Journal of Machine Construction and Maintenance
PROBLEMY EKSPLOATACJI
QUARTERLY
ISSN 1232-9312 4/2017 (107)
p. 81–89

Kamil STATECZNY, Mirosław PAJOR, Karol MIĄDLICKI, Mateusz SAKÓW
Department of Mechanical Engineering and Mechatronics, West Pomeranian University of Technology, Szczecin, Poland
kamil.stateczny@zut.edu.pl, miroslaw.pajor@zut.edu.pl, karol.miadlicki@zut.edu.pl, mateusz.sakow@zut.edu.pl

MEMS BASED SYSTEM FOR CONTROLLING AND PROGRAMING INDUSTRIAL MANIPULATOR FANUC S-420F USING GESTURES

Key words: manipulator, manual control, operator interface, gesture control, MEMS sensor, FANUC KAREL, FANUC S-420F.

Abstract: The paper presents a mobile system dedicated to an industrial manipulator control. For this purpose, an interface was designed to be used for intuitive interaction with a robot, which consists of a measuring scanner for the orientation of the human upper limb, and a glove to measure the deflection of fingers. Basic gestures used for moving the manipulator have been proposed. The paper was based on the previous experience obtained in the implementation of projects which entailed manual programming of CNC machine tools. The system shall be a starting point for the construction of a control interface for other mechanical devices.

System sterowania i programowania robota przemysłowego FANUC S-420F za pomocą gestów bazujący na sensorach MEMS

Słowa kluczowe: robot, manipulator przemysłowy, interfejs operatora, sterowanie gestami, czujniki MEMS, FANUC S-420F, Fanuc Karel, HMI.

Streszczenie: W pracy przedstawiono mobilny system do poruszania manipulatorem przemysłowym. W tym celu zaprojektowano interfejs służący do intuicyjnej interakcji z robotem, na który składa się skaner mierzący orientację kończyny górnej oraz rękawicę do mierzenia ugięcia palców dłoni. Zaproponowano podstawowe gesty do poruszania manipulatorem. Pracę oparto na dotychczasowych doświadczeniach zdobytych przy realizacji projektów nad programowaniem manualnym obrabiarek CNC. System stanowi punkt wyjściowy do budowy interfejsu obsługi innych urządzeń mechanicznych.

Introduction

This article is the result of a non-classical approach in manipulator control. So far, the authors of the article have concentrated on a design of a system for manual control and programming of CNC machine tools [1, 2]. The manual control techniques used with virtual reality interface were implemented to control a manipulator available at the Institute of Manufacturing Engineering. Unfortunately, operators are not able to communicate with the device by using machine code in a relatively simple way, so they have to be equipped with different types of human-machine interfaces to allow the communication. Nowadays, there is an expanding trend moving towards to intuitive and easy-to-use control system for devices. There are many facilities used for the controlling and positioning of manipulators, so there is no need to control the device from the keyboard by entering the desired position. Manipulators are controlled by different types of levers, control panels, and touchpads. Some robots have the ability to program the desired movement by grabbing a robot arm and moving in the intended direction. There has also been research on voice [3–5], vision [6, 7], and sensor fusion [8, 9] interfaces used to control industrial manipulators. Although such programming is the most intuitive, it is not suitable for large robots, because the operator is smaller and unable to reach the tip of the manipulator.

The paper presents a developed method dedicated to manipulator control by tracking the operator's upper limbs motion. The work has been narrowed down to determine only the position of tool centre point of the manipulator in the base coordinate system of the manipulator and without determining its orientation. All measurements, except finger bending measurements, are used to determine the orientation in a given space without a distance measurement. The operator can control the manipulator from any distance. He can even be outside the work area of the manipulator.

