoid robot head design including a flash LiDAR system.

B. Problem

For a given map m , the k -th LiDAR measurement $o_k \in \mathbb{R}^{64 \times 1024}$, and the Visual Inertial Odometry (VIO) pose estimation p_k^O in the fixed odometry frame \mathcal{O} , the goal of the localization system is to find the pose $p_k^M \in SE(3)$ of the head in the map frame \mathcal{M} such that $p_k^M = ICP(m, o_k, p_k^O)$. The overall scheme is presented in Fig. 2.

C. Contributions

The contributions of this paper are the following:

- Successfully implement a LiDAR-based localization combined with a Visual-Inertial Odometry on a TALOS humanoid robot in order to accurately guide it. All computations are performed in real time on the robot's computer.
- Benchmark the localization system against a Motion Capture system.

II. LOCALIZING IN THE MAP

Fig.2 shows the localization pipeline implemented on the robot that will be presented in the following section.

A. Iterative Closest Point

The ICP algorithm [12] is used to register two point clouds with overlapping parts. It has 4 main steps: pre-filtering, correspondence estimation, outlier filtering and lastly error-minimization. The main issue of this algorithm is the high influence of outliers that can result in convergence to a local minimum.

B. ICP-based localization

We use the localization system from [13] and [10] over a known map.

The Autotuned-ICP (AICP) is an ICP registration method that adjusts the outlier filter of the ICP basis by computing an overlap parameter between the reading cloud and the reference cloud. It allows the registration of consecutive point cloud reads to a reference cloud, which in our case will be a pre-built map.

This system uses the ICP implementation proposed by [9] and publicly available under the name *libpointmatcher*¹.

The system needs to be initialized. For this paper, this pose is given by an approximation of the motion capture estimation. Further researches will be conducted on the estimation of this initial pose based on the map and LiDAR data.

At each subsequent steps the ICP refine the transformation between the measure o_k and the map m . As an estimation of the robot's pose p_k^O is given by the VIO, the measure o_k can be expressed in the \mathcal{O} frame. The robot's pose in the map frame \mathcal{M} is then obtained as $p_k^M = c_k p_k^O$, with c_k the alignment transformation between o_k^O and m computed by the ICP.

C. Visual-Inertial Odometry initialization

Kinematic-Inertial Odometry is effective for high-frequency estimation over a short time interval, whereas Visual-Inertial Odometry, through the use of lower frequencies exteroceptive sensors, remains reliable over a longer period. As both odometries are available on the robot and the experiments do not require a high-frequency state estimation, it was decided to use the Visual-Inertial Odometry instead of the Kinematic-Inertial Odometry, used in the initial system [10], [13].

The Visual-Inertial Odometry is given by an Intel RealSense T265 tracking camera with a frequency of 200 Hz. Fig.3 shows the comparison of the VIO to the Motion Capture estimation over 70 seconds, starting from the same initial point.

D. Building the map

In this paper, the indoor environment is known through a given map. In the following sections, different maps are discussed. Some of these maps were tested and compared in the experiment described in Sec.IV.

¹<https://github.com/ethz-asl/libpointmatcher>

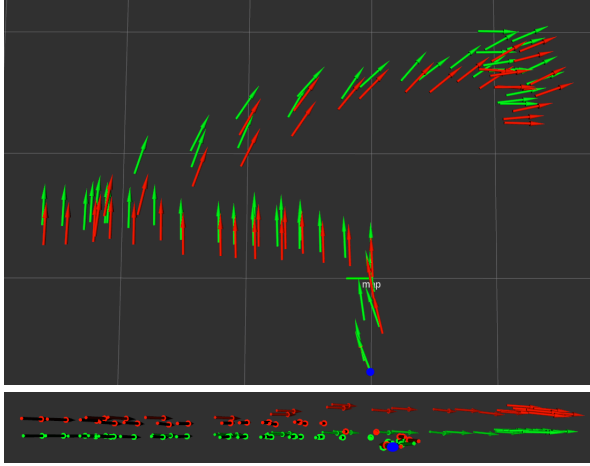


Fig. 3. Downsampled visualization of the Visual-Inertial Odometry (red) and the Motion Capture estimation (green) over the two first targets. Top: view on the (x, y) plane, x upward. Bottom: view on the (z, y) plane, z upward.

