# Human Capabilities Projection: Haptic Device for EOD Robot Control

Daniel CONSTANTIN, Andrei ȚIGĂU, Daniel TOMA, and Dragoș DOGARU

***Abstract*—Human capabilities projection is the robotic manifestation of human-like dexterity and sensory perception through robotic telemanipulation. The goal of human capabilities projection is to leverage robotic systems to accomplish tasks that may not be practical or safe for human execution. Human capabilities projection requires advances in robotic manipulation technologies, intuitive operator control modalities, immersive visual feedback, and effective haptic feedback.**

***Index Terms*—haptic device, robot, manipulation, sensors.**

## I. Introduction

Warfighters, astronauts, and first responders must be able to perform challenging tasks for extended periods of time under stressful conditions. Such tasks include military operations that must be performed in life-threatening battlefield environments or are of long duration, as well as operations related to space and planetary surface exploration [1].

These operational environments may contain hazardous conditions under which personnel must execute combat casualty care, engage improvised explosive devices, or investigate potential chemical, biological, radioactive, or nuclear threat conditions. Many injuries and fatalities occur when highly skilled personnel such as explosive ordnance disposal (EOD) technicians and field medics are in harm's way within these conditions. Finally, there are repetitive or mundane tasks that put our warfighters and first responders at risk of letting their guards down; examples of these types of tasks include forward area logistics, manning security checkpoint stations, border monitoring, maintenance actions on nuclear reactors, and battlefield clearance [1].

Using robotic systems with human-like manipulation capabilities in hazardous environments allows the user to avoid exposure and accomplish the operational task or scenario from a safe standoff range or location [2].

Currently fielded systems available to military personnel have limited capability to address objects in a remote environment in a human-like fashion [2].

These systems typically use low-dexterity end effectors [one degree of freedom (DOF)] and have a single manipulator arm. They typically provide operator control through a series of joysticks, switches, and buttons and lack direct user haptic feedback.

In human beings, part of a manipulation task is executed through visual feedback and the other part through haptic sensation feedback. To achieve true human capabilities projection, we feel that both elements must be present. Haptic feedback can be subdivided into sensations delivered to the skin (tactile) and those that are due to a sense of location in space (kinesthetic or proprioceptive). These sensations are extremely important when conducting manipulation tasks.

Kinesthetic feedback in experimental EOD robots can improve manipulations required for disabling explosive devices [6].

## II. Objectives

The scope of the research was to design a control system for a robotic arm based on the user's hand movement.

In the research, we focused on the following aspects:

– Mechanical transmission;
– Power supply;
– Sensors block;
– Feedback block;
– Interconnecting these four blocks with Raspberry Pi 3 and Python 3.7;
– Testing the final product.

## III. Haptic Device and Hardware Modules

To design the haptic device, electronics and mechanical elements were used (Fig. 3 and Fig. 1). The gripper was designed in CATIA and is put in motion by two independent actuators, each of them acting separately on one axle (Fig. 3).

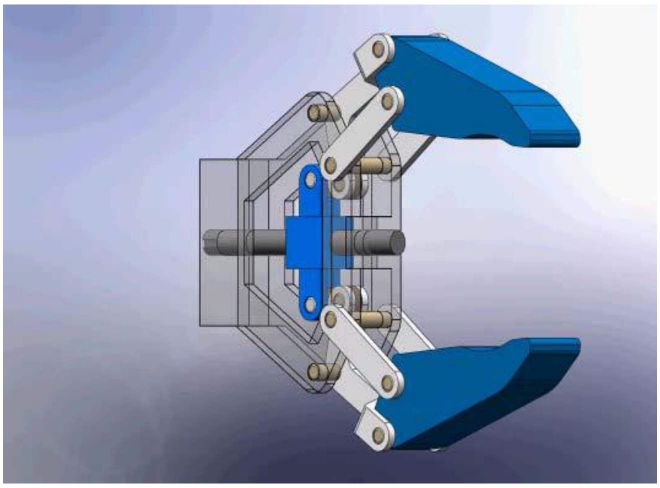

Figure 1. The gripper (CATIA)

D. CONSTANTIN is with the Military Technical Academy "Ferdinand I", 050141, Bucharest, Romania (e-mail: daniel.constantin@mta.ro)

A. ȚIGĂU is with the Military Equipment and Technologies Research Agency, 077025, Clinceni, Ilfov, Romania (e-mail: tigau.andreii@gmail.com).

