2020 IEEE International Conference on
Autonomous Robot Systems and Competitions (ICARSC)
April 15-16, Ponta Delgada, Portugal

Enhanced Performance Real-Time Industrial Robot Programming by Demonstration using Stereoscopic Vision and an IMU sensor

Vítor H. Pinto [*†], António Amorim [†], Luís Rocha [†], António P. Moreira [*†],
[*] Faculty of Engineering of University of Porto (FEUP), Porto, Portugal
Email: {vitorpinto, amoreira}@fe.up.pt
[†] Centre for Robotics in Industry and Intelligent Systems - INESCTEC, Porto, Portugal
Email: {antonio.m.amorim, luis.f.rocha}@inesctec.pt

***Abstract*—Nowadays, industrial robots are still commonly programmed using essentially off-line tools, such as is the case of structured languages or simulated environments. This is a very time-consuming process, which necessarily requires the presence of an experienced programmer with technical knowledge of the set-up to be used, as well as a concept and a complete definition of the details associated with the operations. Moreover, considering some industrial applications such as coating, painting, and polishing, which commonly require the presence of highly skilled shop floor operators, the translation of this human craftsmanship into robot language using the available programming tools is still a very difficult task. In this regard, this paper presents a programming by demonstration solution, that allows a skilled shop floor operator to directly teach the industrial robot. The proposed system is based on the 6D Mimic innovative solution, endowed with an IMU sensor as to enable the system to tolerate temporary occlusions of the 6D Marker. Results show that, in the event of an occlusion, a reliable and highly accurate pose estimation is achieved using the IMU data. Furthermore, the selected IMU was a low-cost model, to not severely increase the 6D Mimic cost, despite lowering the quality of the readings. Even in these conditions, the developed algorithm was able to produce high-quality estimations during short time occlusions.**

***Index Terms*—Programming by Demonstration, Agile Robotic Systems, Industry 4.0, IMU**

I. Introduction

The current industrial environment is characterized by rapid changes in the market requirements and customers' demands, leading to products being increasingly customized and with reduced life-cycles [1]. The combination of these factors makes it imperative for the manufacturing companies to be able to constantly and rapidly adapt the manufacturing operations and their inherent complexity, forcing them to make a technological leap and to invest in innovation, knowledge, and integration of new technologies to stay competitive in the globalized market [2].

In this regard, industrial robots can be a key technology to help manufacturing companies overcome these challenges. The proof of this is their positioning in the context of industry 4.0, where the development of production systems endowed by robotic systems that can complete tasks autonomously while at the same time guaranteeing security, flexibility, versatility, and collaboration, occupy a prominent position taking into account technological advantage [3]. Likewise, in the era of digitization, where the interconnection of all the relevant components for production is upheld as a way of providing the much desired increase in flexibility and throughput of the production system, robotic systems are also seen as smart objects that allow the production management system to sense and act in real-time.

However, and despite the effort of both the scientific and industrial communities to provide proper interfaces to easily and rapidly perform the programming of these robotic systems, they still present limitations when more complex trajectories, like for painting, polishing or flaming, need to be programmed. Many times, in these industrial applications, human craftsmanship needs to be precisely imitated and the available commercial robot programming solutions, based on highly advanced teach pendants and simulation software, fall short to deliver, affecting the large-scale adoption of industrial robots in many industrial sectors [4], [5].

This paper aims to answer these challenges by providing an advanced robot programming by demonstration system capable of capturing, in a fast and intuitive way, the know-how accumulated by specialized technicians and optimized during several years of experience. The system presented is based on previous research work, endowed with a complementary component based on an IMU sensor, to deal with the 6D Marker occlusions during the teaching of complex trajectories. This paper showcases these scientific developments on a specific industrial painting application, however, the results can similarly apply to other use case scenarios in different industrial sectors.

Bearing these ideas in mind, the remainder of the paper is organized as follows: after this introductory section, Section II presents the state-of-art which focuses on intuitive interfaces for industrial robot programming. Section III presents the developed robot programming by demonstration system that is able to tolerate short periods of occlusion of the 6D Marker. Section IV presents the results and respective discussion. Finally, in Section V both the conclusions and future work are presented.

