

## Research Article

# Design of a String Encoder-and-IMU-Based 6D Pose Measurement System for a Teaching Tool and Its Application in Teleoperation of a Robot Manipulator

Ekrem Yavuz <sup>1</sup>, Yavuz Şenol <sup>2</sup>, Merih Özçelik <sup>3</sup>, and Hüseyin Aydın <sup>3</sup>

<sup>1</sup>Department of Mechatronics Engineering, The Graduate School of Natural and Applied Sciences, Dokuz Eylül University, Izmir 35390, Turkey

<sup>2</sup>Department of Electrical and Electronics Engineering, Engineering Faculty, Dokuz Eylül University, Izmir 35390, Turkey

<sup>3</sup>Jeanologia Teknoloji A.Ş., Izmir 35620, Turkey

Correspondence should be addressed to Ekrem Yavuz; [ekrem.yavuz@deu.edu.tr](mailto:ekrem.yavuz@deu.edu.tr)

Received 10 December 2020; Revised 23 April 2021; Accepted 11 May 2021; Published 16 June 2021

Academic Editor: Giuseppe Maruccio

Copyright © 2021 Ekrem Yavuz et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Potassium permanganate is mostly used for lightening denim in the textile industry. It is an inorganic chemical compound and an oxidizing agent. PP is sprayed to denim with a spray gun as microparticles. During the spraying process, these microparticles can be inhaled by workers causing lung diseases in the long term. PP spray is one of the most dangerous practices in the textile industry for the health and safety of workers. Robots can carry out PP spray instead of humans to protect workers' health. Also, robots offer less labor cost, no labor mistake, fast production, and standardization in every product. The use of robots requires the teaching of tasks to robots. For this teaching procedure, it is necessary to record the teaching tool's movements with high precision and accuracy. In this study, it is aimed at measuring the position and orientation, i.e. the pose, of the teaching tool in six dimensions using an inertial measurement unit (IMU) and six string encoders. In addition to this, the measured pose information is transferred to a robot manipulator for tracking the recorded trajectory. In that way, the operator's movements can be copied and then used by robots to regenerate the best product while protecting the health of workers. The pose measurement system was designed and experimentally tested. The obtained results show that the system satisfies accuracy and precision expectations for the target application.

## 1. Introduction

Potassium permanganate ( $\text{KMnO}_4$ ) is an odorless, dark purple salt and is an inorganic chemical compound. It is easily dissolvable in water. It is mostly used in air pollution treatment, water treatment, waste treatment, fruit preservation, fire services, medical purposes, and metal plating industry [1]. In the textile industry, a liquid solution prepared with water and PP (with PP being potassium permanganate, about 2-5% concentration of PP depending upon the desired results and fabric types) is applied with a spray gun to the desired area of the denim to obtain a brighter and abraded look than the other parts of the denim. Besides being an effective procedure, it poses a risk for human and environmental health. It contains manganese which is a chemical element. Also, man-

ganese is a heavy metal and is nonbiodegradable. During the spraying process, chemical microparticles can be inhaled by workers and can cause lung diseases in the long term. Therefore, it is necessary to minimize the contact of PP with workers. It can be an effective solution if the spraying process is performed by a robot. A robot, which is trained by an employee who is an expert in the PP spray, can do the job continuously and without any error. And also, robots offer less labor cost, no labor mistake, fast production, and standardization in every product. Although studies are being carried out to find alternatives to potassium permanganate, the major benefit of this proposed method is fast and standardized production.

Robots have become indispensable tools of today's world due to the many benefits they provide. In the literature,

different types of robots have been introduced specifically for their intended use and application types [2, 3]. The use of robots requires that tasks be taught to robots, and this requires programming them. There are a few different methods for robot programming. The most common practice is to program a robot with a *teach pendant* [4]. A teach pendant is a human-machine interface (HMI) device which is connected to the robot by a cable for on-site monitoring of the state of the robot. While programming a robot by a teach pendant, the operator must manually define each parameter such as target points, commands, functions, speeds, and motion modes. This method is easy and fast for jobs such as welding and screwing, where only a few specific points are taught to robots. However, for jobs that include continuous paths, such as spraying, painting, and flame cutting, this method is insufficient. In this work, we focused on the *Programming by Demonstration* (PbD) method, whose aim is to enable humans to program the robot by guiding it without any robot programming knowledge. The demonstration part of this method can be in various ways such as kinesthetic or teleoperation [5]. In the kinesthetic demonstration, the robot needs to be physically guided by a human teacher, while teleoperation teaching does not require physical guidance. The teacher and the robot may be away from each other, and the teacher does not need to see the robot. Instead of physical guidance, a teleoperation device called the teaching tool is used to establish the connection between the robot and the human teacher. The robot end-effector imitates human arm movements (teaching tool's movements) by using teleoperation. The designed system's purpose is to have the robot imitate the performed movements by a human teacher regardless of the user's knowledge of robotics or programming. The movements of the teaching tool must be recorded continuously by using a tracking system to successfully achieve the imitation process. An accurate and precise recording process can be performed with a teaching tool and a tracking system suitably designed for the study.

Various types of trajectory tracking systems have been introduced in the literature, and they are used in many different fields of application [6, 7]. Tracking systems can be realized with different systems, which are microelectromechanical (MEMS), acoustic, electromagnetic, optical, radiofrequency, and mechanical systems. Each system has cons and pros, and therefore, the selection of the tracking system is made based on the application requirements. For example, in places where there are strong magnetic field sources, the use of electromagnetic-based tracking systems is not convenient [8]. Vision-based tracking systems provide high accuracy, but they have some drawbacks because of the external factors, which are the intensity of light, occlusion, or reflection [9]. In terms of accuracy, the laser tracking systems are the best, but they have a lengthy calibration procedure, and they are expensive and highly dependent on the external effects [10].

A more suitable method utilizes string encoder-based tracking systems. They give noise-free outputs and provide high accuracy. However, these systems have minor structural defects, such as string length uncertainties, dimensional deviations related to the production tolerances, or string loose-

ness [11]. The higher the number of strings, the higher the force applied to the movable object, and this creates difficulty for moving the object at high speeds [12, 13].

Another tracking device is the MEMS-based inertial measurement unit (IMU) which consists of a triaxial accelerometer and a triaxial gyroscope. Some IMUs are also accompanied by a triaxial magnetometer to increase the accuracy of the device. Accelerometers measure linear acceleration, whereas gyroscopes measure angular velocity and magnetometers measure magnetic flux density in three mutually orthogonal directions. The pose of the IMU is determined by these sensors' readings. The IMU has several advantages, but its main drawback is that the movement of an object cannot be tracked for a long time. Essentially, this disadvantage of the IMU comes from the integration of the sensor's output data, which is necessary to calculate the position of the IMU. The increase in output errors over time is entirely due to the integration of noisy sensor data in IMU measurements [14]. IMU errors are categorized into systematic errors and random errors. Systematic errors are permanent, but it is possible to eliminate them by calibration methods. However, random errors are based on the sensors' unpredictability. They have a much more significant impact on the results and cannot be completely eliminated [15]. In addition, magnetic interferences and mechanical vibrations also affect the performance of the IMU. IMUs are used together with other sensors to address these problems and build more accurate systems.

Hybrid systems, consisting of an IMU and a position sensor, can be employed in a variety of applications to track the pose of moving objects more accurately. The IMU can be coupled with an ultrasonic position sensor [16], an optical position sensor [17], an odometer sensor [18, 19], a radiofrequency-based system [20], or a vision-based position sensor [21, 22].

There are many studies related to hybrid systems combining IMU and string encoders [14, 23, 24]. Generally, in these applications, an IMU is used to obtain the orientation, whereas string encoders are used to obtain the position of moving objects.

The string encoder position sensor, consisting of an optical encoder, a spring-loaded spool, and a stainless steel flexible cable, measures one-dimensional displacements from a reference point. While the sensor is mounted on a fixed reference point, the end of the string is connected to a moving object. The sensor generates a signal proportional to the linear extension of the string. This signal is converted into a linear measurement by means of a controller. In applications where more than one string encoder is used, the position (and even orientation) of the moving object in 3D space can be calculated using some special computational techniques. The law of cosines, trilateration, multilateration, and triangulation are some of these special techniques [25].

In this study, a sensor system that consists of an IMU and six string encoder position sensors to measure and record the 6D pose of a teaching tool which is controlled by an operator during a teaching process is proposed. The combination of IMU and string encoders provides calculation convenience, fast response, independence from environmental factors,

and relatively high accuracy. In addition, placement selection of string encoder bodies and their wire ends together with used calculation techniques, which are explained in the following sections, makes this study have better results in comparison to similar studies [14, 23, 26–28]. Moreover, yaw angle calculation by using two additional string encoders eliminates the need for the yaw angle output of IMU which has random walk and drift problems due to its nature. In conclusion, six string encoder position sensors are used to locate the 3D position of the tip of the teaching tool and to find the yaw angle of the tool (rotation around the  $z$ -axis) by using the law of cosines. One IMU is used to find the roll angle (rotation around the  $x$ -axis) and pitch angle (rotation around the  $y$ -axis) of the teaching tool. The measured pose data is then sent to the robot via the transmission control protocol/internet protocol (TCP/IP) to command the robot in order to move its tool within the same continuous path as the teaching tool.

## 2. Materials and Methods

**2.1. Overview of the Designed System.** The designed system consists of two cabinets. The first one is the *movement record cabinet*, where the teaching tool is located, and an operator performs the movement record process. The second one is the *application cabinet*, where the robot is located to execute the recorded movements. The dimensions of these cabinets are 2200 mm  $\times$  1370 mm  $\times$  1480 mm ( $h \times d \times w$ ). Figure 1 shows the technical drawings of these cabinets.

