Prototype of natural user interface applied to a robotic arm for medical attention preventing nosocomial infections in healthcare personnel

Prototipo de interfaz de usuario natural aplicado a un brazo robótico para atención médica previniendo infecciones nosocomiales en personal sanitario

SERRANO-RAMÍREZ, Tomás†*, GUTIÉRREZ-LEÓN, Diana Guadalupe, MANDUJANO-NAVA, Arturo and SÁMANO-FLORES, Yosafat Jetsemaní

Universidad Politécnica de Guanajuato, Automotive Engineering, Mexico.

ID 1st Autor: *Tomás, Serrano-Ramírez* / **ORC ID**: 0000-0001-6118-3830, **Researcher ID Thomson:** G-6039-2018, **CVU CONACYT ID:**493323

ID 1st Coautor: *Diana Guadalupe, Gutiérrez-León* / **ORC ID**: 0000-0001-5051-880X, **Researcher ID Thomson:** G-6035-2018, **CVU CONACYT ID:** 443892

ID 2nd Coautor: *Arturo, Mandujano-Nava* / **ORC ID**: 0000-0003-2022-4397, **CVU CONACYT ID**: 270254

ID 3rd Coautor: *Yosafat Jetsemaní, Sámano-Flores /* **ORC ID**: 0000-0003-4173-6236, **CVU CONACYT ID**: 444850

DOI: 10.35429/JTI.2020.20.7.19.25 Received July 20, 2020; Accepted December 30, 2020

Abstract

In this work, an experimental prototype of natural user interface based on Kinect for the tele-operation of a robotic arm, was developed as a technological support to contribute in facing the global crisis of health generated by COVID-19, a dangerous and highly transmissible virus. Until now, it is causing thousands of deaths and growing of contagion rates around the world, involving a serious situation for healthcare personnel that works in hospitals attending infected population. This system was proposed with the aim to control a robotic arm for medical purposes to achieve a quality medical care to COVID-19 patients without risk implications for healthcare workers associated to nosocomial infections due to direct contact with infected patients, contaminated medical equipment, surgical objects and surfaces. The developed prototype is able of being manipulated in real time requiring neither physical controller nor any contact device to carry out its functions, but simply motion or gesture from the user's arm. It can be applied in areas such as tele-operation, tele-rehabilitation, telehealth nursing, assistive and therapeutic robotic devices, elderly care which are the last tendency in medicine at this time. Kinect V2, the Software Development kit SDK 2.0, Microsoft visual C# and Arduino were used for this purpose.

Natural-user-interface, Robotic-arm, COVID-19

Resumen

En este trabajo, un prototipo de interfaz natural de usuario basado en Kinect para la tele-operación de un brazo robótico fue desarrollado como soporte tecnológico para contribuir a hacer frente a la crisis global de salud generada por COVID-19, un virus peligroso y altamente transmisible. Hasta ahora, éste ha causado miles de muertes y altas tasas de contagios alrededor del mundo, involucrando una seria situación para el personal médico que trabaja en hospitales brindando la atención a las personas infectadas. Este sistema fue propuesto con el objetivo de controlar un brazo robótico para fines médicos para lograr una atención médica de calidad para los pacientes de COVID-19, sin que ello implique un riesgo para sí mismos, ya sea teniendo contacto directo con pacientes infectados, objetos quirúrgicos y superficies contaminadas. El prototipo desarrollado es manipulado en tiempo real sin requerir un control físico u otro dispositivo de contacto para realizar sus funciones, sólo movimiento y gesticulación del propio brazo del usuario. Esto puede ser aplicado en áreas tales como la tele-rehabilitación, cuidado médico a distancia, dispositivos robóticos terapéuticos y de asistencia, lo cual constituye las últimas tendencias en Medicina. Kinect V2, Software Development kit SDK 2.0, Microsoft visual C# y Arduino fueron usados para este propósito.

Interfaz natural de usuario, brazo robótico, COVID-19

Citation: SERRANO-RAMÍREZ, Tomás, GUTIÉRREZ-LEÓN, Diana Guadalupe, MANDUJANO-NAVA, Arturo and SÁMANO-FLORES, Yosafat Jetsemaní. Prototype of natural user interface applied to a robotic arm for medical attention preventing nosocomial infections in healthcare personnel. Journal of Technology and Innovation. 2020. 7-21:19-25.