978-1-7281-7078-7/20/$31.00 ©2020 IEEE

2020 IEEE International Conference on
Autonomous Robot Systems and Competitions (ICARSC)
April 15-16, Ponta Delgada, Portugal

II. Related Work

Considering industrial robots themselves, one of their most limiting characteristics, and that prevents their widespread application into different industrial sectors, is their programming procedure that still today requires long periods of time and that is only accessible to highly specialized users, despite the effort on the contrary. Over the last decade, and in the scientific community, there has been some effort to overcome this limitation, namely through semi-automatic programming based on CAD models [6], speech and gesture recognition [7], [8], and tracking of the human hand [9]. Regarding programming based CAD models, this is, in fact, a major stream with extensive examples of application in the industry. Here, the user interacts with a simulation software in order to define the robot paths. Although their undoubted advantages, these software's are usually somewhat complex and not accessible to operators without any robotics expertise. Moreover, they do not allow the operator to demonstrate the task, but rather provides a simplified way to define robot paths.

Considering the speech and gesture recognition approach, this is another extensively covered method. Nevertheless, this is in fact more an intuitive communication tool with the industrial robot, rather than a programming tool, since in such solution the robot must be pre-programmed with the primitives that correspond to each human gesture. Finally, in the tracking of the human hand approach, the human hand movement is normally captured by tracking a data glove endowed with tactile sensors for grasp recognition. Here, the movement of the human hand can be acquired but the apparatus lacks the robustness to be applied at industrial environments.

Specifically, in the tracking based methodologies, a key part in their integration is the ability to reliably track a moving body, typically a human operator's hand. The aforementioned type of sensors (vision-based and wearable) achieve promising results [10] [11]. However, errors are intrinsically present in the form of occlusions, i.e. obstruction of the vision source. In these instants, there is no motion tracking taking place. Some new approaches aim to compensate these potential time gaps, where the main system in charge of the motion tracking fails, using deep learning [12] and robot reinforcement learning [13]. Nonetheless, its applications are still limited to the recognition of a relatively small array of gestures. Our proposed fail-safe method is based on the introduction of an IMU, which can introduce important redundancy into the system, allowing the pose estimation of the moving body even when the primary system fails during short intervals of time.

III. Robot Programming by Demonstration System

The programming by demonstration system presented in this article is based on previous research work presented by Marcos Ferreira et al. [14], which is called the 6D Mimic system. This solution ultimately delivers the possibility of transferring human expertise to industrial robots. It proposes a new mean for human motion tracking based on an innovative 6D Marker together with a collection of routines that allows a real-time interface with the robot. With this technology, the goal of allowing a human to program an industrial manipulator with a complete abstraction of programming concepts has been achieved. In addition, the process happens while the operator is kept in his zone of comfort: doing the everyday tasks and putting his highly specialized skills into work. This 6D Marker needs to be attached to the operator's tool, allowing the expert to teach by demonstration the operation to the industrial robot, that is then tracked by a stereoscopic vision system, patented by the University of Porto and licensed to FLUPOL company [15]. Both components are managed directly by the 6D Mimic software package. However, and despite the innovative solution presented by the authors, many times, coating, painting, and similar operations are executed over complex geometry, where exists high probability of the 6D Marker to be occluded, and in such cases making it impossible the usage of this programming by demonstration system to completely capture the human motion. To overcome this system limitation, this article proposes the addition of a new tracking system based on IMU that compensates for temporary occlusion of the 6D Marker during the programming by demonstration procedure. For this purpose, it was necessary to develop new hardware and software modules for the control and for the integration of the various subsystems.

