# Touchless Control of a Stationary Manipulator

Vlastimil Slany, Jan Marecek, Eva Krcalova,
Ivan Beloev, Vladimir Krocko, Miroslav Pristavka

***Abstract:*** *This article describes the creation of a non-contact control of a stationary manipulator. The Kinect device from Microsoft is used to capture and evaluate the video image. C ++ and C # were selected as the main programming languages. The C # programming language was used to create a Graphical User Interface (GUI). Microsoft offers direct support for the developers in the SDK (Software Development Kit). As a stationary manipulator, was selected the KATANA HD6M180 from the Swiss company Neuronics HD. This is a manipulator that can be controlled using the Katana Native Interface (KNI) library. This is a C ++ written library. It is a manipulator with 5 degrees of freedom. In this article is also described a GUI design, including a proposal for the individual control movements. Part of this article is also the design and creation of a control library that takes care of the communication between the created control software and the stationary manipulator. This library is written in C ++ programming language.*

***Keywords****: Kinect, KatanaHD300, KNI , automatization, touchless control*

## INTRODUCTION

The aim was to design and implement the Katana stationary manipulator trajectory (HD300s) through body movements. The Kinect device from Microsoft is used to capture motion. It has also been necessary to explore and select the appropriate programming language in terms of compatibility and support for individual devices. Based on the chosen programming language, a suitable SDK library (Software Development Kit) was chosen to enable us to communicate with Kinect device. It was also necessary to devise and choose a suitable control system, where we assign individual movements of human body to controlling each of the engines. An integral part of this work is also the Graphic User Interface (GUI) for communication with the user with respect to the selected programming language. Here, it is necessary to convert these data into a format that is accepted by the stationary manipulator Katana [1].

## MATERIAL AND METHODS

### Used hardware Kinect

The Microsoft Kinect sensor was considered a breakthrough in user interaction and controlled device, whether it is a personal PC or an Xbox 360/ Xbox One game console for which it was originally developed, under the project name Project Natal.

It is a motion control system that records user movement and translates it into control commands. So the user does not need any driver or controlling device and everything is done through the NUI (Natural User Interface). At the time of the release of this device, Microsoft was the first to come with a combination of 3D capture, facial and voice recognition. And this combination of hardware and software has enabled control without any peripherals (such as buttons or controls). All you need to do is stand in front of the Kinect device and use your body and natural moves or voice or gestures [2]

The accuracy decreases with the increasing distance of the object from our sensor. For better illustration, Figure 1 shows maximum and minimum possible distances while maintaining accuracy. It is also necessary to take into account whether the sensor works in the mode of the standard or decreased distance. The mode of the decreased distance was added at newer models, which were designed primarily for PC. For this reason, Kinect can be used in many industries [3].

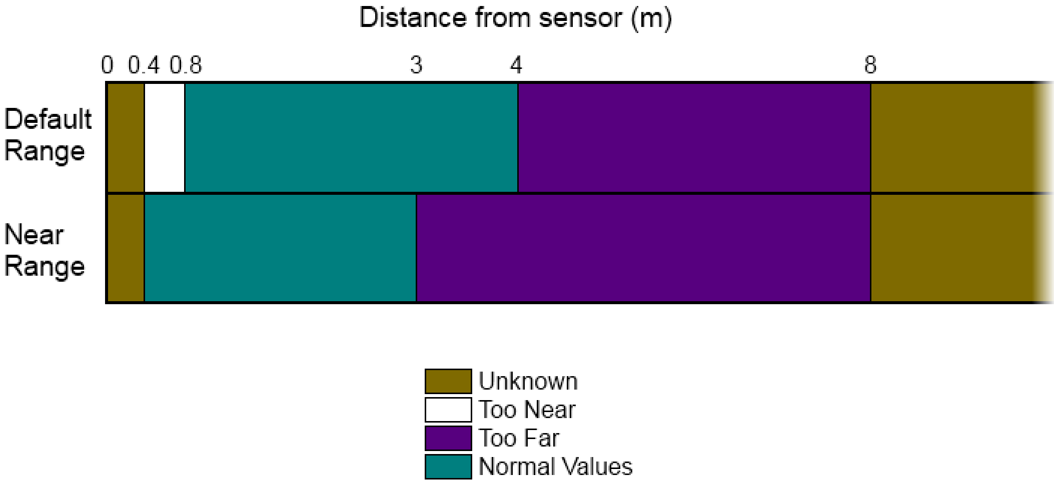


**Fig. 1** Kinect sensor distance – taken from www.msdn.microsoft.com

**Katana HD300s**

It is a stationary manipulator based on the model series HD300 of the Swiss company Neuronics AG. Due to its size and carrying capacity (500 g), it is primarily intended to be handled with small and light objects in practice. One of the typical tasks that this manipulator can perform on production lines is "pick and place". There is also the possibility to extend the camera-scrambling manipulator and to subsequently evaluate the input. This is especially useful on lines where defective products are being checked or automated stacking of items [4].