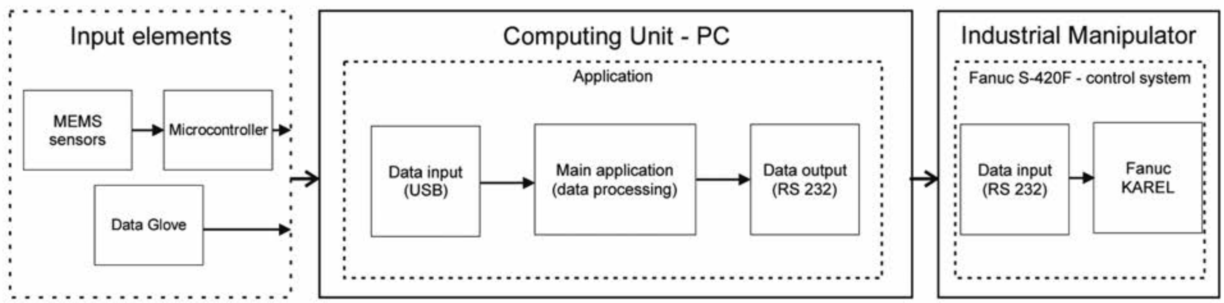

Fig. 1. Block diagram of the operator interface

1. Operator interface

The operator interface (Fig. 1) consists of input elements and a computing platform.

1.1. Input elements

The inputs elements of the operator interface are MEMS sensors and the 3D Data Glove 4 Ultra. The glove is made of bend sensors to determine fingers bending. In this paper, we applied a technique using the acceleration measurements (including gravity), the angular velocity, and the magnetic field measured by means of a tri-axis accelerometer, a tri-axis gyroscope, and a tri-axis magnetometer, respectively. These sensors were integrated in a complete system ADIS16405. Its advantage is that it has a compensation of the input voltage impact and a compensation of the impact of temperature for each of the built-in sensors [10]. It is factory-calibrated sensitivity, with bias and axial alignment.

A gyroscope measures the angular velocity, and by knowing the initial conditions, it is possible to integrate the measured signal with time in order to obtain orientation. However, a measurement error leads to an aggregate error of the transformed orientation, which makes the determination of absolute orientation impossible. An accelerometer measures the Earth's gravity in an absolute