A. Laboratory Set-up

The laboratory set-up is depicted in Fig. 1 and is constituted by the following main components:

- **an ABB IRB 2600 industrial manipulator** and its controller. This robot is equipped with an automatic Spray Gun, an AirPro from GRACO, which is actuated by solenoid valves and the coating paint is injected through a pressurized vase. These two valves are actuated by the ABB controller directly.
- **automatic rotating table**, with an incorporated PLC module. It allows the operator to always be painting on the same position, and just use a pedal for the table to rotate the part. The rotating mechanism is placed in the center of the table and is actuated through an induction motor. To ensure that the table rotates exactly 90º, a magnetic sensor and 4 plates were strategically placed under the top. This is controlled by an adjustable frequency AC drive connected to a Programmable Logic Controller (PLC). The PLC is in charge of dealing with the signals coming from the pedal, the manual coating spray gun on/off switch, the magnetic sensor and the signals from the ABB controller. In addition, it sends signals regarding the switch and the pedal to the SincroBox, which will be further introduced, to inform when the operator acts each one of these. The PLC is an M-DUINO 42+ PLC from Industrial Shields.
- **the 6D Mimic system, endowed by an IMU**. The 6D Mimic is mainly constituted by (1) an industrial PC (NEUOSYS Nuvo-3005TB) where the machine vision processing software is installed, (2) the Sincrobox, whose main purpose is to be the central component of the system making the communication bridge and synchronization

2020 IEEE International Conference on
Autonomous Robot Systems and Competitions (ICARSC)
April 15-16, Ponta Delgada, Portugal

between all the signals (e.g. the pedal and manual spray gun switch signals) from the different hardware components of the entire solution, (3) the Stereoscopic vision system, constituted by two industrial cameras (Mako G125C PoE), mounted on a fixed rail, and that are capable of seeing the working field of the robotic arm, and finally (4) the 6D Marker, that is attached to the manual coating spray gun, as well as a simple switch to check whether the operator is pulling the trigger or not. Also, inside the 6D Marker, support for the IMU and the rest of the electronics associated with it was added, which can be seen in Fig. 2. This piece supports an Arduino Nano, a TTL to RS232 Converter, and an IMU with 10 Degrees of Freedom from Waveshare (MPU9255) set to its highest possible bandwidth.

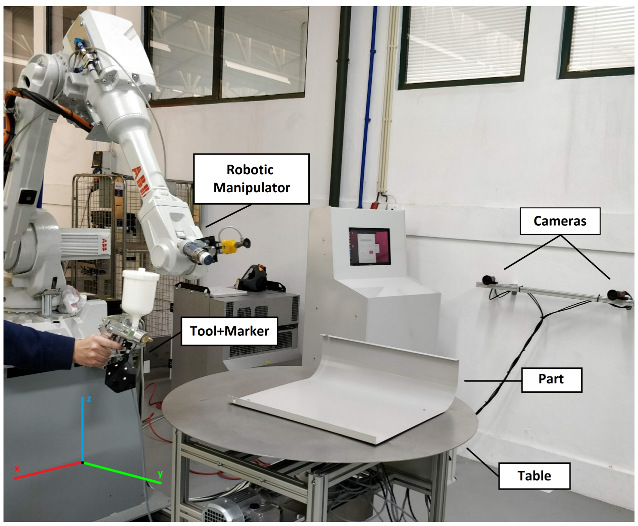

Fig. 1. System Hardware Architecture and its interconnections

The final solution's hardware architecture is depicted in Fig. 3.

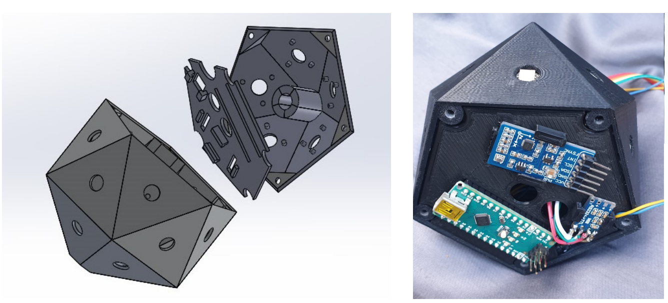

Fig. 2. CAD Representation and Real 6D Marker with Support

B. Software Architecture

The software architecture can be decomposed in the following modules:

- the 6Dof tool pose tracking software, that is responsible for tracking the 6D Marker, resorting to the stereoscopic vision-based system, and compute its pose;

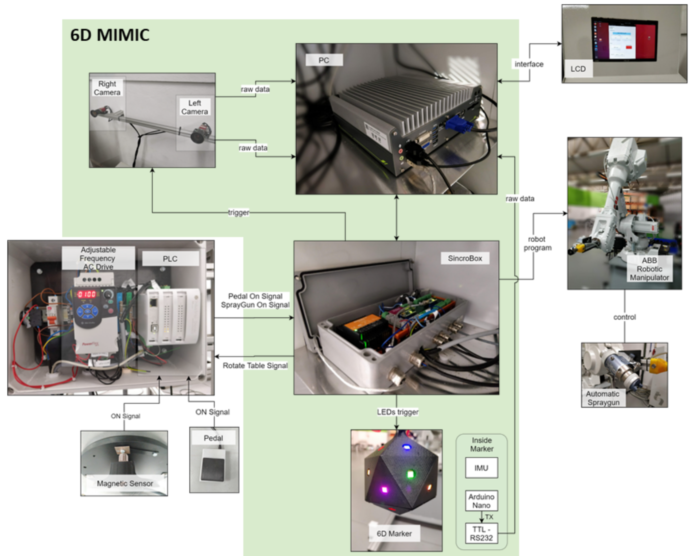

Fig. 3. System Hardware Architecture and its interconnections

- the IMU data acquisition, in charge of sending and receiving information from the IMU sub-system and collect all the data in the appropriate memory buffer;
- the data analysis and processing , in which the results of the 6D MIMIC are analyzed (in order to identify occlusions and compensating with the IMU data in case they occur), and processed into a 3D point time series that characterize the final trajectory.
- the post-processor software, responsible to translate the complete set of 3D points into robot programming languages. Currently, the following robot manufacturers are supported: ABB, YASKAWA and KUKA.

In order to correctly utilize the data provided by the IMU, a transformation between its coordinate frame and the 6D Marker reference frame was required. The model and mounting of the IMU in the tool can provide several physical constraints, leading to non-intuitive transformations. With that in mind, it was possible to compute the homogeneous transformation between these frames, without any previous physical information, using the Kabsch Algorithm [16].

By collecting sets of data points from both sources in different, approximately orthogonal, positions along all 3 axis of the world frame with varying orientations, the aforementioned algorithm computes the optimal rotation matrix between the 2 sets of paired points. For the present application, this procedure only needs to be executed once since the IMU frame was fixed in relation to the 6D Marker frame (Figure 2).

The programming by demonstration procedure begins with an operator performing the motion he wishes to replicate with the tool. While this is taking place, a stereoscopic vision system is tracking the motion of the operator manual tool and recording it for post-processing, which corresponds to the first module. This system provides the absolute pose (*i.e.* position and orientation) of a luminous 6D marker (Figure 2)

2020 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)
April 15-16, Ponta Delgada, Portugal

for any recorded instant, assuming it is correctly detected by the cameras. Currently, this system is used in painting and surface covering applications trough a spray-paint gun that can handle a position error of up to 5 millimeters and an angle error of around 1º. More information about this motion tracking system can be found here [14].

During this process, the IMU is also providing accelerometer and gyroscope data. This data is transmitted via I2C at a frequency of 200 Hz and recorded. This represents the second module.