D. TOMA is with the Armaments Department, 061418, Bucharest, Romania.

D. DOGARU is with the Military Equipment and Technologies Research Agency, 077025, Clinceni, Ilfov, Romania.

Digital Object Identifier 10.32754/JMT.2019.1.02

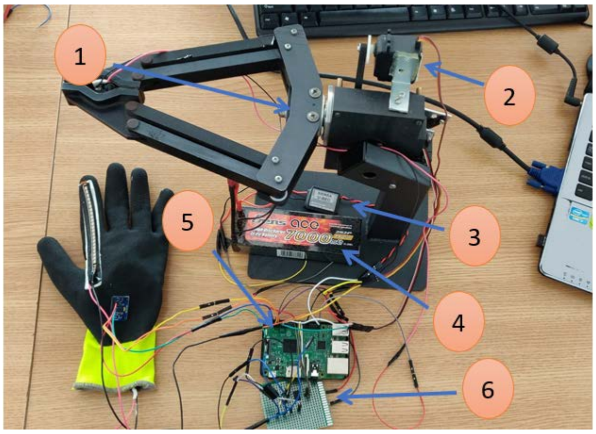


Figure 2. The haptic control system: 1-gripper, 2-GWS S03N-servo motor, 3-voltage stabilizer 7.4V-5V, 4-baterry 7,4V/7000mAh, 5- Raspberry PI, 6-ADC 0832

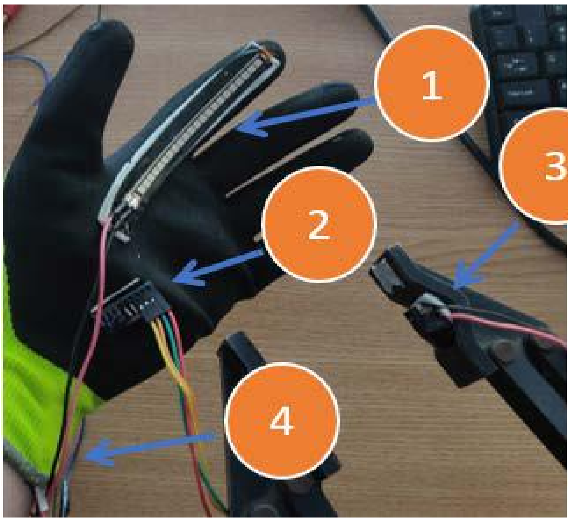


Figure 3. Sensors module: 1-flex sensor, 2-gyro sensor MPU 9255, 3-feedback button, 4-feedback vibration motor

The gripper has the following features:

TABLE I. GRIPPER SPECIFICATIONS

| | |
|---|---|
| Length | 35 cm |
| Width (open) | 27 cm |
| Width (close) | 18 cm |
| The distances between the gripper arms | 18 cm |
| Height | 22 cm |
| Weight | 0.85 kg |

From the analysis of the gripper, we found the need to use servo motor based on the following features:

- Opening angle of the gripper is less than 90°;
- The gripper must generate the natural movements of a hand, its rotation must not be more than 180°;
- Its size is small but the power is high;
- High precision.

The servo-motor has the following features:

TABLE II. GRIPPER SPECIFICATIONS

| | |
|---|---|
| Length | 40 mm |
| Width | 20 mm |
| Height | 35.5 mm |
| Weight | 41 g |
| Torque | 4.8V: 3.38 kg/cm |
| | 6.0: 4.03 kg/cm |
| Speed | 4.8V: 0.23/60° |
| | 6.0V:0/18/60° |
| Pulse period | 20 ms |
| Pulse width | 500-1500 µs |

The power supply of the two servo-motors is made using a 7000 mAh Gens Li-Po battery with 7.4 V output voltage and a Sierra U-Bec voltage stabilizer with an input voltage of 6.5 to 30 V and an output voltage of 5 V / 3 A.

The control signals for the two motor controls are generated by the control and control unit. The control and control unit is a Raspberry PI 3 model B module.

The gyroscopic sensor (MPU 9255) used to determine the degree of rotation of the hand and the transmission of the values to the gripper is a multi-chip module (MCM) composed of two molds integrated into a single QFN package (Quad Flat No Leads - circuits).

The MPU-9255 consists of three independent vibratory MEMS rate gyroscopes, which detect rotation about the X-, Y-, and Z- Axes. When the gyros are rotated about any of the sense axes, the Coriolis Effect causes a vibration that is detected by a capacitive pickoff. The resulting signal is amplified, demodulated, and filtered to produce a voltage that is proportional to the angular rate.