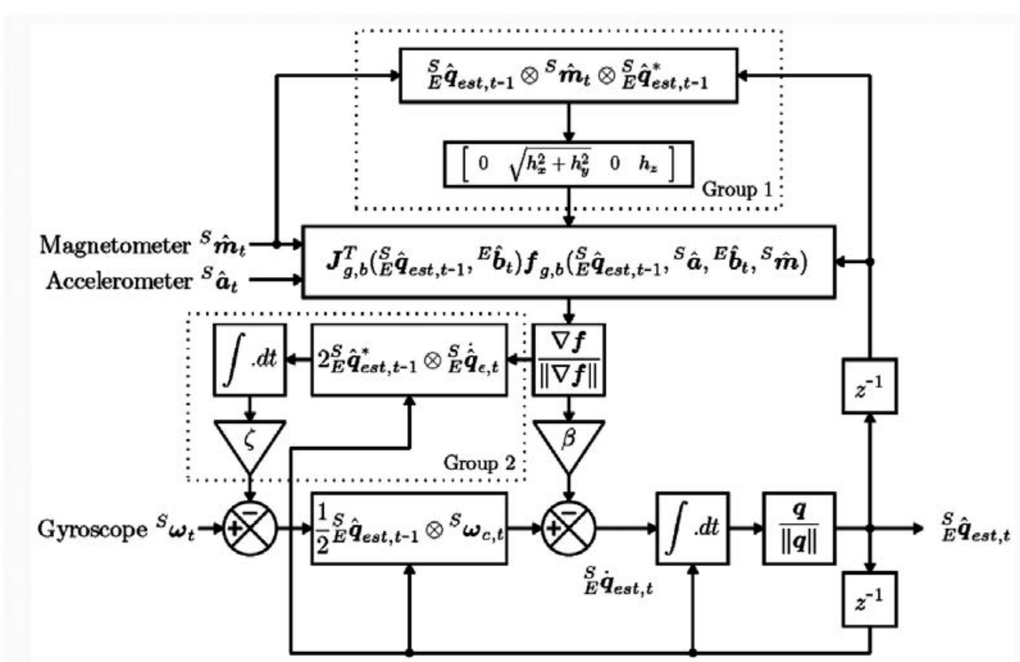

Fig. 2. Block diagram of the orientation for an MARG implementation

system, but there are strong measurement interferences during its movement. The magnetometer measures the Earth's magnetic field, but disturbances from other magnetic sources in the environment could change the readings. In order to calculate orientation on the basis of measurements from the gyroscope, the accelerometer, and the magnetometer, we have used an orientation filter [11] for the MARG (Magnetic, Angular Rate, and Gravity) implementation, whose block diagram is presented in Fig. 2. This filter is characterized with a low computational cost, which enables its implementation on inexpensive microcontrollers. It achieves a higher level of accuracy in comparison to the Kalman filter applied for calculating orientation [11]. The filter was implemented on the STM32F103CBT6 microcontroller on the ZL30ARM instruction set. Figure 3 presents the acceleration sensors along with the platform on which the filter was implemented. The ADIS16362 sensors communicates with the microcontroller by means of an SPI interface (Serial Peripheral Interface), sending the acceleration and angular velocity values necessary for calculating the orientation by means of the filter. Orientation in space is represented by means of a quaternion. Data is transmitted to the computer by means of a USB interface.

1.2. Computing Unit

Since it is a prototype system, due to the implementation speed, the control procedure is implemented on a PC using C++ programming language. The main application modules are responsible for receiving input data and processing them, and for sending information selected for the manipulator control system.

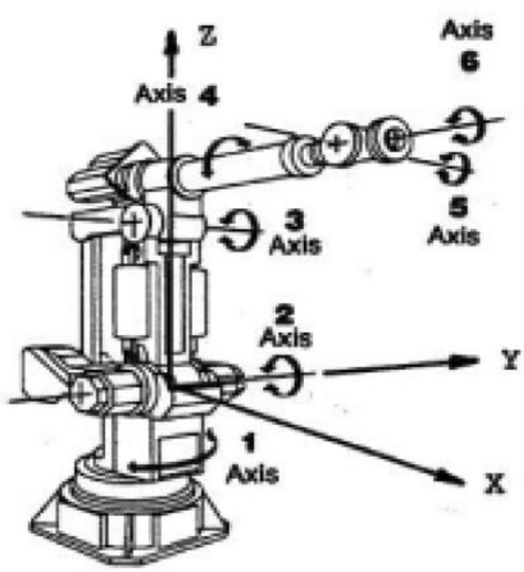

Fig. 4. Design of the FANUC S-420F

1.3. Industrial Manipulator

A FANUC S-420F robot with a RH controller was used to implement and conduct tests of the proposed control system. The robot is driven by 6 servo motors responsible for motion in the respective axes (Fig. 4).