The following coordinate systems are defined in the Katana 6M180 manipulator.

- $K_w$ - World coordinate system
- $K_{tool}$ - Coordinate system of the tool
- $K_B$ - Base coordinate system – pedestal

Figure 2 shows all these systems and their mutual relations.

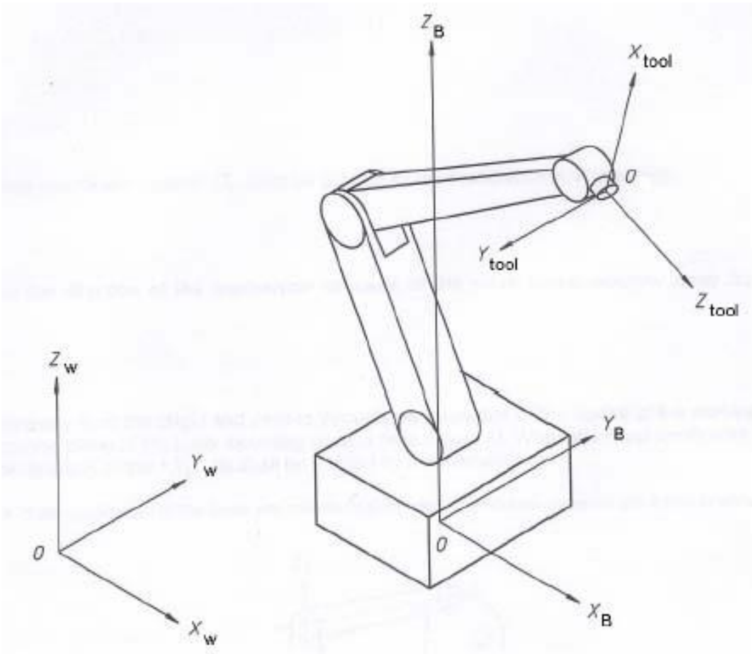


**Fig. 2** All coordinate systems – taken from Neuronics AG

On the figure 3 we can see the Katana 6M180 with the designation of individual motors.

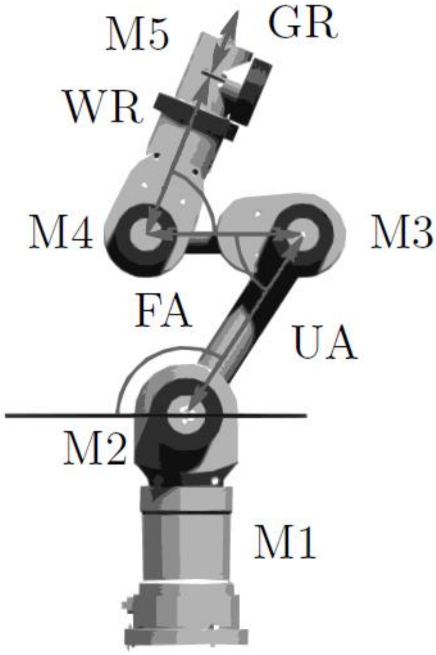


**Fig. 3** Katana 6M180 – taken from Neuronics AG

**KNI Library**

It is a software library that is designed for the development of control software for Katana robots. This library is created in the C ++ programming language. It is therefore divided into classes and structures.

This library provides easy methods for controlling a stationary manipulator Katana without starting basic protocols. As can be seen in Figure 4, it consists of the Communicatrion Device Layer (CDL), the Communication Protocol Layer (CPL), and the Katana Robot Model Layer. (Neuronics AG, 2006c)