* Correspondence to Author (E-Mail: tserrano@upgto.edu.mx)
† Researcher contributing as first author.

Introduction

Worldwide, COVID-19 has been categorized as a novel, rising and highly infectious disease (Chang, Joob and Wiwanitkit, Hsia, 2020), which it is responsible to cause until now (May 8th, 2020), 3,767,744 confirmed cases and 259,593 deaths (WHOa, 2020).

Healthcare personnel is heavily exposed to infection risk during its working sessions due to direct contact with infected people and contaminated objects including equipment, medical instruments and supplies.

Natural user interface is a technology that can be used to control devices without establish a direct contact, specifically by gestures and human body motion that is correlated with the interface reaction (Sommerer *et al*., 2005).

The latest trends in natural user interfaces involve the control of artificial limbs using the own user's extremity movements and gestures. Natural user interfaces have several advantages including (Blake, 2012):

Instant expertise: The user takes advantage of the existing skills or abilities that most of the people domain such as the movement of his own body in order to interact with technological devices. It gives in the majority of cases an instant level of expertise letting the user to apply his natural abilities to control the device.

Progressive learning: Natural user interfaces imitate the way people learn physical skills, allowing them to start with basic tasks and move on to something more advanced step by step, in a steady increment. That is not the case with other interfaces that require a huge training by the user in order to begin the interaction with the device. Direct interaction: Natural user interface imitates the user's interaction with the physical world by having a direct correlation between user action and the natural user interface reaction. It means that the interface actions happen at the same time as user actions or that the motions of elements on the interface follow the motions of the user.

Low cognitive load: The mental effort to manipulate a natural user interface is minimum, due to the user takes advantage of his basic skills or abilities, such as the movement of body, letting him focuses on achieving a task.

As is shown, natural user interfaces represent the future tendency in the interaction among technological devices and human beings.

Kinect V2 is able to facilitate the interaction between human and robots, supporting non- expert users to control them with minimal training, in a more intuitive and natural way. Kinect V2, the Software Development kit SDK 2.0, Microsoft visual C# and Arduino are used for this purpose. Kinect V2 is an accessible but powerful motion sensing input device, capable to track the human body joints in 3D space and recognize hand gestures in real time (Microsoft b, 2014). The joint positions are displayed on screen (skeleton-tracking), converted into angles and sent to the microcontroller board (Arduino), in order to control the arm's position (servomotors). The hands gestures are used to open and close the hand grip or even turn on/off the interface.

By the other hand, the Software Development Kit (SDK 2.0) provided by Microsoft, is free and there will be no fees for runtime licenses of commercial applications developed with it. The interface was written in (C#/WPF), based on a free example provided in SDK2.0 and is capable to track 25 joints per person at 30 frames per second.

Robotic arms are part of this evolution, and many of them which have been successfully used with more conservative interfaces are migrating to the use of natural user interfaces, in areas such as: surgical robots, tele-operation, tele-rehabilitation, assistive and therapeutic devices, human augmentation, tele-skill transfer and soft robotics.

Some representative examples of the latest trends in the research of natural interfaces and robotic arms applied in medicine are as follows: a gesture control for the Da Vinci Robot developed by Laboratory for Computational Sensing and Robotics (LCSR) (Wang et al., 2012) and a soft exoskeleton jacket for human motion interaction carried out by The Biological Systems Engineering Lab at the University of Hiroshima (Vega and Kurita, 2018).

Then, the aim of this work is developing a natural user interface for controlling a robotic arm in real time, without the requirement of a physical controller or any contact device, but simply by the motion or gesture from the user’s arm. This technological application could help the medical staff to face the elevated risk of infection, generated by COVID-19, giving medical attention with high quality, without establish direct contact either infected patients or contaminated objects, in which could be included, the physical controller in a conventional robotic arm. Additionally, the implementation of advices for public issued by WHOb (2020) such as maintaining distance by yourself and others and, avoid touching contaminated surfaces with finger by healthcare personnel.

Design and development of prototype

The natural user interface designed and developed to control a robotic arm is explained below.

The proposed project consists of four hardware modules: 1) motion sensing input device (Kinect V2), 2) computer, 3) microcontroller (Arduino mega 2560) and 4) robotic arm, whose block diagram and prototype are presented in Figure 1 and Figure 2, respectively.