The movement record cabinet has built-in sensors, as shown in Figure 2. This structure consists of one IMU fixed on the teaching tool and six string encoders mounted on the cabinet at a certain distance and direction to each other. Four of the strings from these string encoders were fixed to a point at the front end of the teaching tool, and the other two were fixed to a point at the rear end of the teaching tool. The information received from two string encoders fixed to the front end of the teaching tool was used to calculate and measure the  $x$ ,  $y$ , and  $z$  positions of the tool. The information received from two string encoders fixed to the rear end and two string encoders fixed to the front end of the teaching tool and the pitch angle taken from the IMU are combined and used to calculate and measure the yaw angle. All of these calculations were performed based on the law of cosines.

A button is mounted to the teaching tool to spray the dye by the operator during the movement record process. Spraying starts when the button is pressed and stops when the button is released. The state of the button is recorded together with the pose information.

There is a robot manipulator inside the application cabinet. The robot was mounted on a stationary metal stand. Nozzle end-effectors were mounted to the tool end of the robot and the teaching tool.

**2.2. Sensors and Infrastructure Used.** The overall system setup includes a robot, an IMU, six string encoder sensors, two designed system controller boards, one designed system program, and one personal computer (PC).

**2.2.1. IMU.** The ADIS16480 MEMS IMU [29], developed by Analog Devices, has been chosen for this system. The ADIS16480 is an inertial measurement unit which consists of a triaxial accelerometer, a triaxial gyroscope, a triaxial magnetometer, a pressure sensor, and a sensor fusion algorithm for dynamic orientation estimation. Some important properties of the ADIS16480 are given in Table 1.

According to Table 1, the ADIS16480 has a dynamic accuracy of  $0.3^\circ$  for the roll and pitch angles and  $0.5^\circ$  for the yaw angle. As a result of the tests performed, it was seen that the ADIS16480 had a dynamic accuracy of about  $1^\circ$  for the roll and pitch angles, but there were no precise results for the yaw angle. Some dynamic movement tests on the ADIS16480 showed that the yaw angle error could reach up to  $10^\circ$  within 30 seconds. Even if the ADIS16480 is kept stationary, the error grows fast. Test results of the ADIS16480 on the dynamic movement and stationary state can be seen in Figure 3.

It can be inferred from Figure 3 that the ADIS16480 has high attitude drifts on its outputs, especially on the yaw angle output. The yaw angle output has random walk and bias stability problems. For these reasons, the roll and pitch angle outputs of the ADIS16480 are used, but the yaw angle output is not used in this system.

**2.2.2. String Encoder Position Sensor.** String encoder position sensors are used for one-dimensional displacement measurements in many industrial applications. By combining these sensors with a correct configuration, the position and even the orientation of a movable object can be measured. The best-known feature of a string encoder position sensor is its high stability and noiseless measurement capability. These features ensure the creation of a system with high accuracy.

The proposed system contains six string encoder position sensors. These sensors have two parts: a wire draw mechanism and an incremental encoder. The wire draw mechanisms of these sensors are SICK-PFG13-A1CM0544, and encoders are SICK-DFS60B-S1AC04096. The measurement range of the wire draw mechanism is 0 to 5 m with  $\leq 0.5$  mm reproducibility. The maximum operating speed is 4 m/s, and the maximum wire acceleration is  $4 \text{ m/s}^2$ . The incremental encoder produces 4096 pulses per revolution, and it has a 0.09 mm resolution. The maximum output frequency of the sensor is 600 kHz. Figure 4 shows the string encoder position sensor used in the designed system.

String encoder bodies are mounted to the movement record cabinet, and each of the six wires is passed through separate guidance components to prevent tangles. Wire starting points are assumed as these components' positions. Hence, the positions of string encoder bodies are not important. Ends of these wires are attached to the teaching tool, as seen in Figure 2, to create triangles between the wire starting points and the teaching tool. These triangles are used for the measurements of the positions and the yaw angle, which is explained in Section 2.3. Positions of the wire starting points of the string encoders on the movement record cabinet can be found in Table 2. These positions are given in the sensor coordinate system, and it is not related to the global coordinate system.

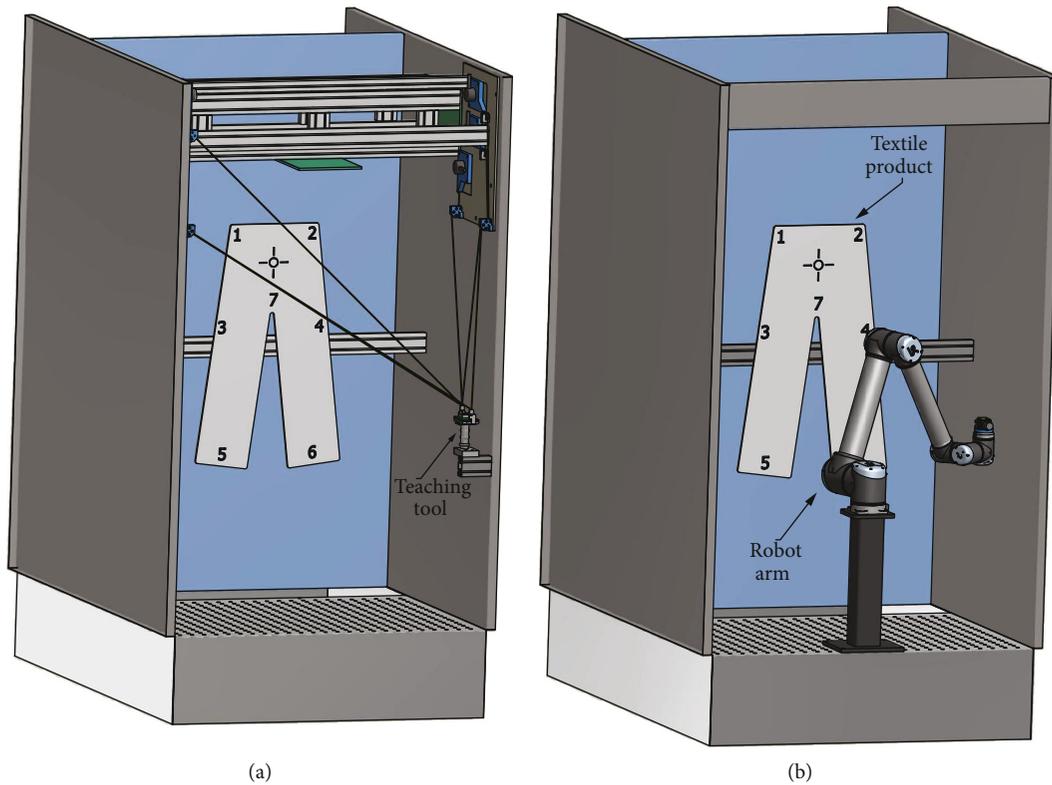


FIGURE 1: (a) Movement record cabinet with the teaching tool and sensor system. (b) Application cabinet with the robot arm.

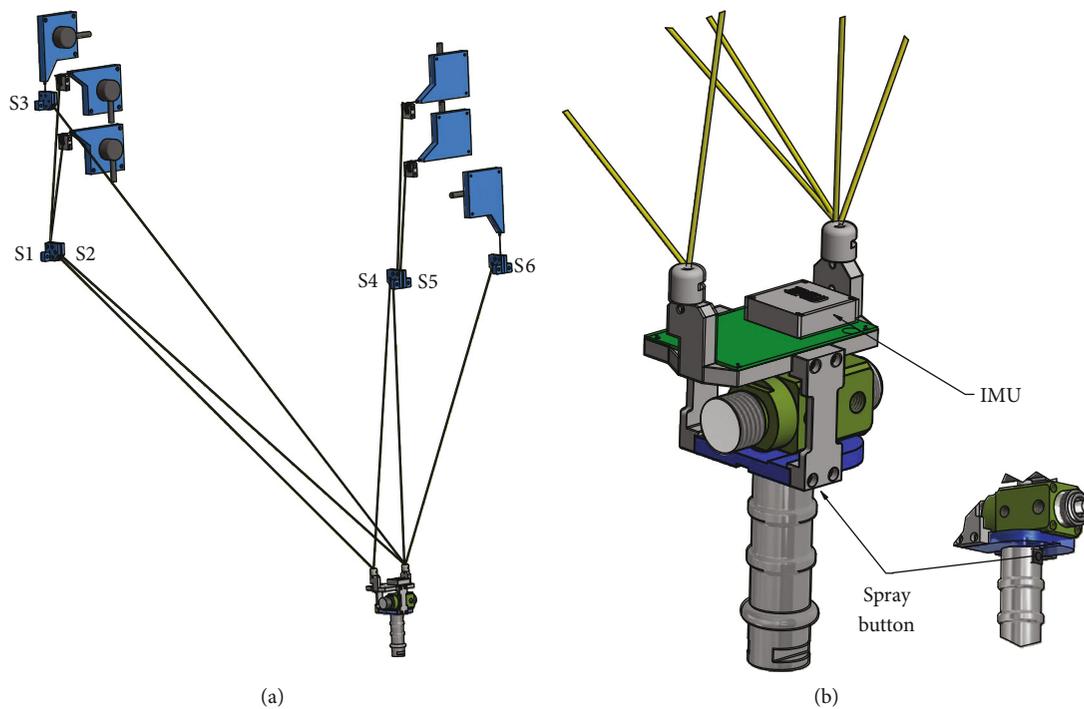


FIGURE 2: (a) Structure of the sensor system. (b) Teaching tool.

Another benefit of the string encoders is that they help the operator to carry the teaching tool's weight. Because the string encoder bodies are mounted at higher points than

the teaching tool operation site, the pulleys apply a force to carry the teaching tool. On the other hand, this force could worsen the response of the teaching tool at high speeds.