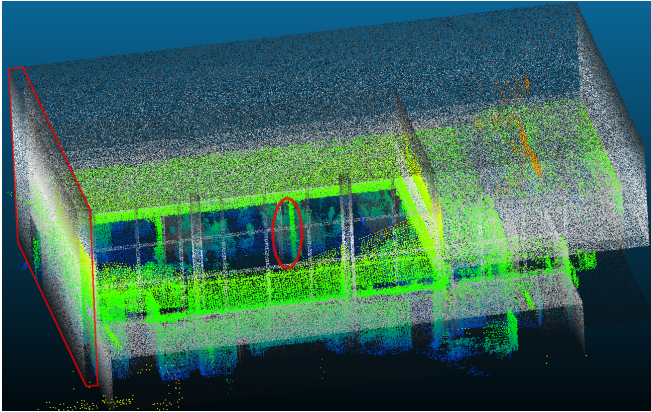


Fig. 4. Comparison between a 3D model (gray scale) and an accumulation of LiDAR scans (color scale). Examples of occluded walls and errors are highlighted.

1) *Architectural 3D model:* Starting directly from an architectural 3D model can be a source of errors. Indeed, as shown in Fig.4, an architectural 3D model can contain occluded walls, e.g. on the left of the figure, or small errors, e.g. the pillars in the middle. When comparing with the points accumulated using the ICP algorithm, mismatches appear with these features and may disrupt the convergence. As the walls in this kind of models are usually parallel in pairs, they create local minima.

A solution to that problem is to use a prior map that has been extracted from the 3D model, in order to remove the occlusions. It can be done manually or by mean of a simulation, provided that the sensor model is known. In this work, it was decided to remove the occluded parts manually to compare with the other proposed map introduced in the following section.

2) *Using a scanning device:* The map can be built prior to the robot localization using a scanning device. As no such specific device was available, it was decided to use the data

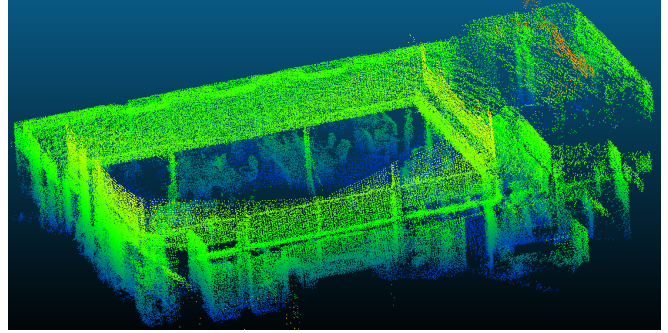


Fig. 5. Map obtained by accumulating multiple LiDAR scans from multiple positions and orientations.

returned by the LiDAR on the new head of TALOS. Fig.5 displays the map.

This kind of model can present some missing parts, e.g. in this experiment some parts of the floor and the ceiling are missing from the map, but it is easier to produce than an architectural map if an appropriate device is available.

III. WALKING

In order to assess the efficiency of the ICP-based localization system, experiments where the robot has to walk towards multiple given positions were performed. For these experiments, the walking algorithm used to generate the motion is the one provided by PAL-Robotics. It was used to send a velocity command to the robot on the topic `/walking_controller/cmd_vel` at a 2Hz frequency. The velocity command was merely computed as the distances along the x and y axis in the coordinate frame linked to the robot and angle between the robot and its goal orientation around the z axis. Thresholds of 0.1m/s for the linear velocities and 0.12 rad/s for the angular velocity were applied on the command in order to not reach the robot limits. Moreover, the command was designed so that the robot stops walking when it is at small enough distances and angle from its target, respectively 0.08 m and 0.07 rad. This stopping distance was implemented to prevent the robot to trample on its target instead of stopping on it, this allows swift stop once on spot. Once the robot stops on a target, the robot is programmed to stay still during 8 s before moving on to its next target.