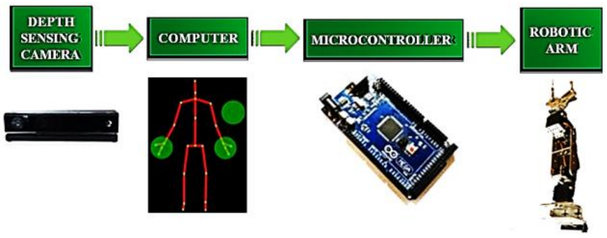


Figure 1 Hardware flow chart for the remote-controlled robotic arm

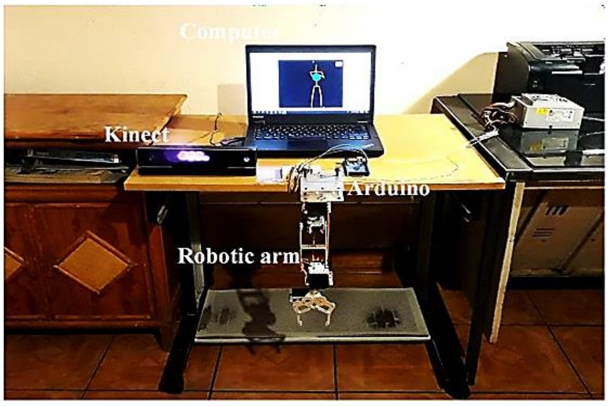


Figure 2 Remote controlled robotic arm and the real position of the hardware modules

Hardware modules are explained as follows:

Motion sensing input device (Kinect V2): the Kinect V2 sensor by Microsoft is used as an input device. It features an RGB camera and a depth sensor (Time of Flight) which provide full-body 3D motion capture at 30 frames per second, sending the data to the computer via USB 3.0.

Computer: it is notebook or a personal computer (PC) which stores and executes the designed natural user interface. The natural user interface processes the Kinect information sent via USB by artificial intelligence (Decision Forest Classifier and Mean shift), obtaining the position of the 25 specific body joints from the user at 30 fps. Natural user interface displays the user skeleton tracking and at the same time, sends via USB the respective angles and hand states to Arduino for controlling the robotic arm.

Microcontroller: once the angles or hand states are sent from the computer using the USB port to the microcontroller board (Arduino mega 2560), it receives them, and for each one, generates a PWM signal to position the servo-motor (belonging to the robotic arm) into a specific angle. The servo-motor angle depends on the magnitude of the received angle from Kinect.

Robotic arm: the prototype is an articulated robotic arm, that tries to emulate the human arm, under the wrist and uses servo-motors to mimic the joints motion. For this prototype, the we emulate the movement from the right shoulder and the open and closed right hand state; three servo-motors mg996r are used for this purpose (0 to 180° turning radius).

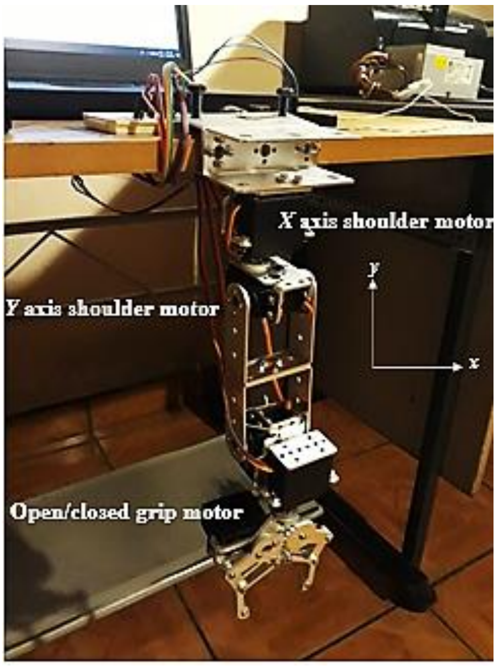
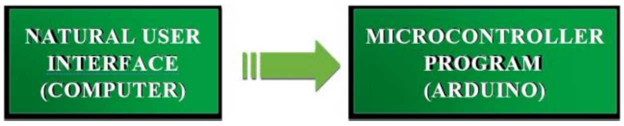


Figure 3 The robotic arm used in this project

The proposed project consists of two software modules as is shown in Figure 4.