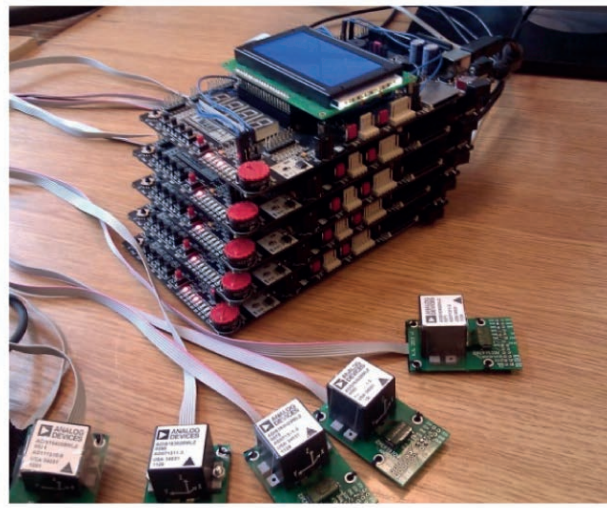

Fig. 3. Sensors along with the microcontrollers

The RH controller is a modular system dedicated to control FANUC industrial manipulators. It consists of the main CPU, a path tracking processor, divided RAM, a bubble memory, and servo motor control systems. Six I/O cards are available with a total of 48 inputs, 48 digital outputs, and two RS-232 communication ports. The device can communicate with external systems, e.g., to feed the status of a production line. The controller also contains six servo amplifiers. A comprehensive description of the robot and the controller can be found in the technical specifications [12]. The RH controller can be programmed by using FANUC Karel language. A comprehensive description of the syntax of the programming language, the user manual for CRT/KB operation control panel, the KCL command prompts, system variables, and error codes can be found in the technical specifications [13]. To communicate with the control system of the manipulator, a serial interface RS232 was used. This interface was used to send information about the desired position of the end-effector of the manipulator. The data can be refreshed every 200 ms. By using the programming language (Karel), an application was written that allowed the system to interpret the received data (desired position – incremental and global). It has also been possible to determine the speed and acceleration that the manipulator should reach to achieve the desired position. The control system implements inverse kinematics and a trajectory generator with linear interpolation.

2. Proposed methods of control

The research was focused on two ways of controlling the manipulator: a "remote control" and by using a system for measuring the position of the hand in the operator coordinate system.

2.1. Control by using a remote device

Figure 5 shows the operator holding the remote control containing an orientation sensor. In addition, Figure 5 also shows the applied frames: {E} – earth frame, {R} – robots frame, {S} – sensor frame, and {P} – remote control frame.

Orientation sensor provides the orientation of the earth relative to the sensor frame. In this case, ${}^{S}_{E}\hat{\boldsymbol{q}}$ and ${}^{P}_{E}\hat{\boldsymbol{q}}$ define the orientation of the earth frame {E} relative to the sensor frame {S} and the orientation of the earth frame {E} relative to the remote-control frame {P}, respectively. Unit quaternion ${}^{A}_{B}\hat{\boldsymbol{q}} = [w, x, y, z]$ may be regarded as a representation of rotation, where the vector determines the direction of axis rotation and the angle θ = *2 arccos(w)* determined the angle by which the rotation has been made [14, 15]. It is possible to translate the quaternion to the rotation matrix unambiguously, which is presented by Eq. 1.

$$
{}^{A}_{B}\boldsymbol{R} = \begin{bmatrix} 1-2\left(y^2+z^2\right) & 2\left(xy-zw\right) & 2\left(xz+yw\right) \\ 2\left(xy+zw\right) & 1-2\left(x^2+z^2\right) & 2\left(yz-xw\right) \\ 2\left(xz-yw\right) & 2\left(yz+xw\right) & 1-2\left(x^2+y^2\right) \end{bmatrix} \tag{1}
$$

Fig. 5. Coordinate systems for controlling by remote device

The sensor S is used to determine the orientation of the remote control relative to the orientation of the robot frame. Therefore, it is necessary to set it in the S sensor orientation as the robot frame. Thanks to using the sensor, the operator does not have to manually enter data of the mutual orientations of these frames. Quaternions are converted into rotation matrices according to Eqs. (2) and (3):

$$ {}^{S}_{E}\hat{\boldsymbol{q}} \rightarrow {}^{S}_{E}\boldsymbol{R} \tag{2}$$

$$ {}^{P}_{E}\hat{\boldsymbol{q}} \rightarrow {}^{P}_{E}\boldsymbol{R} \tag{3}$$

Since the frame {S} has the same orientation as frame {R}; therefore,

$$ {}^{R}_{E}\boldsymbol{R} = {}^{S}_{E}\boldsymbol{R} \tag{4}$$

To describe the orientation of the remote control in the frame {R}, Eq. 5 was used.

$$ {}^{P}_{E}\boldsymbol{R}\,{}^{R}_{E}\boldsymbol{R}^{T} = {}^{P}_{R}\boldsymbol{R} \tag{5}$$

Because it was only possible to control the position, not the speed, the incremental position of the tool centre point was sent to the control system of the manipulator. The vector shift of the tool centre of the manipulator was set as follows:

$$ {}^{R}\boldsymbol{V} = {}^{P}_{R}\boldsymbol{R}\,{}^{P}\boldsymbol{V} \tag{6}$$

where

$$ {}^{P}\boldsymbol{V} = \begin{bmatrix} \delta \\ 0 \\ 0 \end{bmatrix} \tag{7}$$

where
δ – fixed step movement.