IV. EXPERIMENTS ON THE ROBOT

A. Experimental setup

The experimental room is equipped with a Motion Capture system (MoCap) including 20 infrared Qualisys Miquis M3 cameras sampling at up to 650Hz with a 3×10^{-4} m accuracy on the viewed area. This motion capture system was used to record the positions and orientations of the robot's head. We placed 5 passive markers on its torso and 1 on the top of its head. In this study, the MoCap measurements are hypothesized to be the ground truth data. Those measurements were compared to the ones made with

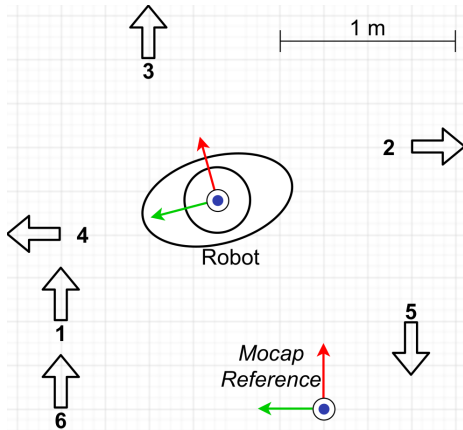


Fig. 6. Goal positions in the experimental room

the ICP-based localization system in order to assess its accuracy and its performances.

A new head was designed for the TALOS robot and is shown on Fig.1. From top to bottom, it is equipped with an Ouster OS1-64 LiDAR, an Intel RealSense T265 and an Intel RealSense D435i. The latter is not used in the experiment.

The LiDAR has a range between 0.8m to 120m, an horizontal field-of-view of 360° and a vertical field-of-view of 33.2° using its 64 laser beams. For the experiment, the LiDAR was set to take 1024 horizontal samples per scan and to return point clouds at 10 Hz.

The Intel RealSense T265 is a tracking camera embedding two Fish-Eye cameras, an IMU and a Vision Processing Unit (VPU). This camera runs a V-SLAM algorithm on the VPU with an output frequency of 200 Hz.

B. Experimental protocol

The performed experiment consists of a series of targets the robot has to successively reach. Thus, six goal positions were defined in the experimental room. Those positions are represented on Fig.6. They have been chosen in such a way that the robot ranges the whole MoCap viewed area and faces diverse orientation changes. The position of the robot at the beginning of the experiment is not defined in advance, it could be anywhere in the viewed area. Note that, if the robot starts from the origin of the MoCap reference frame, it will have to travel around 10.6 m to perform the whole experiment.

During the experiments, a dataset was recorded on a *rosbag* including the measurements made by the MoCap, the velocity orders sent to the robot, the LiDAR data, the tracking camera images and estimations and the ICP-based localization results.

This experiment was successfully performed 3 times with a TALOS robot. Due to technical problems and hardware

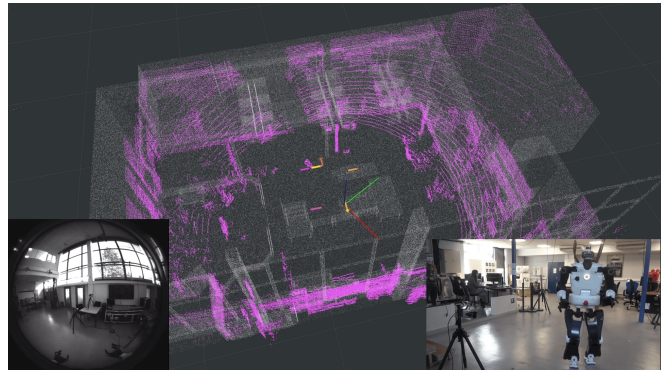


Fig. 7. Frame extracted from the video of the first experiment. The video shows the ICP data (gray: prior map, pink: read point cloud aligned, axis: estimated pose) and the targeted poses (arrows) in the middle, the image from a sensor of the Intel RealSense T265 on the left and the video of the experimental room and the robot on the right.

issues on the robot legs and ankles, further experiments resulted in failures and the fall of the robot, preventing the authors to gather more data. Among those successfully performed experiments, the first one was performed with the map built from the architectural 3D model (Sec.II-D.1) whereas the second and the third ones were performed using the map built by using a scanning device (Sec.II-D.2).

A screenshot from the video of the first experiment is shown in Fig.7. A video explaining the experimental protocol and showing the second experiment is available at <https://youtu.be/0t1bBjDTqMA>.

V. DISCUSSION

A. Comparison between the motion capture system and the ICP-based localization system

During the experiments on the robot, two data sets were collected simultaneously: one with the positions and orientations of the robot head recorded by the MoCap and another recorded by the ICP localization system. As both localization systems do not have the same sampling frequencies, the raw datasets cannot be compared directly. Indeed, the MoCap dataset includes around 40 times the amount of data of the ICP one as Fig.8 shows. This is why both data sets were interpolated in order to have the same length.