TABLE 1: Properties of the ADIS16480 MEMS IMU [29].

Property		ADIS16480		
Angular random walk		0.3°/√hr		
Velocity random walk		0.029 m/sec/√hr		
In-run bias stability		6°/hr		
Accuracy	Static	Roll 0.1°	Pitch 0.1°	Yaw 0.3°
	Dynamic	0.3°	0.3°	0.5°
		Gyroscope (°/sec)	Accelerometer (g)	Magnetometer (mGs)
Dynamic range		±450	±10	±2500
Output noise		0.16	0.0015	0.45
Filter options		Adaptive extended Kalman filter (EKF)		

However, this does not affect the proposed designed system because the system will be used in jobs such as spraying, which do not require high speeds.

**2.2.3. Robot.** After detailed research in the robot market, the UR10 robot was chosen for this study. The UR10 [31] is a 6-DOF serial robot which consists of six revolute joints. Each joint is driven by one motor separately, and each of them has a  $\pm 360$  rotation range and 120-180°/sec speed limit. Its working radius is 1300 mm from the base joint, it has a repeatability of  $\pm 0.1$  mm, and its weight is 28.9 kg. The UR10 has a general-purpose I/O (GPIO) port inside the control box which contains 8-bit digital input, 8-bit digital output, 2-bit analog input, and 2-bit analog output. It also has a tool I/O port at the tool end of the robot which provides 2-bit digital input, 2-bit digital output, and 2-bit analog input ports. All these GPIO ports can be used for adding equipment to the robot, such as sensors, motors, grippers, pneumatic valves, and safety equipment, or for communicating with other systems. The UR10 is a collaborative robot so that it has a force sensor on each joint. It is provided with force limits and collision detection systems. Thus, it allows operating without any additional safety sensors. The robot is equipped with an aluminum tool flange as an end-effector. It allows easy integration of a tool or sensor to the end-effector. The maximum payload handled by the robot attached to the end-effector is 10 kg.

The UR10 has three different programming levels: the Graphical User Interface Level (PolyScope GUI installed on a 12" touchscreen teach pendant), the Script Level (URScript), and the C-API Level. The URScript is a robot programming language developed by Universal Robots for the UR series robots. The URScript language has its own variables, data types, flow control statements, and functions for use in monitoring and controlling the movements of the robot and I/O ports. The URScript programs are executed in real time in the URControl RuntimeMachine (RTMachine). The URControl communicates with the robot at a frequency of 125 Hz, so the maximum sampling rate that the robot can be controlled at is 125 Hz. Programming a robot at the Script Level requires writing a client application (running on a different computer) and connecting to the URControl using a TCP/IP socket [32]. After establishing a

successful connection, the URScript programs can be sent from the computer to the URControl as a form of strings over the socket. The URControl performs calculations such as inverse kinematic calculations, torque calculations, and force calculations to get the tool center point (TCP) of the robot to the target points. Then, it generates and sends control commands to motor drivers of the UR10.

In this study, the designed system program written in the Pascal programming language creates a script program using built-in functions, e.g., *get\_inverse\_kin*, *servoj*, and *movej*, and variables of the URScript programming language. This script program includes information such as the target position and target speed for the TCP of the robot. The designed program communicates with the URControl at a sampling rate of 40 Hz to send commands. The actual status of the robot, e.g., the actual pose of the end-effector, is acquired by the Real-Time Data Exchange (RTDE) interface designed by Universal Robots.

**2.2.4. Position Estimation Using the Law of Cosines.** In trigonometry, the law of cosines is used for the following situations in a nonright triangle:

- (i) If the lengths of two sides and the angle between these sides are known, the length of the third side can be computed
- (ii) If the lengths of all three sides are known, all angles of the triangle can be computed

In an ABC triangle, let  $a$ ,  $b$ , and  $c$  be the lengths of a triangle and  $\alpha$ ,  $\beta$ , and  $\gamma$  are the opposite angles of these sides, respectively.

The law of cosines states that

$$\gamma = \arccos \left( \frac{a^2 + b^2 - c^2}{2ab} \right). \quad (1)$$

In this study, by using the measurements taken from the string encoders, triangles with known side lengths are obtained. In these triangles, the law of cosines is applied by using Equation (1) and the desired calculations are done. A

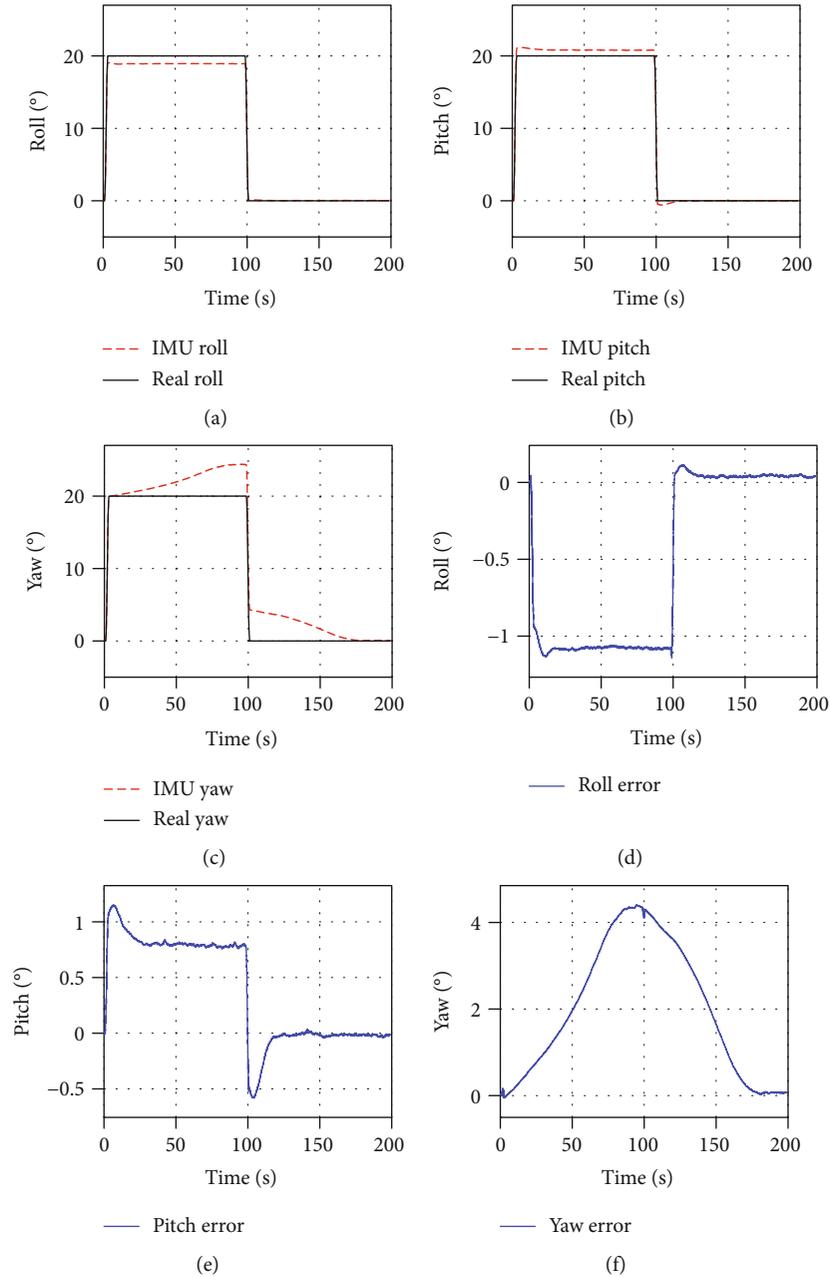


FIGURE 3: Test of the Euler angle outputs of the ADIS16480 on the dynamic movement and stationary state. (a) Roll angle output. (b) Pitch angle output. (c) Yaw angle output. (d) Roll angle error. (e) Pitch angle error. (f) Yaw angle error.

detailed description of these calculations is given in Section 2.3.

**2.2.5. Designed Spray System.** A spray system with nozzle end-effectors was designed. The parts of the spray system are as follows: a programmable logic controller (PLC) unit, an HMI, two nozzle end-effectors (also called a spray gun), a dye reserve tank, a manual air pressure regulator, three electropneumatic air pressure regulators, three mist separators, relays, valves, a compressed dry air inlet, rubber hoses, an electric panel box, and two communication cables. Nozzle end-effectors were mounted to the tool end of the UR10 robot and the teaching tool. The PLC unit and other electrical

components were mounted inside the panel box, and the HMI was mounted on the front side of the panel box. The program of the PLC is written in the Ladder Diagram (LD) programming language on Control FPWIN Pro programming software, which is developed by Panasonic Corporation. GUI of the HMI is designed in EasyBuilder Pro software, which is developed by Weintek Labs. The PLC unit has communications between the PC and the HMI through the RS-232 protocol. The reserve tank was used to reserve the liquid dye solution, which can be prepared by using potassium permanganate (PP) and water or any other dye materials. There is a mixer on top of the reserve tank to prevent the formation of residues inside the tank. The manual

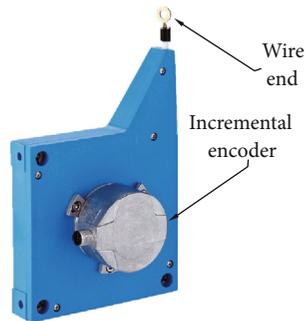


FIGURE 4: Wire draw mechanism and incremental encoder of the string encoder position sensor [30].

TABLE 2: Positions (in mm) of the wire starting points on the movement record cabinet in the sensor coordinate system.