The direction of the vector ${}^{P}\boldsymbol{V}$ was determined by the orientation of the remote control. Sense was selected by pressing one of two buttons. After the button was pressed, the incremental position ${}^{R}\boldsymbol{V}$ was send to the manipulator's control system.

2.2. Control by using a movement scanner

For a more intuitive control of the manipulator, a scanner of the operator's upper limb movements was designed. The scanner determines the position of the hand in the operator frame. The sensors have been deployed on the operator's body and have been taken into consideration of the movement system of the human body muscles. Since it was impossible to position the sensors on each bone and determine their orientation, we constructed a simplified model of human kinematics and deployed sensors in strategic locations to determine the position of hands in space. Fig. 6 shows the distribution of the sensors systems.

The point defining the position of the hands was calculated according to Fig. 7. Coordinate systems for controlling the robot using the operator's hand movements are shown in Fig. 8.

Quaternions are converted into rotation matrices according to Eq. (8):

$$ {}^{Si}_{E}\hat{\boldsymbol{q}} \rightarrow {}^{Si}_{E}\boldsymbol{R} \tag{8}$$

The position of the hand in the frame {O} is determined according to the following equation:

$$ {}^{O}\boldsymbol{P}_{n+1} = \prod_{k=1}^{n} {}^{Sk}_{E}\boldsymbol{R}^{T}\,{}^{Sk}\boldsymbol{P}_{k+1} \tag{9}$$

where
n – the number of segments in the scanner.

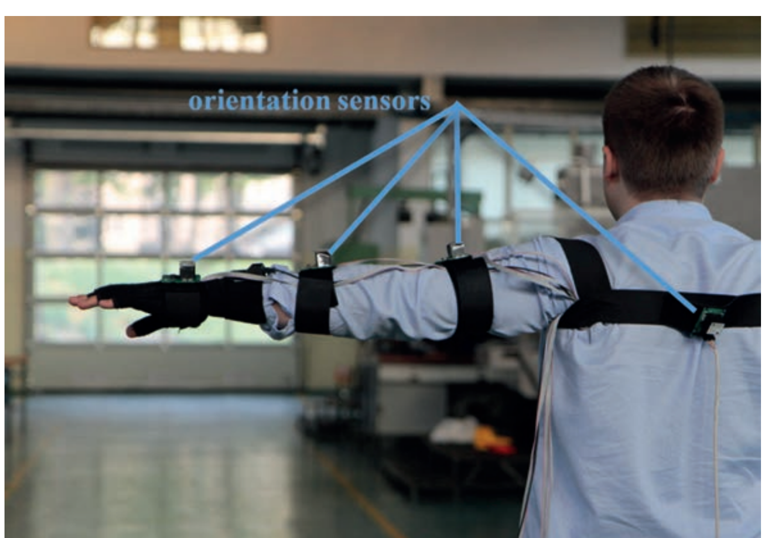

Fig. 6. Distribution of the orientation sensors on a operator eqqupied with momement scanner

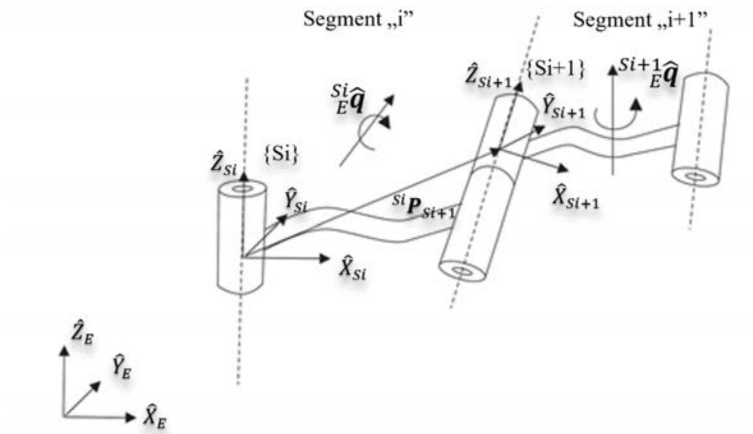

Fig. 7. Determining the position of segments

Fig. 8. Coordinate systems used in movement scanner calculations

The frame {O} corresponds to i = 0; therefore, it is the base system for the scanner of the upper limb movements. The operator position in the frame {E} is unknown, but his orientation relative to the frame is known. Performing a hand gesture, it is possible to designate a free vector.