When the operator finishes painting the part, the third module is initiated, which means the data is analyzed and processed. First, the resulting motion tracking is proofed in order to detect potential faults that may have occurred. If, at any time, a fault is detected, the corresponding IMU readings are used in order to estimate the pose of the tool. The accelerometer data is used to obtain the position trough double integration and the orientation change between two time instants is estimated using the gyroscope. Due to magnetic disturbances in our workspace, it is not viable to utilize the magnetometer data, as well as the built-in position estimation that relies on it. In order to estimate the position of the tool, first we calculate the expected acceleration $a[i]$, based on the last N non-occluded position measures, using Equations 1 and 2.

$$y(t) - y_0 = v_0 \cdot t + a_0 \cdot \frac{t^2}{2} + a_a \cdot \frac{t^3}{6} \tag{1}$$

$$a(t) = a_0 + a_a \cdot (t - t_0) \tag{2}$$

Where $y(t)$ is the measured position and $a(t)$ is the acceleration at instant t, y_0, v_0, a_0 and a_a are the position, velocity, acceleration and jerk on the $t - N$ position reading, respectively. These equations represent continuous motion, which we discretize in order to use the last N readings to compute the values of v_0, a_0 and a_a via Least Squares Estimation.

By comparing the expected acceleration $a[i]$ to the acceleration read by the IMU $a_{read}[i]$, we accurately calculate the IMU offset for instant $[i]$ (Equation 3). This value also compensates gravity's pull g, which is a constant factor.

$$\mathit{offset} = a_{read}[i] - a[i] \tag{3}$$

Where a_{read} is the value of the acceleration read from the IMU for instant i.

After calculating the offset value, the measured acceleration from the IMU is rotated to the global frame and adjusted according to a gain K (Equation 4). This value was calculated offline, only once, from the difference between the velocity obtained by differentiating the position readings and integrating the acceleration readings. However, trough experimentation, we verified that, if the absolute value of the expected acceleration was too high, it was best to skip the offset calculation for the instant we are analysing as the calculated value was erroneous. When the previous condition is met, we simply assume that the offset value for the instant is the same as for the previous one.

$$a_{IMU} = K \cdot R_{body}^{global} \cdot R_{IMU}^{body} \cdot (a_{read} - \mathit{offset}) \tag{4}$$

Where the generic R_x^y is the rotation matrix from the x frame to the y frame and a_{IMU} is the measured acceleration given by the IMU in the global frame.

The position of the moving body is then estimated, assuming a constant acceleration model between two successive readings, by using the measured and calculated values:

$$Yest[i] = Yest[i_{-1}] + v_0 \cdot (i - i_{-1}) + a_{IMU} \cdot \frac{(i - i_{-1})^2}{2} \tag{5}$$

Where $Yest[i]$ is the estimated position at instant i. Several factors determine the accuracy of this estimation such as the duration of the fault, the quality of the IMU (*i.e.* cost) and the complexity of the recorded movements. However, taking advantage of the fact that this final processing takes place *offline* and the stereoscopic vision system provides an absolute pose, the IMU based estimation can be corrected with the divergence of the last estimated and the first absolute pose readings after the fault. Therefore, we can update the estimated position based on their previously calculated values and a correction factor according to equation 6.

$$pos_e[i-k] = pos_e[i-k] + \delta \cdot \frac{Np - k}{Np} \tag{6}$$

Where δ is the correction factor based on the mentioned divergence, pos_e is the estimated pose from the IMU data, Np is the number of estimated points during the occlusion and $k = 1, 2, \ldots, Np$. This final correction allows for a robust and very high accuracy estimation. The recorded tool path is then converted into a 3D point time series, and stored locally. Finally, this list of 3D points is then translated into robotic programming language, using the developed post-processor software module. It currently supports three different robot manufacturer programming languages, but can easily be extended in case of need.

IV. Results

In this section the demonstration and testing scenario will be characterized. Furthermore, the performance of the proposed teaching by demonstration system against this scenario will also be presented and discussed.

A. Demonstration Scenario Description

As previously stated, the data provided by the IMU is constantly being recorded. However, it is only used in the event of a fault in the main motion tracking system. A fault in a vision based system is typically a byproduct of an occlusion (Figure 4). In this sense, the motions to be analysed are a linear motion along one axis and a two-dimensional circular motion, both with a time interval where an occlusion of the 6D Marker occurs. These motions were chosen in order to represent the possibility of attempting to recreate the painting of a complex part whose volume might eventually hide the tool from the Field of View (FOV) of the cameras.