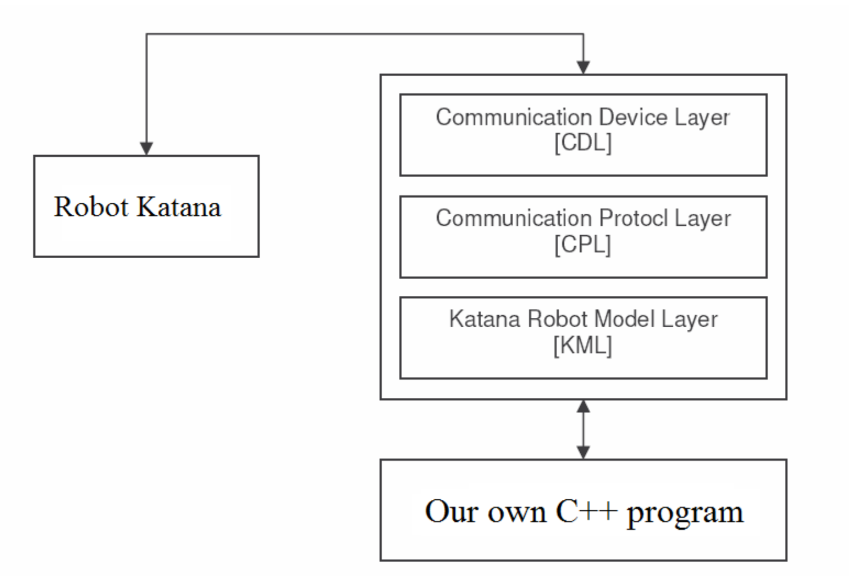


**Fig. 4** Structure of the KNI library – taken and edited from Neuronics AG

## RESULTS AND DISCUSSION

This work is based on relative control. The concept of relative control is based on the comparison of two skeleton images. It is therefore necessary to save the image of the skeleton, wait for the new skeleton image, to make a comparison between these images based on the obtained and calculated data. This solution is simpler than absolute control in our case, but it also has its own pitfalls. The first is the need to calibrate our user. Each person has different body proportions, so the calculated values can vary considerably among individual people. It is also necessary to carefully treat the movement limits so that these limits cannot be exceeded. Calibration of the user is performed using the start gesture. The other option is to use the inverse kinematics inverse role to calculate the position of the manipulator with five degrees of freedom based on scanned values [5].

This means that until the user has made a predetermined start gesture from which it is possible to read the necessary data for controlling the stationary manipulator, he will not be allowed to manipulate with manipulator. The main advantage of this approach is to temporarily interrupt scanning, positioning the hand to the desired starting position, and then continue driving manipulator. Therefore, unlike the absolute control, it is not necessary to navigate the user to the previous end position, but the user can determine his start position himself. Thus, the main disadvantage of the previous steering system is that we need to change the position of one engine without affecting the other engine. Here, we just can stop the scanning, put our hand to a different starting position from which it will be possible continue to control the manipulator. We have 5 engines, which can be controlled. For example, on Figure 5, we can see minimal and maximal position for controlling engine 1 [6].

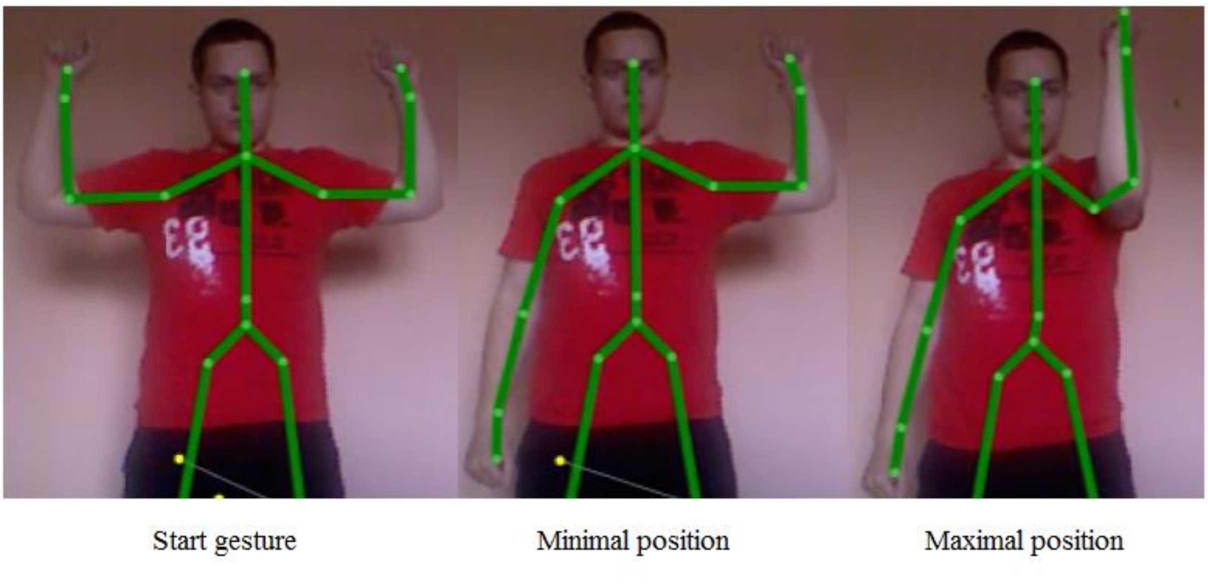


**Fig. 5** The pattern of start gesture, minimal and maximal position for engine 1

Convert output scanned data from Kinect device to stationary input data of the manipulator depends largely on the option of reading data. In this work is described option, which is called (in this paper) as relative trajectory planning. It is necessary to compare the two captured skeleton records. For this purpose, the image of the skeleton according to which the last evaluation was performed is stored in memory. After that, a new image is taken, which is compared to the last recorded image.