$$ {}^{O}\boldsymbol{V} = {}^{O}\boldsymbol{P}'_{n+1} - {}^{O}\boldsymbol{P}_{n+1} \tag{10}$$

where

${}^{O}\boldsymbol{P}_{n+1}$ – hand position at time t,

${}^{O}\boldsymbol{P}'_{n+1}$ – hand position at time $t + \Delta t$ where Δt is sampling time.

Vector $^{O}\boldsymbol{V}$ can be expressed in the frame {R} according to the following relation:

$$^{R}\boldsymbol{V} = {}^{R}_{E}\boldsymbol{R}\,{}^{O}_{E}\boldsymbol{R}^{T}\,{}^{O}\boldsymbol{V} \tag{11}$$

Movement of the manipulator is possible only after closing the hand. Incremental position $^{R}\boldsymbol{V}$ is then sent to the control system.

3. Experimental results

The proposed systems were tested on a test stand in the technology hall of the Institute of Materials Science and Engineering, West Pomeranian University of Technology, Szczecin. The tests were focused on the possibilities of using the method and the ease of control, rather than on the accuracy of the method. In the tests, the operator moved his hand along the designated trajectories. During the movements, the hand and robot TCP positions were recorded. In the first experiment, the desired trajectory was a circle with a radius of 200 mm. The results of the first experiment are presented in Figure 9 and 10.

In the second experiment, the original trajectory was an ellipse with a radius of the minor axis of 200 mm and the major axis of 250 mm.

The experiments revealed that the proposed robot control system was much more intuitive than the operation control panel (teach pendant) or command line or the programming language (KAREL) provided by the manufacturer. In the tested system, the operator did not have to know how to use the operation control panel or command line or know specific commands. The operator could move the robot TCP along circles and ellipses without programming, and these trajectories would be difficult to perform even using a tech pedant.