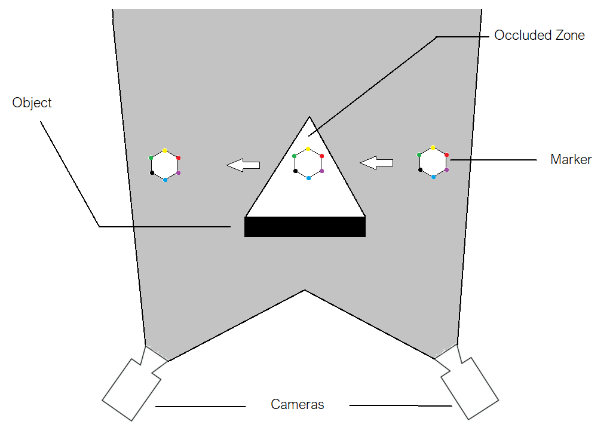

Fig. 4. Example of camera occlusion

B. Results Analysis and Discussion

As for the aforementioned first motion, it is possible to observe the unprocessed position estimation on a single axis, Y, in Figure 5. The position given by the main motion tracking system consist of the blue dots and the red dots consist of the IMU based position estimation. A discrepancy in the final estimated point and the first measured point after the occlusion is visible and represents an error of approximately 20 millimeters.

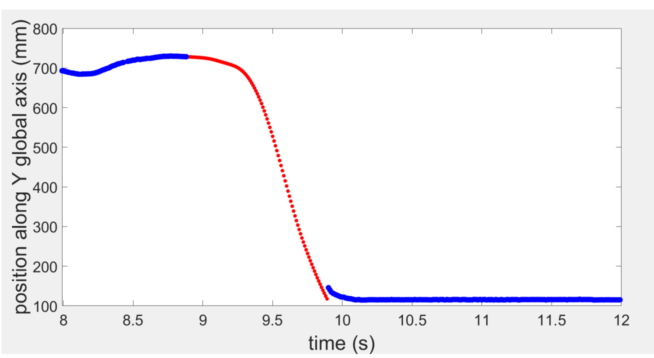

Fig. 5. First motion - IMU data position estimation (red) and 6D Mimic given position (blue)

The estimated set of points is then corrected according to the procedure specified in Section III-B and can be observed in Figure 6. The previous error has been nullified and scaled over the remaining estimated points. Thanks to this correction, the estimation's accuracy becomes substantially higher since the observable error is close to null and the estimated trajectory is maintained. The impact of this process will be further analysed in the following test.

The other motion that was chosen to be analyzed was slightly more complex since it is described along 2 axis. However, the concept is the same: at some point during the movement of the tool, the 6D Marker was occluded. A possible scenario where this could occur is, for example, attempting to paint the inside of a box. The unprocessed results can be observed in Figure 7. During these tests, we used the Optitrack system to establish a ground truth. This system tracks the motion described by a set of reflective spheres, equipped in the tool, using infrared cameras with precision of less than 1 mm. The position given by the Optitrack system is represented by the blue dots. Aftwerwards, we selected a time interval where we ”simulated” occlusions in order to test the estimation. As expected, since this movement is more complex, there is a much more predominant error in the final point of estimation, approximately 100 millimeters, corresponding the difference between the green and blue points.

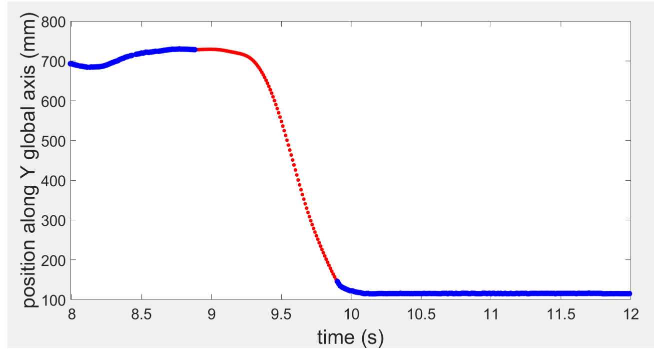

Fig. 6. First motion - Corrected estimation with offline processing and 6D Mimic given position (blue)