First of all, for the three datasets, a delay between the ICP system and the MoCap can be measured shifting the two datasets to minimize the difference between the (x, y, z) positions and the θ orientation measured by the two localization systems. This delay scores between 0.51s and 0.62s depending on the experiment and is mainly due to the computation time of the ICP. In the following, this delay has been removed from the ICP set.

Then, the differences between the two datasets on the position and on the orientation can be computed. These differences will now be referred as offsets. The evolution of



Fig. 8. Visualization of the recorded data on Rviz (in red the MoCap set and in green the ICP set)

	On x (mm)	On y (mm)	On z (mm)	Around z (rad)
Exp. 1	18.7 ± 17.2	24.9 ± 19.3	9.2 ± 7.2	0.025 ± 0.019
Exp. 2	14.8 ± 17.3	19.9 ± 17.8	5.4 ± 4.4	0.023 ± 0.027
Exp. 3	13.8 ± 15.9	18.9 ± 15.7	5.6 ± 4.8	0.024 ± 0.027

TABLE I

COMPUTED ERRORS BETWEEN THE MOCAP AND THE ICP DATA SETS FOR ALL THE EXPERIMENTS

these offsets over the whole experiments is shown in Fig.9. These offsets are of two kinds :

- Structural offsets due to a difference between the coordinate frames of both localization systems. We make the assumption that these offsets are constant as it seems to be the case in Fig.9 and can easily be computed as the mean of the differences between the datasets over the experiment duration. Thus, these offsets score 0.19 m, 0.058 m, 0.097 m respectively along the x , y and z axis and 1.12×10^{-4} rad around the z axis for the first experiment. These offsets are lower for the second and the third experiment (respectively 0.045 m, 0.021 m, 0.058 m and 0.048 rad) because the coordinate systems were reset after the first experiment. In the following, these structural offsets are considered equal to zero as they have been removed to the ICP dataset as the MoCap was taken as the ground truth. The results after removing these offsets are represented in Fig.10.
- Errors due to real differences of results between the two localization systems. They can be computed as the mean of the differences between the two datasets after removing the structural offset. These errors are presented in Tab.I.

During the 3 experiments, the robot started from 3 different positions with diverse orientations. As the results shown in Tab.I are similar regardless of the experiment, we can conclude that the ICP-based localization system does not depend on the starting pose. Moreover, the robot traveled different distances according to the experiments, respectively 8.93 m, 11.05 m and 10.69 m. As Fig.10 shows, the error

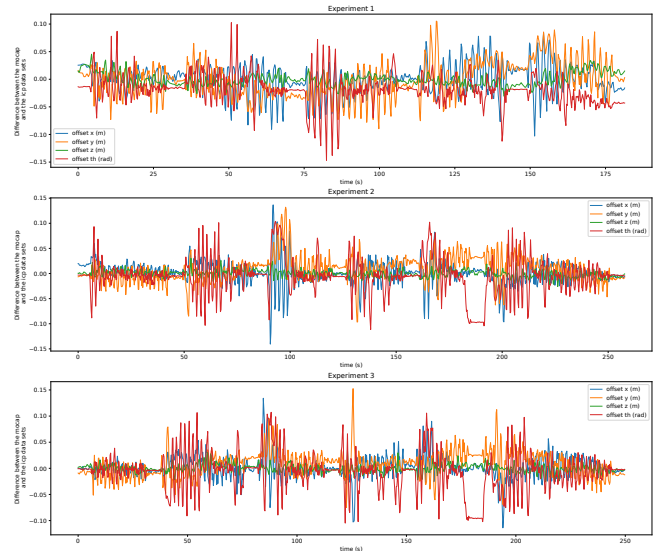


Fig. 9. Errors between the 2 datasets during the 3 experiments (x in blue, y in orange, z in green and θ in red)

does not seem to increase with the traveled distance, at least for distances around 10 m .