Based on this comparison, it is possible to determine which joint has the largest change in its position compared to the last image. It was also necessary to choose the appropriate interval in which scanning and skeleton comparison will be performed. After testing the individual positions, the 200 ms interval was selected. The choice of this interval was largely influenced by the technical parameters of the stationary manipulator, when it is not possible to send the control information in the interval less than 200 ms. Commands that are sent in less than 200 ms are ignored and will not be executed. As mentioned above, the Katana stationary manipulator has 5 engines that can be controlled. Each of these engines has a different range in which we can move. In table 1 we can see the ranges of encoders of individual motors.

The Kinect returns the numerical distance of the knuckles from the point "0" as the output, which in our case is the Kinect device position itself. This distance is reported in meters with an accuracy of 1 cm. The output is a decimal number with accuracy of 2 decimal places. In order to be able to convert these input data to the input data of the stationary manipulator, motions have been determined to uniquely identify the engine to be controlled.

Then a stationary step of 1000 encoder units is added to the particular encoder in the evaluated direction. If the encoder value is equal to the maximum or minimum amount, and the stationary manipulator should exceed this value, the encoder value is set to either maximum or minimum, and cannot therefore exceed this limit. However, if any program bug could occur to circumvent this security algorithm, this situation is treated in library (KNI) of the stationary manipulator Katana. There should therefore be no damage to the stationary manipulator due to exceeding the limits of the individual encoders.

## ACKNOWLEDGEMENT

The research was financially supported by the AF – IGA IP 2016/44 – Návrh procesu monitoringu dat v exrémních prostředích a následné automatizované zpracování naměřených dat.

## REFERENCES

[1] Kinect for Windows SDK 2.0. [Online] Available: https://www.microsoft.com/en-us/download/details.aspx?id=44561

[2] Evaluation of Pointing Strategies for Microsoft Kinect Sensor Device [online]. Available: http://diuf.unifr.ch/main/diva/sites/diuf.unifr.ch.main.diva/files/REPORT_DARIA.pdf

[3] Napoli, A., Glass, S., Ward, C., Tucker, C., Obeid, I., (2017) Performance analysis of a generalized motion capture system using microsoft kinect 2.0Biomedical Signal Processing and Control, 38, pp. 265-280. DOI: 10.1016/j.bspc.2017.06.006

[4] Kanellakis, C., Kyritsis, G., Tsilomitrou, O., Manesis, S., (2015) A low-cost stereoscopic μp-based vision system for industrial light objects grasping. 23rd Mediterranean Conference on Control and Automation, MED 2015 – Conference. Proceedings, art. no. 7158837, pp. 759-765. DOI: 10.1109/MED.2015.7158837

[5] Korenko, M., Beloev, H., Kaplík, P., (2013) Quality control, using PPAP method. *scientific monograph.* 1st ed., Ruse: University of Ruse Angel Kanchev, 2013. pp. 138. ISBN 978-619-7071-12-2.

[6] Hassanov, K., Erbolat, Y., (2014) Inertial motion capture based reference trajectory generation for a mobile manipulator. Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction-HRI'14 [online]. 2014 New York, New York, USA: ACM Press: 202-203. DOI: 10.1145/2559636.2559812. ISBN 9781450326582. Availabe: http://dl.acm.org/citation.cfm?doid=2559636.2559812.

[7] NEURONICS KATANA. (2011) Katana exactech [online]. Available: http://katana.exactec.com/

[8] Prístavka, M., Beloev, H., Kročko, V., (2014) Quality Control in Production Processes. 1st ed. Ruse: University of Ruse Angel Kanchev, 2014. ISBN 978-619-7071-62-7.

**CONTACTS**

Vlastimil Slaný, Mendel University in Brno, Department of Agricultural, Food and Environmental Engineering, e-mail: xslany@mendelu.cz

Jan Mareček, Mendel University in Brno, Department of Agricultural, Food and Environmental Engineering, e-mail: jan.marecek@mendelu.cz

Eva Krčálová, Mendel University in Brno, Department of Agricultural, Food and Environmental Engineering, e-mail: xkrcalo1@node.mendelu.cz

Ivan Beloev, Department of Transport, Faculty of Transport, University of Ruse, 8, Studentska str., 7017 Ruse, Bulgaria, e-mail: ibeloev@uni-ruse.bg

Vladimir Krocko Department of Quality and Engineering Technologies, Faculty of Engineering, Slovak University of Agriculture in Nitra, Tr. A. Hlinku 2, 949 76 Nitra, Slovak republic, e-mail: vladimir.krocko@uniag.sk

Miroslav Prístavka, Department of Quality and Engineering technologies, Faculty of Engineering, Slovak University of Agriculture in Nitra, Tr. A. Hlinku 2, 949 76 Nitra, Slovakia, miroslav.pristavka@uniag.sk