The MPU-9255's 3-Axis accelerometer uses separate proof masses for each axis. Acceleration along a particular axis induces displacement on the corresponding proof mass, and capacitive sensors detect the displacement differentially. The MPU-9255's architecture reduces the accelerometers' susceptibility to fabrication variations as well as to thermal drift. When the device is placed on a flat surface, it will measure 0g on the X- and Y-axes and +1 g on the Z-axis. The accelerometers' scale factor is calibrated at the factory and is nominally independent of supply voltage. Each sensor has a dedicated sigma-delta ADC for providing digital outputs. The full scale range of the digital output can be adjusted to ±2 g, ±4 g, ±8 g, or ±16 g.

The 3-axis magnetometer uses highly sensitive Hall sensor technology. The magnetometer portion of the IC incorporates magnetic sensors for detecting terrestrial magnetism in the X-, Y-, and Z- Axes, a sensor driving circuit, a signal amplifier chain, and an arithmetic circuit for processing the signal from each sensor. Each ADC has a 16-bit resolution and a full scale range of ±4800 µT.

This voltage is digitized using individual on-chip 16-bit Analog-to-Digital Converters (ADCs) to sample each axis. The full-scale range of the gyro sensors may be digitally programmed to ±250, ±500, ±1000, or ±2000 degrees per second (dps). The ADC sample rate is programmable from 8,000 samples per second, down to 3.9 samples per second, and user-selectable low-pass filters enable a wide range of cut-off frequencies [24].

The embedded Digital Motion Processor (DMP) is located within the MPU-9255 and offloads computation of motion processing algorithms from the host processor. The DMP acquires data from accelerometers, gyroscopes, magnetometers and additional 3rd party sensors, and processes the data. The resulting data can be read from the DMP's registers or can be buffered in a FIFO. The DMP has access to one of the MPU's external pins, which can be used for generating interrupts [24].

The purpose of the DMP is to offload both timing requirements and processing power from the host processor. Typically, motion processing algorithms should be run at a high rate, often around 200 Hz, in order to provide accurate results with low latency.

 Digital Object Identifier 10.32754/JMT.2019.1.02

The MPU-9255 communicates to a system processor using either a SPI or an I2C serial interface. The MPU- 9255 always acts as a slave when communicating to the system processor. The LSB of the of the I2C slave address is set by pin 9 (AD0) [24].

The MPU-9255 has limited capabilities as an I2C Master and depends on the system processor to manage the initial configuration of any auxiliary sensors. The MPU- 9255 has an interface bypass multiplexer, which connects the system processor I2C bus (SDA and SCL) directly to the auxiliary sensor I2C bus (AUX_DA and AUX_CL) [24].

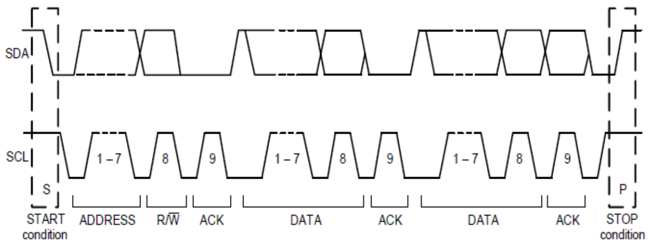


Figure 4. Complete I2C Data Transfer

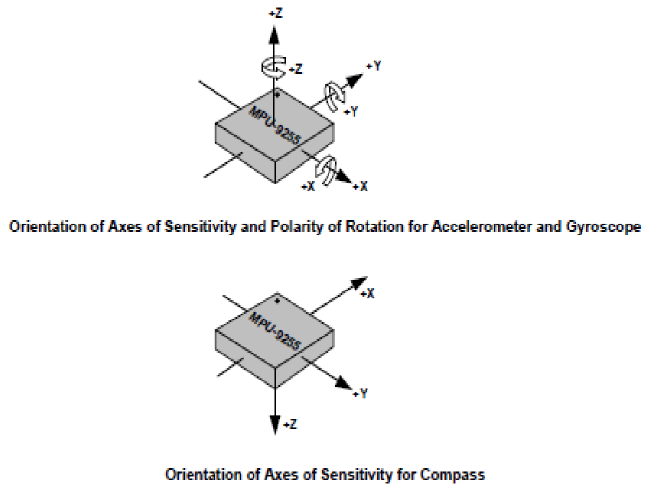


Figure 5. Orientation of Axes for MPU 9255

The flex sensor has the following features [23]:

- Angle displacement measurement;
- Bends and flexes physically with motion device;
- Flat resistances: 25 kiloohms;
- Bend Resistances Range: 45 k to 125 kiloohms.