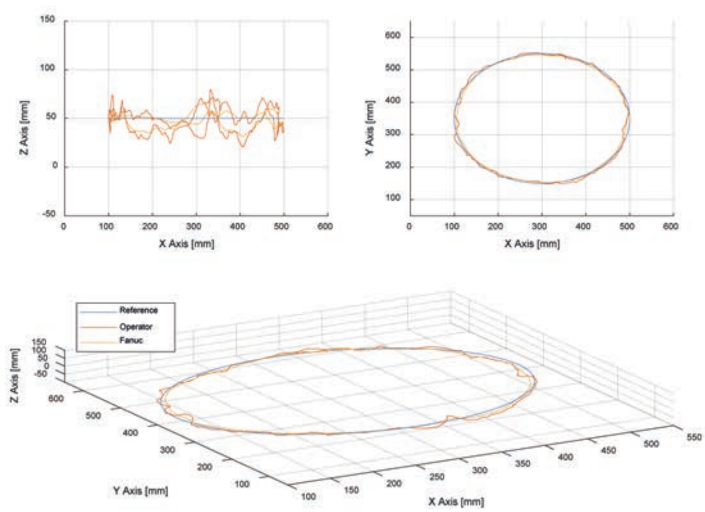

Fig. 9. Method A results – moving along circle with radius 200 mm

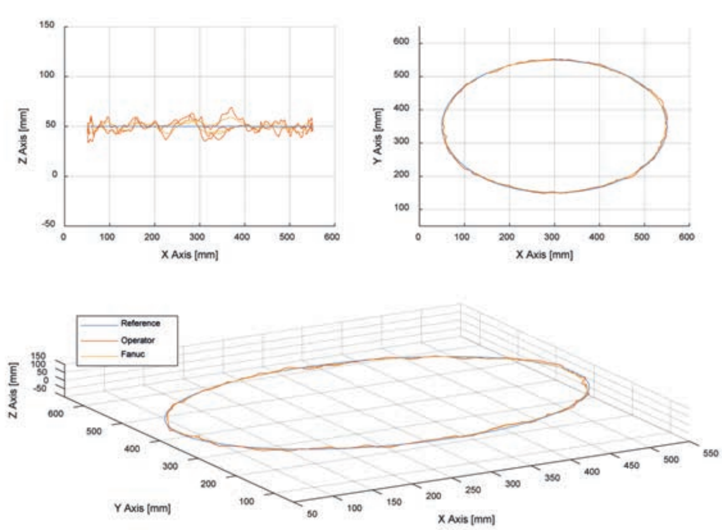

Fig. 10. Method B results – moving along circle with radius 200 mm

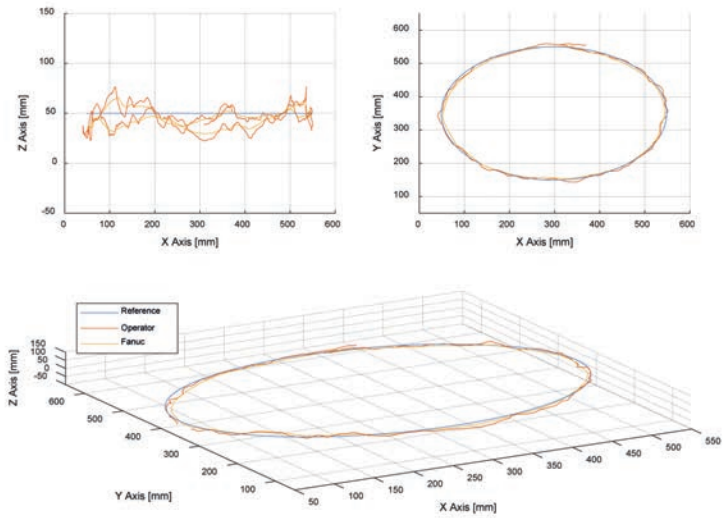

Fig. 11. Method A results – moving along ellipse with a radius of minor axis 200 mm and major axis 250 mm

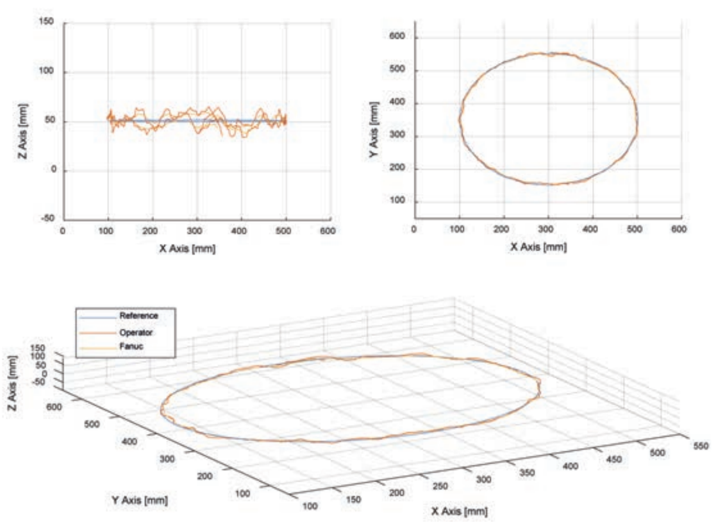

Fig. 12. Method B results – moving along ellipse with a radius of minor axis 200 mm and major axis 250 mm

However, the proposed system has its drawbacks, which are mainly caused by hardware imperfections. The main issues are caused by slow interface between robot RH controller and MEMS controller. When operator moved quickly, the reaction of the robot was delayed. Another disadvantage affecting the accuracy of the system is the magnetic interference between sensors. In the first control method, where one sensor was used, the errors were about ±20 mm. In the second method, where there are more sensors, errors were about ±30 mm. Given the purposes of the system, the values are sufficient for tasks such as pre-positioning or moving objects. It is also possible to apply additional support functions that improve accuracy.