NATURAL USER INTERFACE (COMPUTER)

MICROCONTROLLER PROGRAM (ARDUINO)

Figure 4 Software flow chart for the remote-controlled robotic arm

Natural user interface

In this section the proposed natural interface to control the robotic arm is explained in detail. The following operative system and tools were used to develop the natural interface:

- Windows 10.
- Kinect for Windows Software.
- Development kit 2.0 (SDK 2.0).
- Visual Studio 2017.
- Microsoft visual C#.

The Kinect sensor was developed by Microsoft Corporation, which provides a Software Development Kit (SDK 2.0) that lend to programmers generate applications supporting corporal motion and voice recognition. SDK 2.0 carry out its functions with the operative systems Windows 8 and Windows 10 and it does not require fees for runtime licenses of commercial applications developed with it (Microsoft a,b, 2014).

We began the natural user interface development with a free basic skeleton-tracking sample code, provided by Microsoft SDK 2.0. This sample was written in (C#/WPF) and is capable to track 25 joints per person at 30 frames per second and identify open/closed hand gestures.

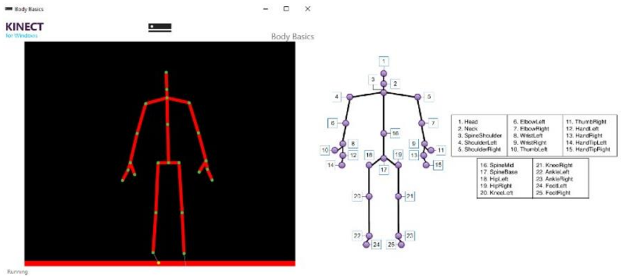


Figure 5 Skeleton-tracking sample code, provided by Microsoft SDK 2.0, which detects 25 body joints (Ahmed *et al*., 2015)

The free skeleton-tracking sample code, provided by Microsoft SDK 2.0 to demonstrate the Kinect capabilities, is used as a basis in the design and development of the Natural User Interface which controls the Robotic Arm. The first step is to calculate the angles from the user´s arm using the body Joints detected by Kinect. It is called forward kinematics and is explained in the next section.

Forward Kinematics

Forward kinematics require the application of kinematic equations corresponding to a robot, to calculate the location of the end- effector from specified values for the joint parameters (Al-Ammri and Ahmed, 2018). In this case the Joint parameters are the body joints detected from the Kinect sensor and the end effector is the robotic arm. The robotic arm must be controlled by the right arm, therefore, two body joints from the user arm (ShoulderRight and HandRight, Figure 5) are used to calculate the arm angles. In specific two angles are needed to control the shoulder movement from the robotic arm: the angle between the arm and the XZ plane, and the angle between the arm’s projection in XZ plane and Z axis (Figure 6 and considering ShoulderRight at origin.

These angles will be used to control the shoulder servomotors from the robotic arm (Figure 3).

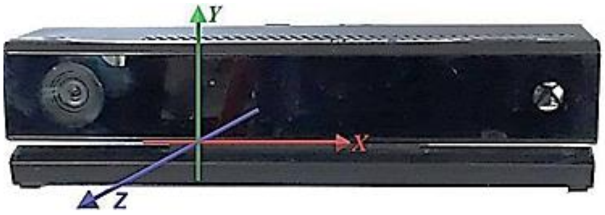


Figure 6 Kinect V2 coordinate system

These angles must be calculated using vector calculus following the next steps:

- Get the coordinates ($x1$, $y1$, $z1$) from ShoulderRight and ($x2$, $y2$, $z2$) from HandRight joints.

- Move the vector formed by the union of these two joints (pointing from ShoulderRight to HandRight) to the coordinate origin:

$$A^{\rightarrow} = (x2 - x1,\ y2 - y1,\ z2 - z1) \qquad (1)$$

- Calculate the vector magnitude an its projection in XZ plane magnitude:

$$[\vec{A}] = \frac{\sqrt{(X-X)^2+(y-y)^2+(z-z)^2}}{2 \quad 1 \quad 2 \quad 1 \quad 2 \quad 1} \qquad (2)$$

$$[\vec{P}] = \frac{\sqrt{(X_2-X_1)^2+(z_2-z_1)^2}}{} \qquad (3)$$

Calculate the angle α between the arm and the XZ plane and the angle β between the arm's projection in XZ plane and the z, applying the equations presented in Ec. 4 and Ec. 5