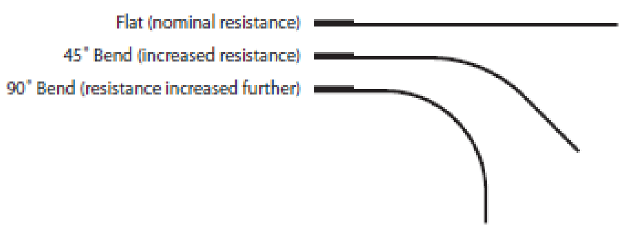


Figure 6. The operating principle of the flex sensor

## IV. Command and Control Algorithm of the Haptic Device

The block diagram and the connection between the main components of this device are shown in Fig. 7.

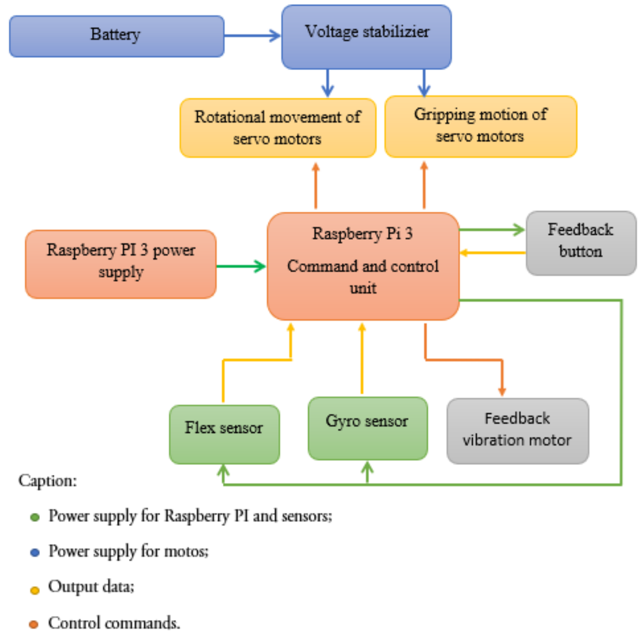


Figure 7. Overview of the various system components, connections and signals

The algorithm has been developed in Python 3.7 and contains functions that help define the initial data, detect the movement of the user's hand, calculate the angle of rotation of the hand, calculate the palm angle of flexion, determine the movement parameters of the servo motors but also integrate into a system. Fig. 8 illustrates the sequence of accessing the main functions of the algorithm:

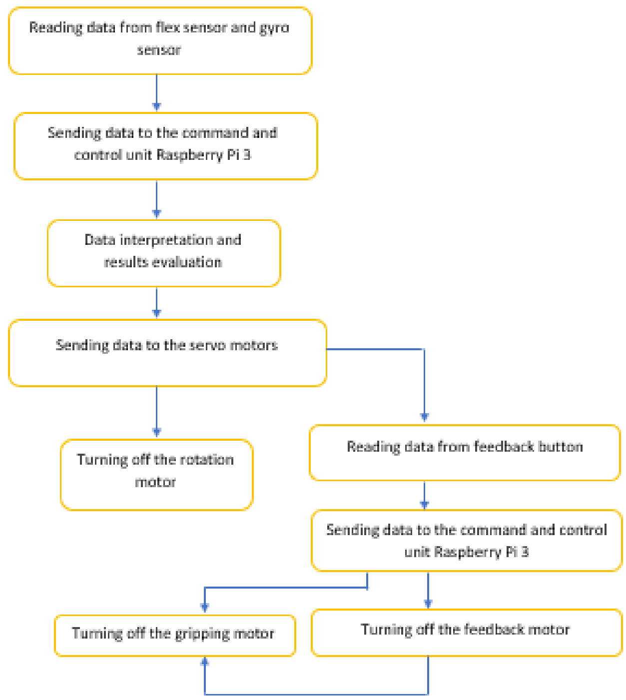


Figure 8. The main functions of the algorithm

Based on the functional flow of the software component, the algorithm integrating all modules was developed by the following steps:

1. In order to read the data provided by the sensors, their processing, their transmission by SPI or I2C, we have inserted the necessary libraries within the algorithm developed in Python.

```
import spidev
import time
import os
import RPi.GPIO as GPIO
from time import sleep
import smbus
import math
```
Figure 9. Necessary libraries

2. In the Python programming language, after inserting the libraries, you need to define the pins you use.

```
GPIO.setmode(GPIO.BOARD)
GPIO.setwarnings(False)
buton = 40
vibration = 37
GPIO.setup(7, GPIO.OUT)
pwm2=GPIO.PWM(7, 30)
pwm2.start(0)
GPIO.setup(buton, GPIO.IN, pull_up_down=GPIO.PUD_UP)
GPIO.setup(vibration, GPIO.OUT)
GPIO.output(vibration, GPIO.LOW)
GPIO.setup(11, GPIO.OUT)
pwm=GPIO.PWM(11, 50)
pwm.start(0)
```
Figure 10. Define ports and set their functionality

3. Considering that the sensor that determines hand flexion is analog and the Raspberry Pi 3 control and control unit does not have an analogue-to-digital converter in its architecture, we used ADC 0832 to make communication between the sensor and the Raspberry Pi.

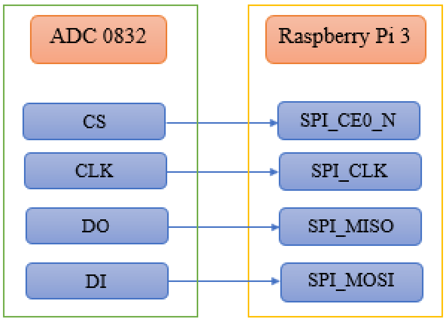


Figure 11. ADC0832-Raspberry Pi 3 connection

4. Reading the values from the MPU 9255 sensor is done via the I2C serial interface. The I2C bus is a serial master-slave bus for long distance communications with another device. The data transmission is synchronous on two bidirectional lines: serial data line (SDA) and serial clock line (SCL). The bus transmission speed ranges from 100 kbps for bidirectional communication in standard mode to 5 Mbps for unidirectional "ultrafast" communication [24].

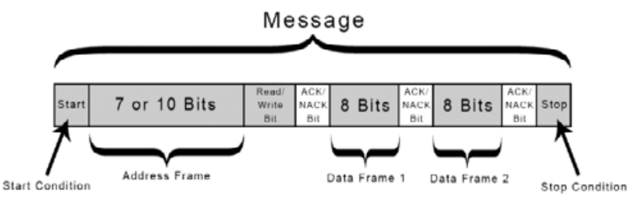


Figure 12. I2C protocol

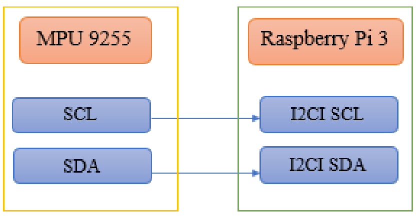


Figure 13. MPU 9255- Raspberry Pi 3 connection

5. The next step in developing the haptic control algorithm is to define the function for setting the working angle of the servo motors and setting the value of the servo motors.

```
def SetAngle(angle, pinuri, aaa):
    duty= angle / 18 + 2
    GPIO.output(pinuri, True)
    aaa.ChangeDutyCycle(duty)
    sleep(1)
    GPIO.output(pinuri, False)
    aaa.ChangeDutyCycle(0)
```
Figure 14. Setting the working angle of the servo motors

6. The following operations are performed in the "while" loop:
- Reading data from the flex sensor;
- Setting input data for the gripper servo motor;
- Reading data from the feedback button;
- Vibration motor control;
- Reading data from gyro sensor;
- Setting input data for the rotation servo motor.