Sensor number	x position (mm)	y position (mm)	z position (mm)
S1	0	0	0
S2	0	0	0
S3	0	0	434
S4	0	1378	0
S5	0	1378	0
S6	-400	1378	0

air pressure regulator was used to regulate the compressed dry air to desired levels and supply the air-driven components. Mist separators were used to clean the inlet compressed air from mist and particles. Outlets of the mist separators were connected to the electropneumatic air pressure regulators. The outlet of the electropneumatic regulator was connected to the reserve tank to adjust the dye pressure on the outlet of the tank, and outlets of the other two electropneumatic regulators were connected (commonly) to the nozzle end-effectors to adjust atomizing air pressure and spray pattern size. Electropneumatic regulators are communicating with the PLC unit through 4-20 mA analog communication. Pressure parameters of these electropneumatic regulators can be set and monitored from both the HMI and the PC. The other components are auxiliary components used to operate the spray system. The designed spray system and nozzle end-effectors are shown in Figure 5.

The operator can spray the dye solution into the desired areas of the denim by pressing and releasing the button, which was placed on the teaching tool. Pressing and releasing times are recorded with timestamps by the computer together with the positions and orientations of the teaching tool. Working periods of the spray gun of the robot are controlled according to this record.

In order to test the precision of the spray system, an image with a size of  $450 \times 618$  pixels was used. Contours of the image were calculated and transformed to pose data. A paper is placed in front of the robot, and the pose data was sent to the robot. Total spraying time was around 2 seconds for this test. The result of the test is shown in Figure 6.

According to this test, the spray system works perfectly and it satisfies the precision expectations of the target application.

**2.2.6. Designed System Controller Boards.** In this system, there are two designed and developed controller boards, which are shown in Figure 7. The first one is mounted on top of the teaching tool and called the orientation board. There are one ADIS16480 IMU, one microcontroller, one USB port, and various electrical components on the board. The microcontroller is STM32F407 MCU which has a 32-bit ARM Cortex-M4 core running at 168 MHz. The program of the microcontroller was written in the C++ programming language on the Keil Microcontroller Development Kit (MDK). This controller board's primary purposes are to supply the ADIS16480 IMU, to communicate with the IMU via the Serial Peripheral Interface (SPI) protocol, and to perform data transmission with the PC via the Universal Serial Bus (USB) port. The orientation angle values measured by the IMU can be read from the PC via this board. Also, all setting parameters of the IMU can be accessed and adjusted from the PC.

The second controller board works as a data acquisition board. There are one microcontroller, six 8-pin female connectors, six buffer amplifiers, one USB port, one 16-pin input-output socket, and various electrical components on the board. This microcontroller is also STM32F407 MCU, and the program of the microcontroller was written in the C++ programming language on the Keil MDK. Each string encoder produces an electrical signal proportional to its wire's linear extension. These signals are transferred to the controller board through 8-pin connectors (one for each encoder). These signals are processed on the buffer amplifiers and sent to the microcontroller. After that, the microcontroller transmits these signals to the PC via the USB port.

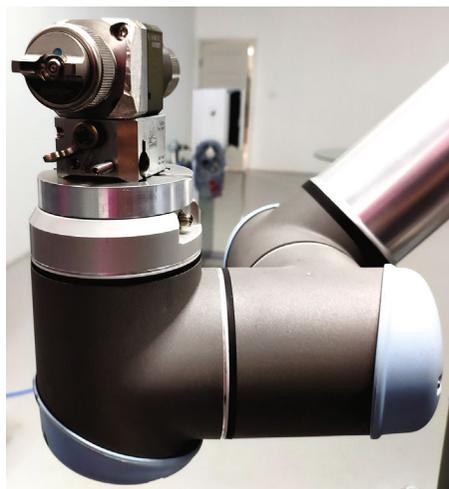
**2.2.7. Designed System Program.** The data collected from the controller boards are transferred to the PC via USB ports. The PC runs the designed system program which is written in the Pascal programming language by using the Delphi integrated development environment (IDE). This program calculates and records the  $x$ ,  $y$ ,  $z$ , roll, pitch, and yaw values of the tip of the teaching tool at a 40 Hz sampling rate during movement. Then, all these calculated values are rearranged in a form by using built-in functions and variables of the URScript and transferred to the controller of the robot (URControl) as a script code. The robot can perform these recorded movements at any time.

The program also has an interface to interact with the operator and shows the information about the system such as the real-time pose of the teaching tool and initialization parameters of the string encoders and IMU. The operator can start and stop the movement record, can see the list of all recorded movements, and can also select one of them to transfer it to the robot using the interface.

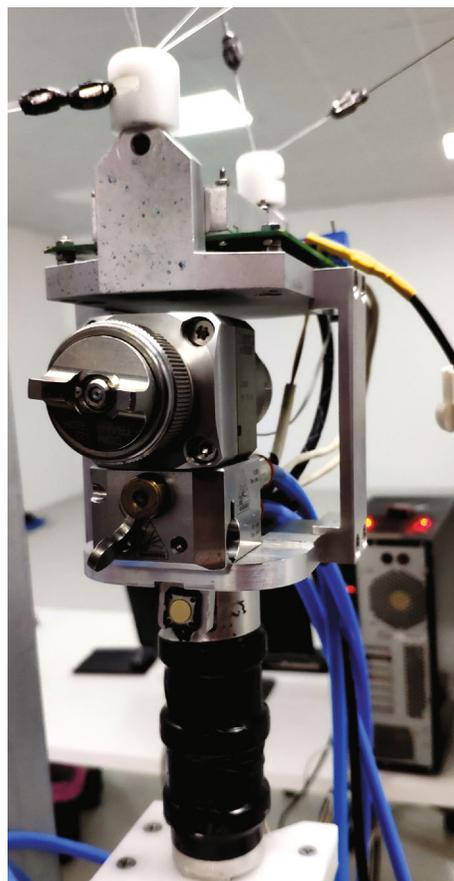
**2.3. Measurement of the Pose of the Tip of the Teaching Tool.** Figure 2 shows the developed sensor mechanism which combines six string encoders and one IMU to measure the position and orientation of the tip of the teaching tool.



(a)



(b)



(c)

FIGURE 5: (a) Spray system. (b) Nozzle end-effector mounted to the robot. (c) Nozzle end-effector mounted to the teaching tool.

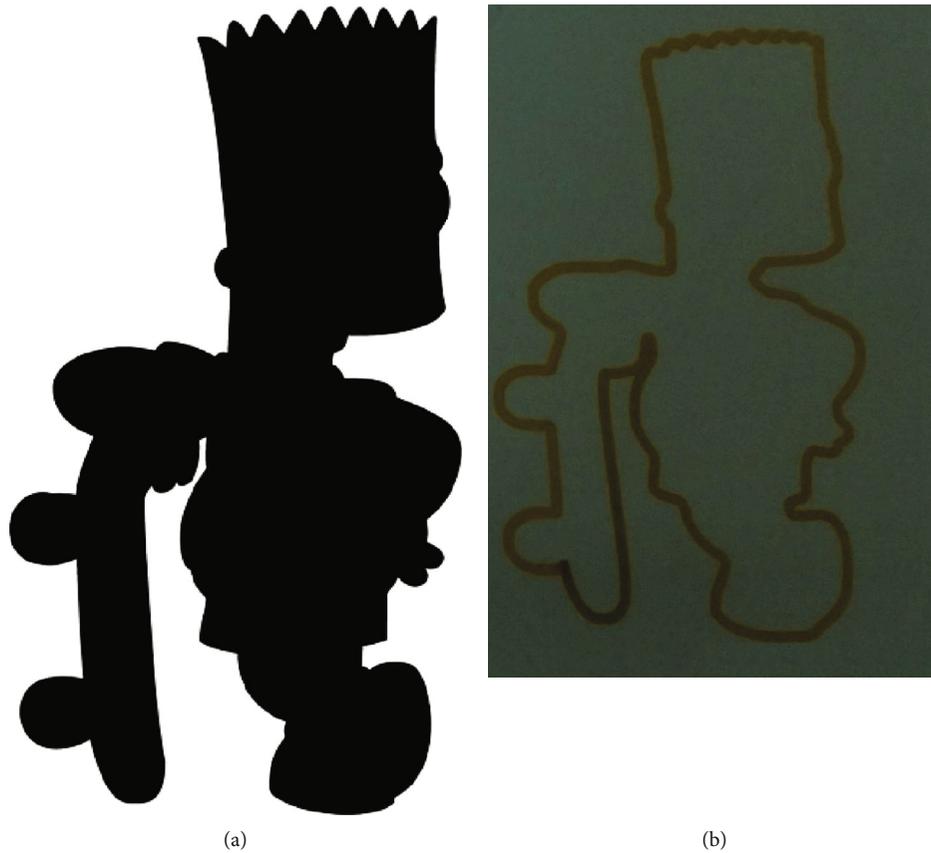


FIGURE 6: (a) Original image. (b) Result of spray by the robot.