B. Accuracy on the goal positions

The goal positions can easily be identified on the datasets as they match with the points with zero velocity. To assess the accuracy of the ICP-based localization system on achieving the goal positions, the difference between the measured position and the desired goal position can be computed. The absolute average error on the goal positions (\pm standard deviation) is 0.025 ± 0.016 m on the x axis, 0.039 ± 0.042 m on the y axis and 0.028 ± 0.012 rad around the z axis. All these errors are lower than the tolerance admitted on stops stated in Sec.III which means that the accuracy of the ICP localization system is satisfactory for the targeted application. Moreover, with such accuracy, more complex localization tasks could be considered. In future work, the ICP-based localization system will be used to place the robot in front of an object to perform a task requiring accuracy such as drilling or crossing debris.

C. Effect of the map on the ICP-based localization system

As Tab.I shows, there is no significant difference (less than the standard deviations) in the errors between the first experiment and the second and the third experiments whereas the ICP-based localization system was performed with different maps. This means that, even if the map built from the architectural 3D model is a priori less accurate than the map built using a scanning device, the ICP localization system has an accurate behavior in both cases.

VI. CONCLUSION

We presented a pipeline to localize the TALOS robot in an indoor environment based on LiDAR and Visual-Inertial Odometry. The VIO is used to initialize the ICP

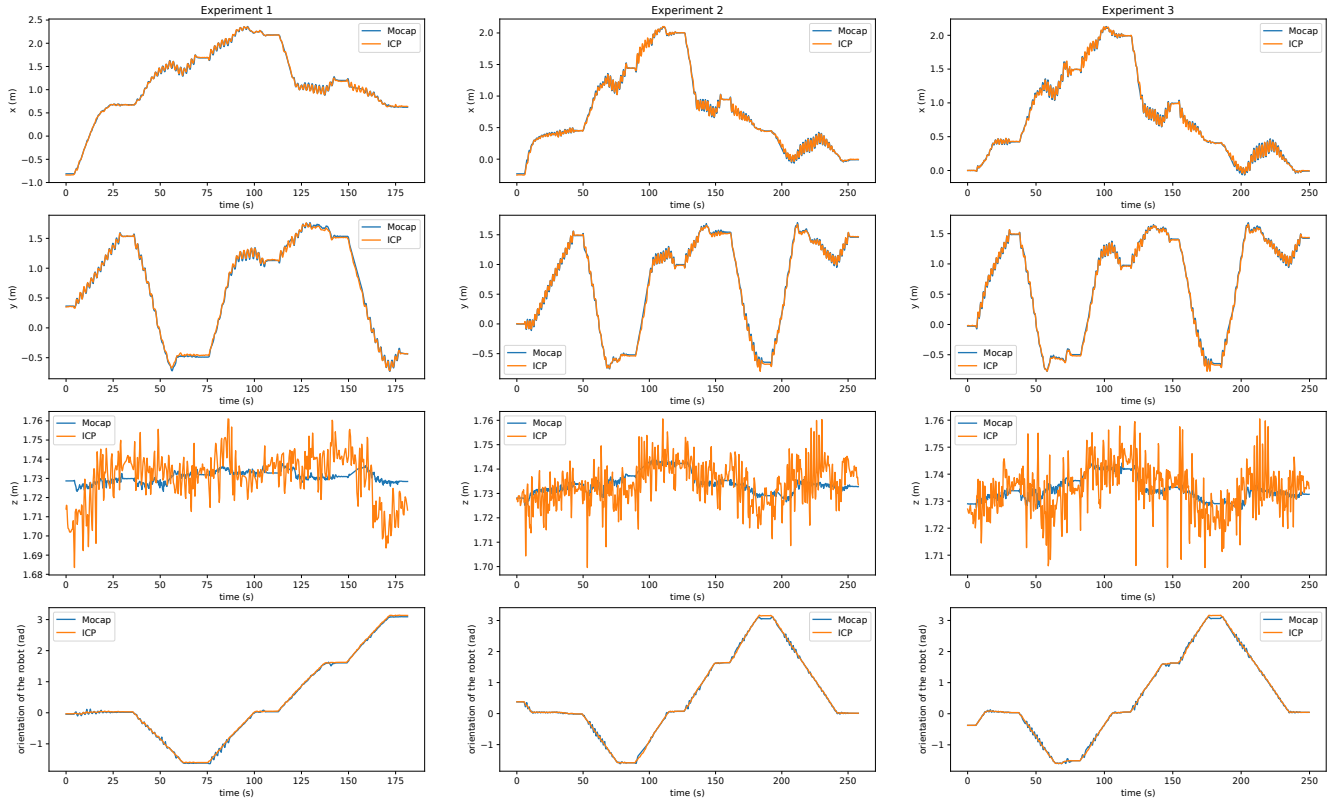


Fig. 10. Position (x on the first row, y on the second row, z on the third row) and orientation (θ on the last row) of the robot measured by the MoCap (in blue) and by the ICP localization system with the removed delays and structural offsets (in orange) during the 3 experiments

steps registering read point clouds to a map point cloud. The system was successfully deployed on the robot and tested in a walking situation. The results were compared to a MoCap system considered here as the ground truth.