## V. System Capacity Analysis

The algorithm for controlling the haptic device determines its movement based on readings from the sensors as follows:

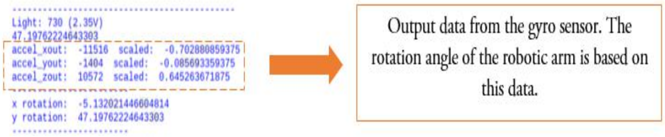


Figure 15. Output data from the gyro sensor

1. The user's hand is horizontal => The gripper of the haptic device is in horizontal position.

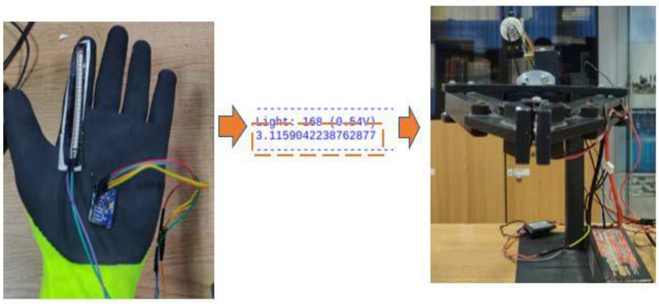


Figure 16. The gripper in horizontal position

2. The user's hand is vertical at the 90° angle => The gripper of the haptic device is in upright position at an angle of 90° to the horizontal.

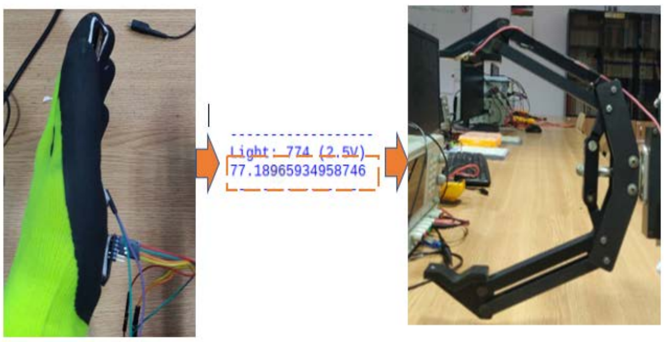


Figure 17. The gripper in vertical position

 Digital Object Identifier 10.32754/JMT.2019.1.02

3. Flex sensor is stretched => The gripper will open.


Figure 18. The gripper is open

4. Flex sensor is flexed => The gripper will be closed.

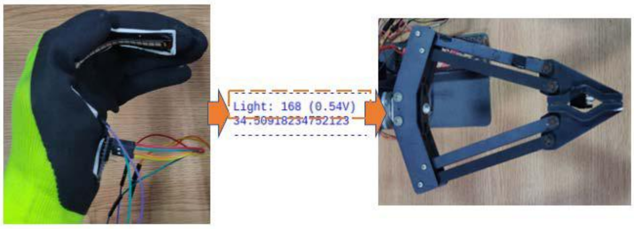


Figure 19. The gripper is closed

## VI. The Limits of the Haptic Device

### A. Mechanical problems

From the point of view of achieving the haptic gearing, we found that the servo motors had the necessary features to move the robotic arm. The arm rotation mechanism has proven to be insufficient to use a single wheel attached to the servo motor and a transmission belt for precise rotation and accurate manipulation of an object based on the information transmitted from the sensor module attached to the human factor.

### B. Sensor problems

The flex sensor used to determine the degree of flexion of the user's hand was not a high-quality sensor, affecting the processing and transmission of data from the sensor to the control module.

### C. Algorithm problems

The haptic control algorithm methodology aims to validate the proposed solution to initiate this project and creates new research directions to bring this project to an operational level. The front system can only be used under certain conditions, without precise control and accurate manipulation of objects. Moreover, using the control and command unit Raspberry Pi, the number of sensors and servo motors used in the haptic device control is constrained. In terms of acquiring data from sensors and transmitting them to control modules, the algorithm can be simplified by using treads.

## VII. Conclusion and Future Challenges

Research in the field of robotics and applications of this type may lead to the development of robotic systems used both in military and civilian fields that can substitute the presence of human factor in the field where it is required to act and allow remote user control. This involves several stages in the development and management of the project, some of which have been pursued in the present project. This haptic device confirms the integration of a sensor system and a mechanical drive confirming the possibility of a real-time remote manipulation of objects by a user with a glove. Moreover, the integrated system can be attached to a mobile platform in order to be able to easily move and perform the related tasks. To test the device in a varied environment, such as attached to a mobile platform, it would be necessary to improve both the sensory module and the algorithm.

In order to continue the project from the demonstrator phase to the prototype, the sensor system can be improved by using several higher quality sensors. Moreover, the use of servo motors smaller than current ones would help reduce the size of the entire device, which would lead to its use in several situations. Making a bluetooth or wireless transmission between the touchscreen module and the haptic device to create a safe distance from the remote control would be a new direction of development that would add to this project. Multiple directions of development arise from the wishes of device utility on a large scale, but the continuation of the project is also strictly dependent on the specific tasks that should be fulfilled.