Summary

The paper presents and discusses the concept and the constituent elements of the proposed system and test results. Conducted studies have shown that it is easier to control the robot by remote control containing an orientation sensor as well as using a scanner of the upper limb movements based on the orientation sensors for determining the location. Better smooth movement using the proposed remote-control systems would be obtained if it was possible to control the speed of the tool centre point of the manipulator in a given direction and sense. It was worth noting the fact that the magnetic interference can cause erroneous determination of orientation. Therefore, an optical supervisory system might be sufficient. With smaller energy efficient sensors, it will be possible to design a system hidden in clothing providing wireless communication. Since the main aim of the research was to test the possibility of using a new control approach to operate industrial manipulators, in the next stage of the project, the precision can be improved.

References

1. Stateczny K., Pajor M., “Project of a manipulation system for manual movement of CNC machine tool body units,” Advances in Manufacturing Science, vol. 35, pp. 33–41, 2011.
2. Stateczny K., “Construction of a system scanning the movement of human upper limbs," Zeszyty Naukowe/Akademia Morska w Szczecinie, no. 32 (104) z. 1, pp. 75–80, 2012.
3. Laureiti C. et al., “Comparative performance analysis of M-IMU/EMG and voice user interfaces for assistive robots,” in 2017 International Conference on Rehabilitation Robotics (ICORR), 2017, pp. 1001–1006.
4. Majewski M., Kacalak W., “Conceptual Design of Innovative Speech Interfaces with Augmented Reality and Interactive Systems for Controlling Loader Cranes,” in Artificial Intelligence Perspectives in Intelligent Systems, Vol 1, vol. 464, R. Silhavy, R. Senkerik, Z. K. Oplatkova, P. Silhavy, and Z. Prokopova, Eds. (Advances in Intelligent Systems and Computing, Berlin: Springer-Verlag Berlin, 2016, pp. 237–247.
5. Zinchenko K., Wu C.Y., Song K.T., “A Study on Speech Recognition Control for a Surgical Robot,” IEEE Transactions on Industrial Informatics, vol. 13, no. 2, pp. 607–615, 2017.
6. Shirwalkar S., Singh A., Sharma K., Singh N., “Telemanipulation of an industrial robotic arm using gesture recognition with Kinect,” in 2013 International Conference on Control, Automation, Robotics and Embedded Systems (CARE), Jabalpur, 2013, pp. 1–6: IEEE.
7. Pajor M., Miądlicki K., Saków M., “Kinect sensor implementation in FANUC robot manipulation,” Archives of mechanical technology and automation, vol. 34, pp. 35–44, 2014.
8. Yepes J.C., Yepes J.J., Martínez J.R., Pérez V.Z., “Implementation of an Android based teleoperation application for controlling a KUKA-KR6 robot by using sensor fusion,” in 2013 Pan American Health Care Exchanges (PAHCE), 2013, pp. 1–5.
9. Bonilla I. et al., “A vision-based, impedance control strategy for industrial robot manipulators,” in 2010 IEEE International Conference on Automation Science and Engineering, 2010, pp. 216–221.
10. (12.08.2017). ADIS16405 Data Sheet. Available: http://www.analog.com/en/products/sensors/inertial-measurement-units/adis16405.html.
11. Madgwick S.O.H., Harrison A.J.L., Vaidyanathan R., “Estimation of IMU and MARG orientation using a gradient descent algorithm,” in 2011 IEEE International Conference on Rehabilitation Robotics, 2011, pp. 1–7.
12. “Fanuc Robotics Maintenance manual [MARMKS42H1174EF][B-67205EG01].pdf,” ed.
13. “Fanuc Robotics MAROKENHA0885EF – Enhanced KAREL Operations Manual v. 2.22 R.pdf,” ed.
14. DeLoura M., Perełki programowania gier: vademecum profesjonalisty. Helion, 2002.
15. Matulewski J., Grafika, fizyka, metody numeryczne: symulacje fizyczne z wizualizacją 3D. Wydawnictwo Naukowe PWN, 2010.