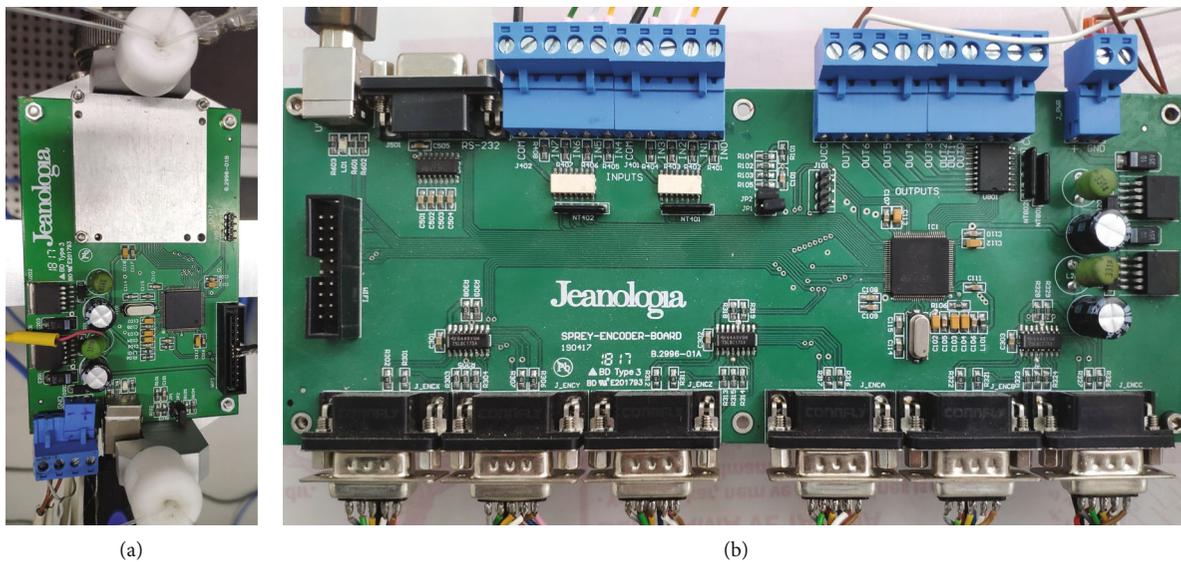


FIGURE 7: (a) Orientation board. (b) Data acquisition board.

The roll, pitch, and yaw angles are restricted to within  $\pm 75^\circ$  about the reference frame. This working range is enough for the aim of the application.

*2.3.1. Measurement of the x, y, and z Positions of the Tip of the Teaching Tool.* First of all, string encoder position sensors

consist of incremental encoders, and due to the nature of encoders, they start counting from zero. Therefore, the teaching tool is fixed to a known location, which can be seen in Figure 1, called the home position. The home position is assumed at the (0, 0, 0) point in the global coordinate system. The initialization process of the string encoders is always

done while the teaching tool is in the home position. The wire lengths of the string encoders at the home position are measured manually, and this data is entered into the system program. After the initialization process is finished, the calculation process of the  $x$ ,  $y$ , and  $z$  positions is automatically started inside the system program. Henceforth, the movement record process can be started by pressing the *start recording* button on the system program.

The selection of the positions of the guidance components and the measurements of the positions of the tip of the teaching tool is described as follows:

- (1) Two points are selected on the movement record cabinet. The first point can be chosen arbitrarily. However, the second point must be chosen on a different location on the observed axis and the same locations on the other two axes. This means that if the  $x$ -axis is the observed axis and if the first point is chosen as ( $x = 0, y = 0, z = 0$ ), then the second point must be chosen as ( $x = c, y = 0, z = 0$ ) and variable  $c$  can take any value. These points are defined as anchor points, and guidance components are mounted to these points. The fixed distance between the anchor points is measured
- (2) The cables that come out of two anchor points are fixed to the same point in front of the teaching tool
- (3) There is a triangle with all three side lengths known. The distances between two anchor points are known (fixed), and the distances between anchor points to the teaching tool are also known (read from the string encoder). Now, the law of cosines can be applied to this triangle
- (4) Calculate the observed axis position of the teaching tool according to the formulas below. Figure 8 illustrates the calculation procedure of the  $x$  position

On the triangle in Figure 8, TTF is the teaching tool front junction point of the four string encoders (S2, S3, S5, and S6). The  $x$  length is the distance to be calculated. The  $c$  length is the distance between guidance components of the string encoders S5 and S6, and this length is fixed to 400 mm. The  $a$  and  $b$  lengths are read from S5 and S6 string encoders. Using the formula in Equation (1), the law of cosines can be applied to the triangle S5TTFs6 and  $\cos \alpha$  can be calculated. On the triangle TTFs5B, the  $x$  length can be calculated using the following equation:

$$x = \cos \alpha * b. \quad (2)$$

The  $y$ - and  $z$ -axis position measurements are done with the same rules. For the  $y$ -axis position measurement, S2 and S5 string encoder measurements are used. For the  $z$ -axis position measurement, S2 and S3 string encoder measurements are used. String encoder numbers are defined in Figure 2.

**2.3.2. Measurement of the Yaw Angle.** When the teaching tool is stationary, IMU measurements must be constant. How-

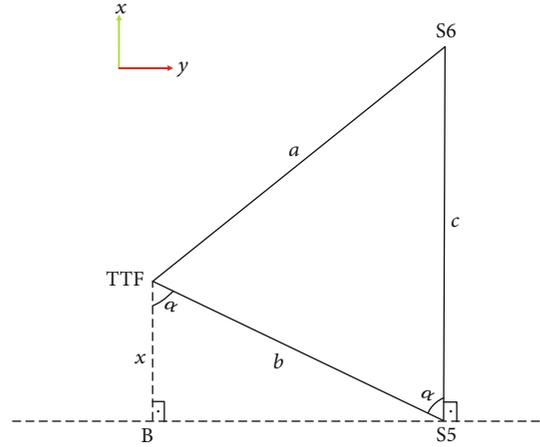


FIGURE 8: Visualization of the calculation of the  $x$  position.

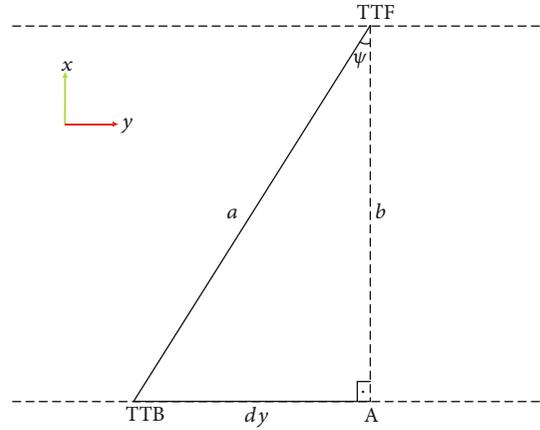


FIGURE 9: Visualization of the calculation of the yaw angle.



FIGURE 10: Test environment.

ever, as explained before, it is not possible to take constant measurements from IMU due to its nature. Notably, the yaw angle is very noisy since the  $z$ -axis is parallel to the gravity vector. The roll and pitch angle measurements of IMU always have better accuracy than the yaw angle measurement. In order to increase the system accuracy, the yaw angle

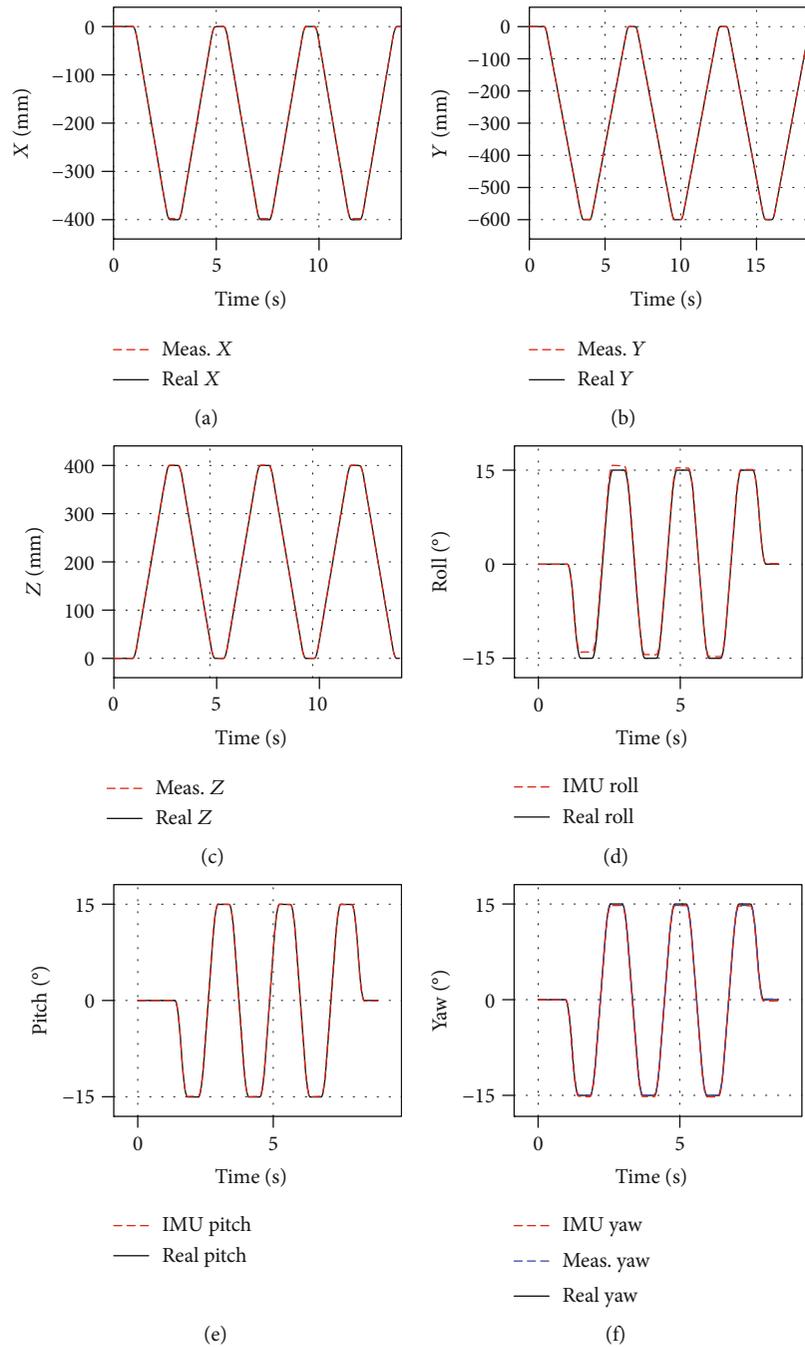


FIGURE 11: Position and orientation measurements on the independent movement test. (a)  $x$  position measurement. (b)  $y$  position measurement. (c)  $z$  position measurement. (d) Roll angle measurement. (e) Pitch angle measurement. (f) Yaw angle measurement.

is measured using string encoder position sensors, as shown in Figure 9. Hereafter, this angle will be referred to as measured yaw, while the yaw angle of the IMU will be referred to as IMU yaw.