## References

[1] R. Edmondson, K. Light, A. Bodenhamer, P. Besscher, L. Wilkinson, "Enhanced operator perception through 3D vision and haptic feedback", Proc. SPIE 8387, Unmanned Systems Technology XIV, 838711, May 1, 2012.

[2] "JAUS Service Interface Definition Language" Society of Automotive Engineers, Revision 1.0, 2008.

[3] UGV Interoperability Profile (IOP) – "Overarching Profile JAUS Profiling Rules, Robotic Systems", Joint Project Office, Version 0, 2011.

[4] T. Galluzo and D. Kent, "The OpenJAUS Approach to Designing and Implementing the new SAE JAUS Standards", AUVSI Unmanned Systems Conference, 2010.

*[5]* A. Fisch, C. Mavroidis, Y. Bar-Cohen, J. Meli-Huber, *Haptic Devices for Virtual Reality, Telepresence and Human-Assistive Robotics, Department of Mechanical and Aerospace Engineering, Rutgers University*, The State University of New Jersey.

[6] M. S. Johannes, R. S. Arminger, M. J. Zeher, *Dexterous Robotic Telemanipulation with Haptic Feedback, Johns Hopkins APL Technical Digest*, vol. 31, no. 4, 2013.

[7] J. G. S. Demers, J. M. A. Boelen, I. P. W. Sinclair, "Freedom 6S Feedback Hand Controller", MPB Tehnologies Inc. 1998. doi: 10.1145/274497.274515

[8] S. Grange, F. Conti, P. Herlmer, P. Rouiller, C. Baur, *The Delta Haptic Device*, Institut de Systemes *Robotiques*, Ecole Polytechnique Federale de Lausanne, 2000.

[9] "How do you choose a haptic device?", MPB Technologies Inc. doi: 10.1007/s10916-016-0459-8

[10] M. Benali-Khoudja, M. Hafez, J.M. Alexandre, A. Kheddar, *Tactile interfaces: a state-of-the-art survey*, Laboratoire Systemes Complexes - University of Evry.

[11] A. Kron, G. Schmidt, B. Petzlod, M.I. Zah, P. Hinterseer, E. Steinbach, *Disposal of Explosive Ordnances by Use of a Bimanual Haptic Telepresence System*, Institute of Automatic Control Engineering (LSR), Technische Universitat Munchen.

[12] A. M. Okamura, "Introduction to haptics", Stanford University, 2015.

[13] C. Pachierotti, S. Sinclair, M. Solazzi, A. Frisoli, V. Hayward, D. Prattichizzo, "Wearable Haptic Systems for the Fingertip and the Hand: Taxonomy, Review and Perspectives", IEEE vol. 10, 2017. doi: 10.1109/TOH.2017.2689006

[14] O. Tavsel, *Mechatronic Design of an Explosive Ordnance Disposal Robot*, Izmir Institute of Technology, 2005.

[15] E. W. Tunstel JR., K. C. Wolfe, M. D. M. Kutzer, M. S. Johannes, "Recent Enhancements to Mobile Bimanual Robotic Teleoperation with Insight Toward Improving Operator Control", Johns Hopkins APL Technical Digest, vol. 32, no. 3, 2013.

[16] J. Jurmain, "Overview of Haptic Science and Technology with Applications to Telemanipulation", Dartmouth College, 2006.

[17] M. A. Srinivasan, *What is Haptics?*, Laboratory for Human and Machine Haptics: The Touch Lab, Massachusetts Institute of Technology.

[18] J. J. Berkley, *Haptic Devices*, Mimic Technologies Inc., 2003.
[19] D. Talabă, A. Amditis, *Tools and Methods Based on Virtual Reality, Springer Science+ Business Media*, 2008. doi: 10.1155/2011/295074
[20] J. L. Thompson, *Redesigning the Human-Robot Interface: Intuitive Teleoperation of Anthropomorphic Robots*, The University of Texas at Austin, 2014.
[21] *ADC-0832-N 8-Bit Serial I/O A/D Converters with Multiplexer Options*, Texas Instruments, March 2013.
[22] *Flex Sensor*, SpectraSymbol.com.
[23] *MPU 9255 Product Specification*, InvenSenses Inc.
[24] https://eltechs.com
[25] https://en.wikipedia.org/wiki/Raspbian
[26] https://en.wikipedia.org/wiki/Python_(programming_language)