The ICP-based localization system shows an average error of 0.016 m, 0.021 m and 0.0067 m along the x , y and z axis and of 0.024 rad around the z axis with respect to the motion capture system taken as the ground truth. Those results show that the ICP is accurate enough to target, in future work, more challenging localization tasks such as precisely placing the robot in front of an object to interact with it.

REFERENCES

- [1] O. Stasse, T. Flayols, R. Budhiraja, K. Giraud-Esclasse, J. Carpentier, J. Mirabel, A. Del Prete, P. Souères, N. Mansard, F. Lamiraux, J.-P. Laumond, L. Marchionni, H. Tome, and F. Ferro, "TALOS: A new humanoid research platform targeted for industrial applications," in *IEEE International Conference on Humanoid Robotics, ICHR*, 2017.
- [2] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "Monoslam: Real-time single camera slam," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 1052–1067, 2007.
- [3] M. Fallon, S. Kuindersma, S. Karumanchi, M. Antone, T. Schneider, H. Dai, C. P. D'Arpino, R. Deits, M. DiCicco, D. Fourie, T. Koolen, P. Marion, M. Posa, A. Valenzuela, K.-T. Yu, J. Shah, K. Iagnemma, R. Tedrake, and S. Teller, "An Architecture for Online Affordance-based Perception and Whole-body Planning," *Journal of Field Robotics*, vol. 32, no. 2, pp. 229–254, 2015.
- [4] N. G. Tsagarakis, D. G. Caldwell, F. Negrello, W. Choi, L. Baccelliere, V. Loc, J. Noorden, L. Muratore, A. Margan, A. Cardellino, L. Natale, E. Mingo Hoffman, H. Dallali, N. Kashiri, J. Malzahn, J. Lee, P. Kryczka, D. Kanoulas, M. Garabini, M. Catalano, M. Ferrati, V. Varricchio, L. Pallottino, C. Pavan, A. Bicchi, A. Settini, A. Rocchi, and A. Ajoudani, "Walk-man: A high-performance humanoid platform for realistic environments," *Journal of Field Robotics*, vol. 34, no. 7, pp. 1225–1259, 2017.
- [5] A. Hornung, K. M. Wurm, and M. Bennewitz, "Humanoid robot localization in complex indoor environments," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2010, pp. 1690–1695.
- [6] F. Dellaert, A. Hornung, and M. Bennewitz, "Improved proposals for highly accurate localization using range and vision data," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012, pp. 229–254.
- [7] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, "Monte carlo localization for mobile robots," in *IEEE International Conference on Robotics and Automation (ICRA)*, vol. 2, 1999, pp. 1322–1328.
- [8] F. Dellaert and M. Kaess, "Factor graphs for robot perception," *Foundations and Trends® in Robotics*, vol. 6, no. 1-2, 2017.
- [9] F. Pomerleau, F. Colas, R. Siegwart, and S. Magnenat, "Comparing icp variants on real-world data sets," *Autonomous Robots*, no. 3, p. 133–148, Feb. 2013.
- [10] S. Nobili, R. Scona, M. Caravagna, and M. Fallon, "Overlap-based icp tuning for robust localization of a humanoid robot," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 4721–4728.
- [11] A. Tanguy, D. D. Simone, A. I. Comport, G. Oriolo, and A. Kheddar, "Closed-loop MPC with dense visual SLAM - stability through reactive stepping," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2019, pp. 1397–1403.
- [12] P. J. Besl and N. D. McKay, "A method for registration of 3-d shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, 1992.
- [13] M. Ramezani, G. Tinchev, E. Iuganov, and M. Fallon, "Online lidar-slam for legged robots with robust registration and deep-learned loop closure," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2020.