In Figure 9, TTF is the teaching tool front junction point of four string encoders, whereas TTB is the teaching tool back junction point of two string encoders. The  $y$  position of the TTF is measured using S2 and S5 string encoders, and the  $y$  position of the TTB is measured using S1 and S4 string encoders. The  $dy$  length is the difference between these two

measurements.  $a$  is a fixed distance with a length of 123.5 mm. The yaw angle can be calculated using the following equation:

$$\psi = \arcsin \left( \frac{dy}{a * \cos \beta} \right) * \frac{180}{\pi}, \quad (3)$$

where  $\beta$  is the pitch angle of the teaching tool which is read from the IMU.

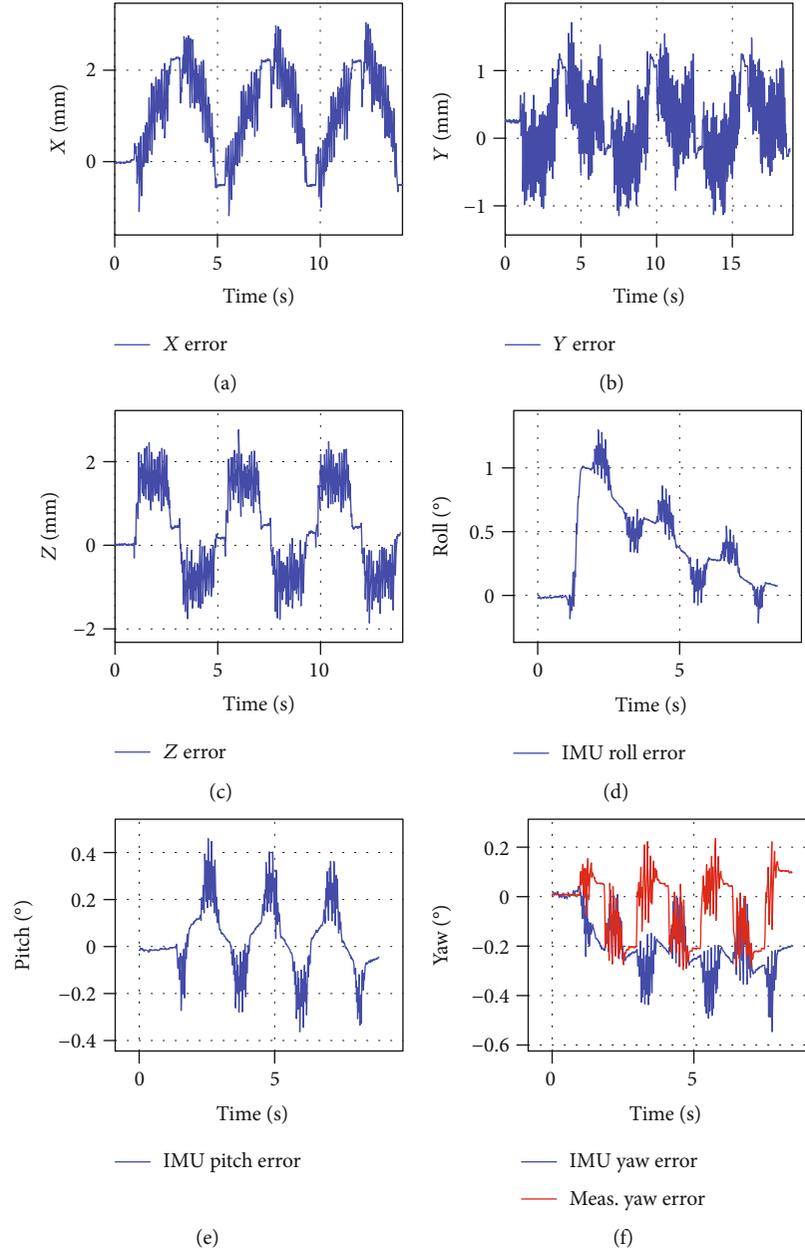


FIGURE 12: Position and orientation errors on the independent movement test. (a)  $x$  position error. (b)  $y$  position error. (c)  $z$  position error. (d) Roll angle error. (e) Pitch angle error. (f) Yaw angle error.

TABLE 3: RMS errors of position and orientation measurements on the independent movement test.

Observed data	RMS errors
$x$ (mm)	1.4054
$y$ (mm)	0.5842
$z$ (mm)	1.0868
IMU roll ( $^{\circ}$ )	0.5177
IMU pitch ( $^{\circ}$ )	0.1401
IMU yaw ( $^{\circ}$ )	0.2259
Measured yaw ( $^{\circ}$ )	0.1331

In contrast to the yaw angle output of the IMU, this solution provides stable and accurate results without any noise or error accumulation.

**2.3.3. Reading of the Roll and Pitch Angles.** The roll and pitch angle values of the teaching tool are directly read from the ADIS16480 IMU at a 40 Hz sampling rate. The yaw angle values are also read from the IMU for comparison with the measured yaw angle, but the yaw angle output of the IMU is not used in the system due to the relatively high errors. Internal EKF of the ADIS16480 is enabled for better orientation estimations. It was observed that the yaw angle output of the EKF has noticeably slow response times when the

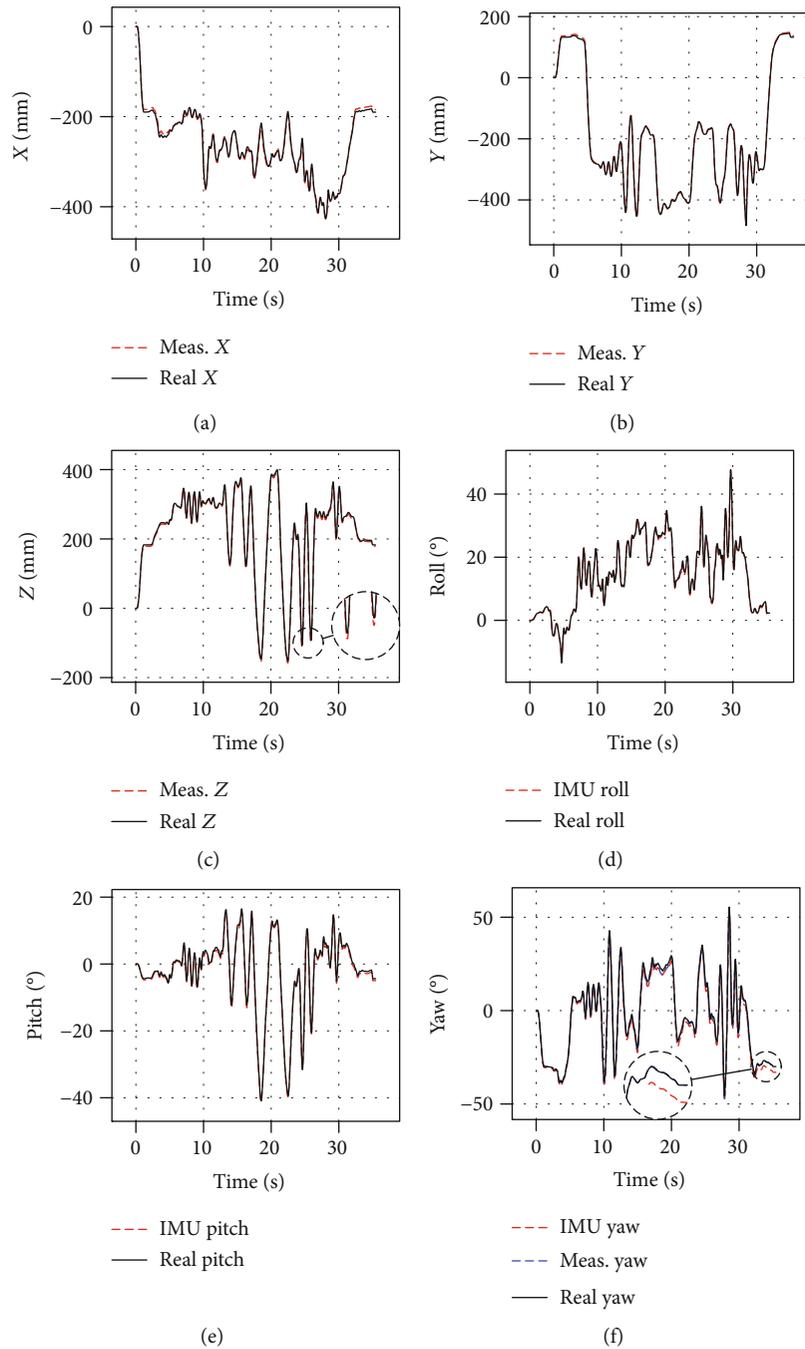


FIGURE 13: Position and orientation measurements on the compound movement test at 100% robot speed. (a)  $x$  position measurement. (b)  $y$  position measurement. (c)  $z$  position measurement. (d) Roll angle measurement. (e) Pitch angle measurement. (f) Yaw angle measurement.

magnetometer of the IMU is enabled. The magnetometer is disabled to prevent this problem.