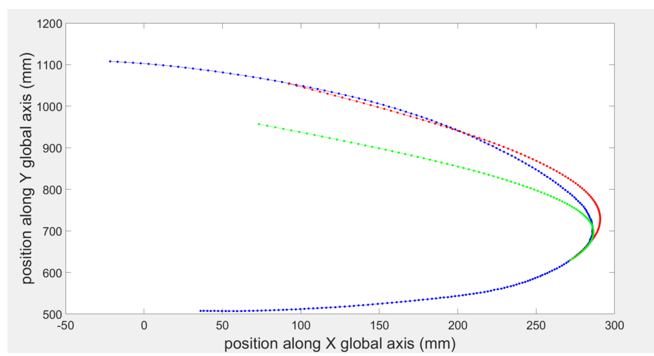

Fig. 7. Second motion - non-corrected IMU data position estimation (green), corrected IMU data position estimation (red) and real position (blue)

However, despite the errors, the direction of motion during the estimation process is more or less correct. Therefore, the best possible estimation stems from minimizing the error while maintaining the motion. The results of the correction applied to the estimation during the offline processing phase, visible as the red dots in Figure 7, showcase our attempt to reach that goal.

In this second motion, the accuracy of the proposed estimation process is clear. The IMU data produces a somewhat reliable pose estimation, however, the trajectory it describes is the main takeaway. By taking advantage of the offline processing to apply a correction to the estimation it becomes possible to achieve a highly accurate motion estimation during the occlusion period. By comparing the red and blue dots in Figure 7, we can confirm the previous statement, observing an error of 17 millimeters in the worst case.

In both tests, the occlusion had a duration of 1 second. This time interval was selected based on the accuracy of the estimated trajectory: since the IMU used was low-cost, the readings are eventually overflowed with errors. However, the optimal interval aims to be the maximum time span where a very accurate trajectory estimation is still obtained. The presented results demonstrate that the 1 second interval is a significant amount of time where large shifts in trajectory and position can take place, while maintaining the high quality of the estimation.

V. Conclusions and Future Work

This paper presented a programming by demonstration system, capable of dealing with short-period occlusions of the operators tool. The proposed system is based on a previous research work, the 6D MIMIC system, that was endowed with an IMU. The proposed solution was tested in a realistic use case scenario, and the obtained results allows to conclude that with the introduction of the IMU, the system's robustness increased significantly since it provided a way to deal with the 6D Marker occlusion. Despite using a low-cost IMU, with the proposed system it was possible to achieve a robust and highly accurate pose estimation of the operators tool by fusing both sources of data (IMU and 6D Mimic), capable of maintaining said accuracy during occlusions of up to 1 second. For future work, the aim is to increase the duration in which it still possible to rely solely on the IMU. This can be achieved by adding some form of post-processing to the IMU readings and/or using a higher quality IMU, albeit with higher cost. Since we are only utilizing the IMU during occlusions, we also plan to pursue the implementation of a Kalman Filter which permanently fuses the information from both sensors. However, for our use case, such a method was not necessary since the 6D Mimic system already provided a sufficiently accurate estimation which we don't believe can be improved using such a low-cost IMU. Moreover, the offline processing section can be taken advantage of in different ways, namely performing the correction trough other methods, which we will consider in the future.

Acknowledgment

This research work is supported by the Portuguese Foundation for Science and Technology (FCT) project COBOTIS (PTDC/EME-EME/32595/2017). This work is also co-financed by the ERDF – European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation - COMPETE 2020 and the Lisboa2020 under the PORTUGAL 2020 Partnership Agreement, and through the Portuguese National Innovation Agency (ANI) as a part of project PRODUTECH SIF: POCI-01-0247-FEDER-024541.

References

[1] K. Sherringham and B. Unhelkar, *Knowledge Workers and Rapid Changes in Technology*. Singapore: Springer Singapore, 2020, pp. 1–48.

[2] W. Banaś and A. Sekala, "Concepts of flexible production line, on the example of robotic cell," in *Modern Technologies in Industrial Engineering II*, ser. Advanced Materials Research, vol. 1036. Trans Tech Publications Ltd, 11 2014, pp. 749–754.

[3] B. Chen, J. Wan, L. Shu, P. Li, M. Mukherjee, and B. Yin, "Smart factory of industry 4.0: Key technologies, application case, and challenges," *IEEE Access*, vol. 6, pp. 6505–6519, 2018.

[4] M. Ghahramani, A. Vakanski, and F. Janabi-Sharifi, "6d object pose estimation for robot programming by demonstration," in *Progress in Optomechatronic Technologies*. Springer, 2019, pp. 93–101.

[5] I. K. Singgih, O. Yu, B.-I. Kim, J. Koo, and S. Lee, "Production scheduling problem in a factory of automobile component primer painting," *Journal of Intelligent Manufacturing*, pp. 1–14, 2020.

[6] H. Chen and N. Xi, "Automated tool trajectory planning of industrial robots for painting composite surfaces," *The International Journal of Advanced Manufacturing Technology*, vol. 35, no. 7, pp. 680–696, Jan 2008. [Online]. Available: https://doi.org/10.1007/s00170-006-0746-5

[7] S. Waldherr, R. Romero, and S. Thrun, "A gesture based interface for human-robot interaction," *Autonomous Robots*, vol. 9, no. 2, pp. 151–173, Sep 2000. [Online]. Available: https://doi.org/10.1023/A:1008918401478

[8] J. N. Pires, "Robot-by-voice: Experiments on commanding an industrial robot using the human voice," *Industrial Robot: An International Journal*, vol. 32, pp. 505–511, 12 2004.

[9] R. Zollner, O. Rogalla, R. Dillmann, and M. Zollner, "Understanding users intention: programming fine manipulation tasks by demonstration," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, Sep. 2002, pp. 1114–1119 vol.2.

[10] F. Ordóñez and D. Roggen, "Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition," *Sensors*, vol. 16, no. 1, p. 115, 1 2016. [Online]. Available: http://www.mdpi.com/1424-8220/16/1/115

[11] M. T. Wolf, C. Assad, M. T. Vernacchia, J. Fromm, and H. L. Jethani, "Gesture-based robot control with variable autonomy from the JPL BioSleeve," in *2013 IEEE International Conference on Robotics and Automation*. IEEE, 5 2013, pp. 1160–1165. [Online]. Available: http://ieeexplore.ieee.org/document/6630718/

[12] D. Wu, L. Pigou, P.-J. Kindermans, N. D.-H. Le, L. Shao, J. Dambre, and J.-M. Odobez, "Deep Dynamic Neural Networks for Multimodal Gesture Segmentation and Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 8, pp. 1583–1597, 8 2016. [Online]. Available: https://ieeexplore.ieee.org/document/7423804/

[13] F. Chao, Z. Wang, C. Shang, Q. Meng, M. Jiang, C. Zhou, and Q. Shen, "A developmental approach to robotic pointing via human–robot interaction," *Information Sciences*, vol. 283, pp. 288–303, 11 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0020025514004010

[14] M. Ferreira, P. Costa, L. Rocha, and A. P. Moreira, "Stereo-based real-time 6-dof work tool tracking for robot programing by demonstration," *The International Journal of Advanced Manufacturing Technology*, vol. 85, no. 1, pp. 57–69, Jul 2016. [Online]. Available: https://doi.org/10.1007/s00170-014-6026-x

[15] C. G. D. C. Paulo, L. M. P. Sergio, and G. M. M. A. Paulo, "3d object motion tracking and locating system by means of synchronised light emitters with a stereoscopic vision system," Patent Application WO 2010/046 759 A2. [Online]. Available: https://lens.org/099-937-885-906-015

[16] W. Kabsch, "A solution for the best rotation to relate two sets of vectors," *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography*, vol. 32, no. 5, pp. 922–923, 1976.