### 3. Results

In order to test the accuracy of the pose measurement system, a test environment was created. The robot was moved from the application cabinet to the movement record cabinet, and the teaching tool was attached to the end-effector of the robot. The robot was used as an absolute reference, and the measurements taken from the pose measurement system

were compared with the absolute pose of the robot. The test environment is shown in Figure 10.

In the first experiment, the position and orientation measurements were tested independently. The reason for this is to see whether any calculation or software mistakes are present. In the first step, the robot is moved along the  $x$ -axis direction from 0 to  $-400$  mm and then back to 0 mm, followed by a few repetitive movements. In that time, the pose data was read from both the robot and the sensor system at a 40 Hz sampling rate. At the end of this test, the measured  $x$  position data is compared with the robot's measured  $x$

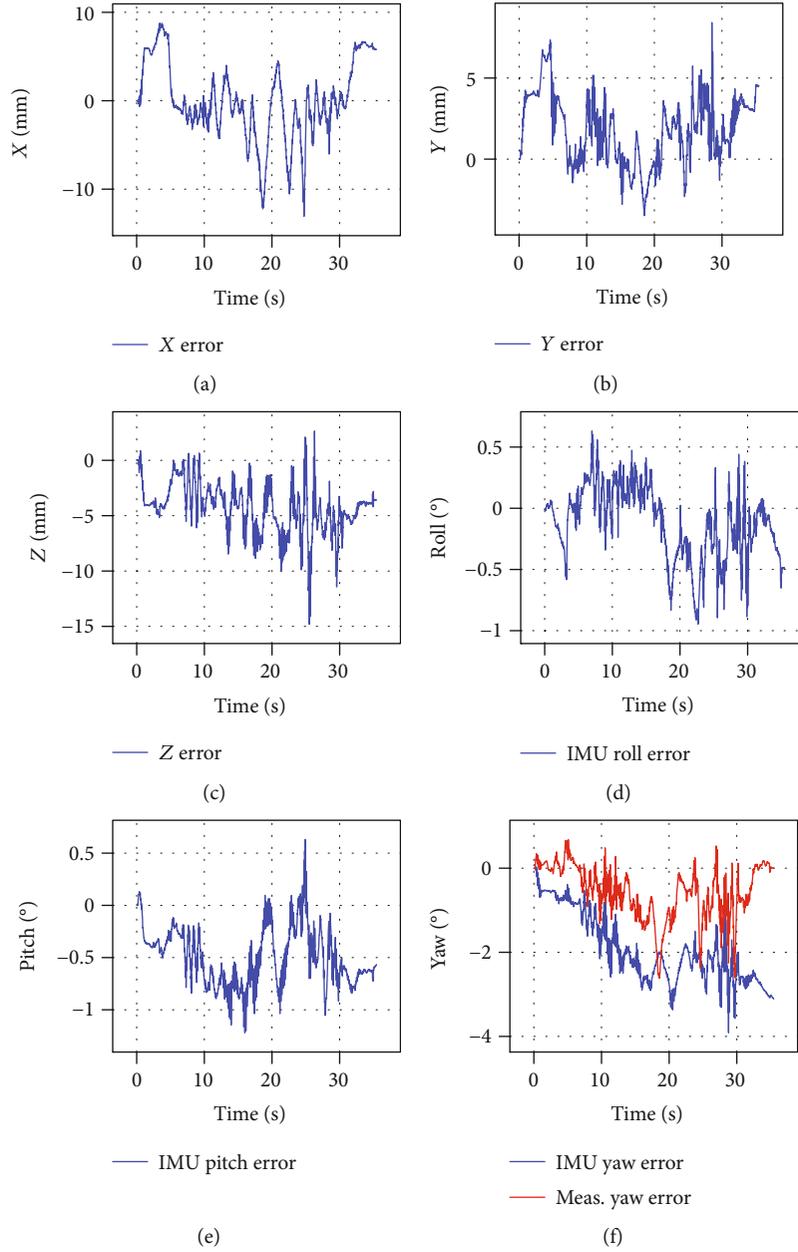


FIGURE 14: Position and orientation errors on the compound movement test at 100% robot speed. (a)  $x$  position error. (b)  $y$  position error. (c)  $z$  position error. (d) Roll angle error. (e) Pitch angle error. (f) Yaw angle error.

TABLE 4: RMS errors of position and orientation measurements on the compound movement test at various robot speeds.

Observed data	RMS errors at various robot speeds				
	20%	40%	60%	80%	100%
$x$ (mm)	4.3398	4.2952	4.3013	4.2866	4.3603
$y$ (mm)	2.7378	2.7135	2.6621	2.7068	2.7323
$z$ (mm)	4.5901	4.6159	4.5780	4.5721	4.7349
IMU roll ( $^{\circ}$ )	0.5484	0.3009	0.2795	0.2657	0.3144
IMU pitch ( $^{\circ}$ )	0.4362	0.3606	0.4358	0.5180	0.5627
IMU yaw ( $^{\circ}$ )	7.4704	5.3151	3.7204	2.7342	2.0912
Measured yaw ( $^{\circ}$ )	0.8195	0.7958	0.8306	0.8317	0.8390

position data. This experiment is done independently for the  $x$ ,  $y$ , and  $z$  position measurements and the roll, pitch, and yaw angle measurements. Measurements of each pose data of the sensor system and the robot are plotted as a function of time in Figure 11. Also, the errors (the difference between the measured pose by the sensor system and the robot pose) in measurements for each pose data are plotted as a function of time in Figure 12. Root mean square errors (RMSE) of the position and orientation measurements are shown in Table 3.

In the second experiment, a record which has been previously recorded by an operator in the movement record cabinet is sent to the robot. This record contains about 35.5 seconds of compound movements. Same as in the first experiment, pose measurements and robot pose data have been

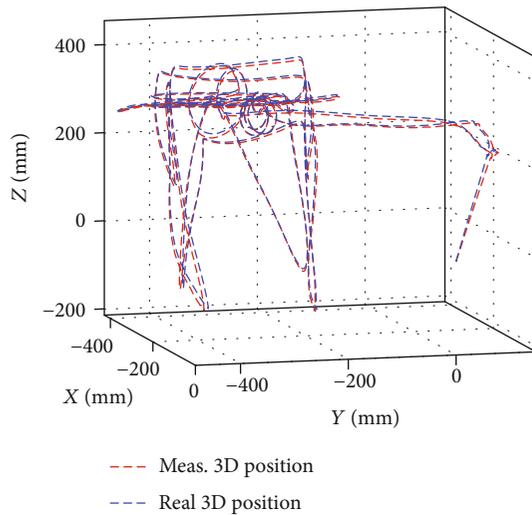


FIGURE 15: 3D position graph of the compound movement test.

recorded at a 40 Hz sampling rate during this movement. To observe the pose measurement system accuracy, the robot is moved with the same recorded data at various speeds: 20%, 40%, 60%, 80%, and 100%. 100% speed represents the original record speed of the operator, while 20% speed represents five times slower than the original record speed. Measurements of each pose data of the sensor system and the robot are plotted as a function of time in Figure 13. Also, the errors in measurements for each pose data are plotted as a function of time in Figure 14. RMS errors of the position and orientation measurements are shown in Table 4. The 3D position graph of the compound movement test is shown in Figure 15.

When Tables 3 and 4 are compared, it is seen that RMS errors in Table 4 are larger than the ones in Table 3 in all position measurements. The reason for these larger errors on positions is the unintentional changes that occur at wire lengths due to the interaction of the wires at the junction points during the 6-dimensional compound movement of the teaching tool. Also, the accuracy of the IMU is affected by compound movements, especially in the yaw angle. This is an expected result for the reasons explained earlier.

As seen from the position values given in Table 4, RMS errors are hardly ever affected by different robot speeds. When IMU RMS errors are examined, the roll and pitch axes are slightly affected with respect to the yaw axis. While the IMU yaw RMS error is  $7.47^\circ$  at 20% robot speed, it is  $2.09^\circ$  at 100% robot speed. However, measured yaw RMS errors are not affected by different robot speeds, as seen from Table 4. The correction rate obtained in 20% robot speed is approximately 89%, while it is 60% in 100% robot speed. This result shows that measured yaw is much more stable and reliable compared to IMU yaw.

#### 4. Discussion

Experimental results show that the 3D RMS position and orientation errors of the designed system are about 1.87 mm and  $0.55^\circ$  in the independent movement test and 6.99 mm and  $1.06^\circ$  in the compound movement test, respectively. If the

yaw angle values are read from the IMU instead of calculating with string encoders, then the 3D RMS orientation error rises to  $2.19^\circ$  in the compound movement test. In comparison to the previous version of this work [24], one can see that both the position and orientation errors are reduced due to using a better initialization procedure of the string encoders and the new method to measure the yaw angle. According to the results of [26], the average position and orientation errors of the system are about 5 mm and  $6^\circ$ , respectively. In [27], it can be seen that the position and orientation errors of the system are about 200 mm and  $9^\circ$ , respectively, in compound motion. In [28], the cable-driven parallel robot system has 9.4 mm position and  $1.2^\circ$  orientation errors, respectively. However, there is no information about the motion type, motion duration, or motion size. In [23], it is reported that the maximum position error is 37 mm. When all these results are evaluated, it can be seen that the proposed designed system gives a better position and orientation accuracy and precision.

This system is also tested in a real-time application. The results show that the robot follows the operator's movements with a time delay of 25 ms. This time delay can be further reduced by increasing the communication speed between the URControl and the system program. For the UR10 robot, the time delay can be reduced down to 8 ms approximately.

#### 5. Conclusion

In this paper, a custom-designed pose measurement system which is used to record movements of the tip of the teaching tool is presented. In addition to this, a spraying application which uses the designed system in teleoperation of a robot manipulator is presented. The designed system includes an IMU, six string encoder position sensors, a UR10 robot manipulator, a spraying system, two custom controller boards, and a custom system program. The  $x$ ,  $y$ , and  $z$  positions of the tip of the teaching tool are measured by using four string encoders, the yaw angle is measured by adding two more string encoders, and the roll and pitch angles are read from the IMU. A significant accuracy improvement has been achieved by measuring the yaw angle with string encoders instead of using the yaw angle output of the IMU. The law of cosines is applied for measurements of the  $x$ ,  $y$ , and  $z$  positions and the yaw angle. First, the designed system measures the position and orientation of the tip of the teaching tool and records movements which are provided by a human operator. At the same time, the system also records spraying periods during the movement record. Then, the recorded movements and spraying job are executed by an industrial robot at the same speed or up to four times faster than the operator without compromising quality. For data acquisition, two controller boards were designed for an IMU-and-string encoder system. A user interface program for the designed system was developed in the Pascal language by the Delphi compiler. The performance of the designed system is tested with a configuration where the robot is moved to the movement record cabinet, and the teaching tool is fixed to the end-effector of the robot. The 3D RMS errors are about 1.87 mm in positioning and  $0.55^\circ$  in orientation

angles in the independent movement test and 6.99 mm in positioning and 1.06° in orientation angles in the compound movement test. Experimental results show that the designed system satisfies the accuracy and precision expectations of the target application. Besides, it is seen that the designed system has better accuracy and precision than similar systems which also use string encoders and/or IMU. The offline and real-time implementations of the proposed system are possible. The application area of the designed system is not limited to the spraying process. The designed system can be used successfully in industrial automation applications such as bleaching, painting, and palletizing without any robot programming knowledge.

The future work of this study is planned in two phases. The first phase is aimed at correcting yaw angle deviations by processing raw data from the gyroscope, accelerometer, and magnetometer of IMU using sensor fusion algorithms. In the second phase, it is aimed at improving the created sensor fusion algorithm so that it can calculate both position and orientation with high accuracy. These planned future studies will simplify the design by reducing the number of string encoders so that this will add portability to the movement record system and increase the variety of target applications.

### Data Availability

The movement record datasets used to support the findings of this study are currently under embargo while the research findings are commercialized. Requests for data (6/12 months) after the publication of this article will be considered by the corresponding author.

### Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

### Acknowledgments

The patent application of this work has been submitted to the Turkish Patent and Trademark Office with the application number 2019/05154.

### References

- [1] G. A. Dickson and D. R. Gray, "Chemical for bleaching textiles," 1990, US Patent 4,919,842.
- [2] M. H. Korayem, "Path planning of mobile elastic robotic arms by indirect approach of optimal control," *International Journal of Advanced Robotic Systems*, vol. 8, no. 1, p. 10, 2011.
- [3] M. H. Korayem, V. Azimirad, H. Vatanjou, and A. H. Korayem, "Maximum load determination of nonholonomic mobile manipulator using hierarchical optimal control," *Robotica*, vol. 30, no. 1, pp. 53–65, 2012.
- [4] M. Gao, Y. Lin, and Y. Yan, "Design and implementation of teach pendant for six degrees of freedom industrial robot," in *2015 Fifth International Conference on Instrumentation and Measurement, Computer, Communication and Control (IMCCC)*, pp. 1929–1933, Qinhuangdao, China, September 2015.
- [5] B. Akgun, M. Cakmak, J. W. Yoo, and A. L. Thomaz, "Trajectories and keyframes for kinesthetic teaching: a human-robot interaction perspective," in *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*, pp. 391–398, Boston, Massachusetts, USA, 2012.
- [6] P. Božek, A. al Akkad M, P. Blištan, and I. Ibrahim N, "Navigation control and stability investigation of a mobile robot based on a hexacopter equipped with an integrated manipulator," *International Journal of Advanced Robotic Systems*, vol. 14, no. 6, 2017.
- [7] A. Kilin, P. Bozek, Y. Karavaev, A. Klekovkin, and V. Shestakov, "Experimental investigations of a highly maneuverable mobile omnivheel robot," *International Journal of Advanced Robotic Systems*, vol. 14, no. 6, 2017.
- [8] S. R. Kirsch, C. Schilling, and G. Brunner, "Assesment of metallic distortions of an electromagnetic tracking system," in *Medical Imaging 2006: Visualization, Image-Guided Procedures, and Display*, vol. 6141, p. 61410J, International Society for Optics and Photonics, 2006.
- [9] T. B. Moeslund and E. Granum, "A survey of computer vision-based human motion capture," *Computer Vision and Image Understanding*, vol. 81, no. 3, pp. 231–268, 2001.
- [10] S. Aoyagi, A. Kohama, Y. Nakata, Y. Hayano, and M. Suzuki, "Improvement of robot accuracy by calibrating kinematic model using a laser tracking system-compensation of non-geometric errors using neural networks and selection of optimal measuring points using genetic algorithm," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5660–5665, Taipei, Taiwan, October 2010.
- [11] F. Thomas, E. Ottaviano, L. Ros, and M. Ceccarelli, "Performance analysis of a 3-2-1 pose estimation device," *IEEE Transactions on Robotics*, vol. 21, no. 3, pp. 288–297, 2005.
- [12] J.-P. Merlet, "Kinematics of the wire-driven parallel robot marionet using linear actuators," in *2008 IEEE International Conference on Robotics and Automation*, pp. 3857–3862, Pasadena, CA, USA, May 2008.
- [13] F. Thomas, E. Ottaviano, L. Ros, and M. Ceccarelli, "Uncertainty model and singularities of 3-2-1 wire-based tracking systems," in *Advances in Robot Kinematics*, pp. 107–116, Springer, 2002.
- [14] S.-H. P. Won, F. Golnaraghi, and W. W. Melek, "A fastening tool tracking system using an imu and a position sensor with Kalman filters and a fuzzy expert system," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 5, pp. 1782–1792, 2009.
- [15] D. Tomaszewski, J. Rapinski, and M. Smieja, "Analysis of the noise parameters and attitude alignment accuracy of ins conducted with the use of MEMS-based integrated navigation system," *Acta Geodynamica et Geomaterialia*, vol. 12, no. 2, pp. 197–209, 2015.
- [16] Z. He and Z. Wang, "Motion measurement using inertial sensors, ultrasonic sensors, and magnetometers with extended Kalman filter for data fusion," *IEEE Sensors Journal*, vol. 12, no. 5, pp. 943–953, 2012.
- [17] R. Atac and E. Foxlin, "Scorpion hybrid optical-based inertial tracker (HOBIT)," in *Head-and Helmet-Mounted Displays XVIII: Design and Applications*, vol. 8735, p. 873502, International Society for Optics and Photonics, 2013.
- [18] R. Pirnk, M. Hrubaš, D. Nemeč, T. Mravec, and P. Božek, "Integration of inertial sensor data into control of the mobile platform," *Federated Conference on Software Development and Object Technologies*, 2015, pp. 271–282, Springer, 2015.

- [19] J. Georgy, T. Karamat, U. Iqbal, and A. Nouredin, "Enhanced MEMS-IMU/odometer/GPS integration using mixture particle filter," *GPS Solutions*, vol. 15, no. 3, pp. 239–252, 2011.
- [20] S. Zhang, W. Wang, and T. Jiang, "WiFi-inertial indoor pose estimation for micro aerial vehicles," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 5, 2020.
- [21] G. Nützi, S. Weiss, D. Scaramuzza, and R. Siegwart, "Fusion of IMU and vision for absolute scale estimation in monocular slam," *Journal of Intelligent & Robotic Systems*, vol. 61, no. 1–4, pp. 287–299, 2011.
- [22] M. Alataise and G. Hancke, "Pose estimation of a mobile robot based on fusion of IMU data and vision data using an extended Kalman filter," *Sensors*, vol. 17, no. 10, p. 2164, 2017.
- [23] S.-h. P. Won, W. Melek, and F. Golnaraghi, "Fastening tool tracking system using a Kalman filter and particle filter combination," *Measurement Science and Technology*, vol. 22, no. 12, p. 125108, 2011.
- [24] E. Yavuz, Y. Senol, M. Ozcelik, and H. Aydin, "Design of a string-encoder and IMU based 6D pose measurement system," in *2019 11th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, pp. 1–6, Pitesti, Romania, June 2019.
- [25] A. Boukerche and E. Nakamura, "Localization systems for wireless sensor networks," *IEEE Wireless Communications*, vol. 14, no. 6, pp. 6–12, 2007.
- [26] J. Zhang, L. I. Jinsong, Q. I. Liwei, and Z. Dandan, "Kinematic analysis of a 6-DOF wire-based tracking device and control strategy for its application in robot easy programming," in *2009 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 1591–1596, Guilin, China, December 2009.
- [27] J. Andrade-Cetto and F. Thomas, "Wire-based tracking using mutual information," in *Advances in Robot Kinematics*, pp. 3–14, Springer, 2006.
- [28] H. Wang, T. Gao, J. Kinugawa, and K. Kosuge, "Finding measurement configurations for accurate robot calibration: validation with a cable-driven robot," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1156–1169, 2017.
- [29] Analog Devices Inc, "Datasheet of ADIS16480," August 2018.
- [30] A. G. Sick, "Online datasheet of EcoLine PFG13-A1CM0544 wire draw encoders," March 2019.
- [31] Universal Robots, "Service manual of UR10/CB3, revision UR10 En 3.1.3," October 2018.
- [32] Universal Robots, "The URScript programming language, version 3.8," March 2019.