$$\alpha = sin^{-1} \left(\frac{\vec{A}_y}{\vec{}}\right) |A| \qquad (4)$$

$$\beta = sin^{-1} \left(\frac{\vec{A}_x}{|\vec{P}|}\right) \qquad (5)$$

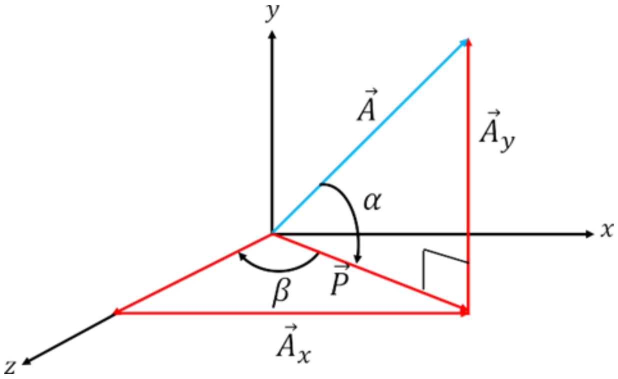


Figure 7 3D space interpretation for the robotic arm forward kinematics

Once the angles α and β are calculated, they are sent via USB to Arduino for controlling two servomotors belonging to the robotic arm.

Microcontroller program

The microcontroller board (Arduino) is programed to communicate with the natural user interface throughout USB port, receiving two angles (0 to 180°) from the user right-arm position and the right-hand state (open Close). These variables are converted to PWM (Pulse Code Modulation) and sent to the Arduino ports in order to control three servomotors mg996r. The servomotors control the arm movement and the grip from the robotic arm.

Results

Interactive Functionality. At this moment, the robotic arm can be controlled by the natural user interface but it lacks of interactivity, or a dialog between the user and Computer, wasting the powerful functionality that Kinect offers in this category. For this purpose, three hand gestures detected by the improved Kinect V2 will be used in order to communicate with de computer these gestures are: open, closed and lasso as shown in Figure 8. When Kinect detects some of these gestures, it sends to the computer Boolean variables that can be read with the appropriate command in SDK 2.0.

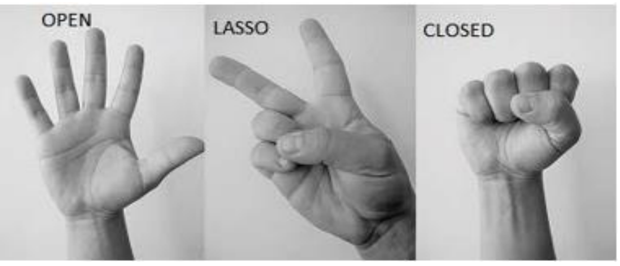


Figure 8 Three hand gestures detected by the Kinect V2 sensor
Source: (Arizpe, 2016)

We improve the natural user interface in order to turn on and off the device at distance, using hand gestures as can be seen in Figure 9. It avoids undesired and unsafe movements for the robotic hand, in the transition between these two states (on/off) and facilitating the user switching.

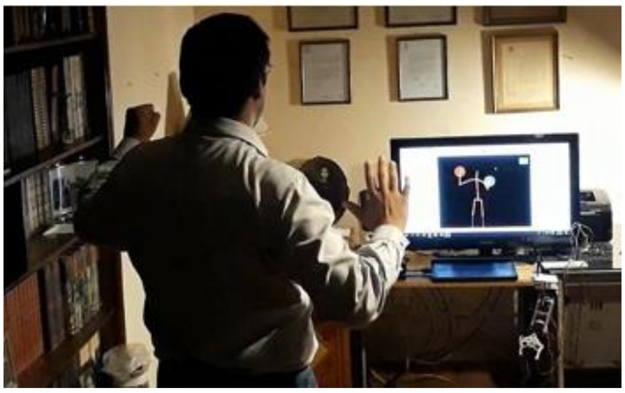


Figure 9 Natural user interface developed in this project

When a user is in front of the Kinect, the sensor only displays the skeletal tracking, but the robotic arm is turned off. When the user closes the left-hand, the user can activate the robotic hand guiding the right-hand position at screen over a green circle, once they match the device is activated. When the user wants to quit, he opens the left hand and guides the right-hand position over the green circle and the robotic hand is deactivated.

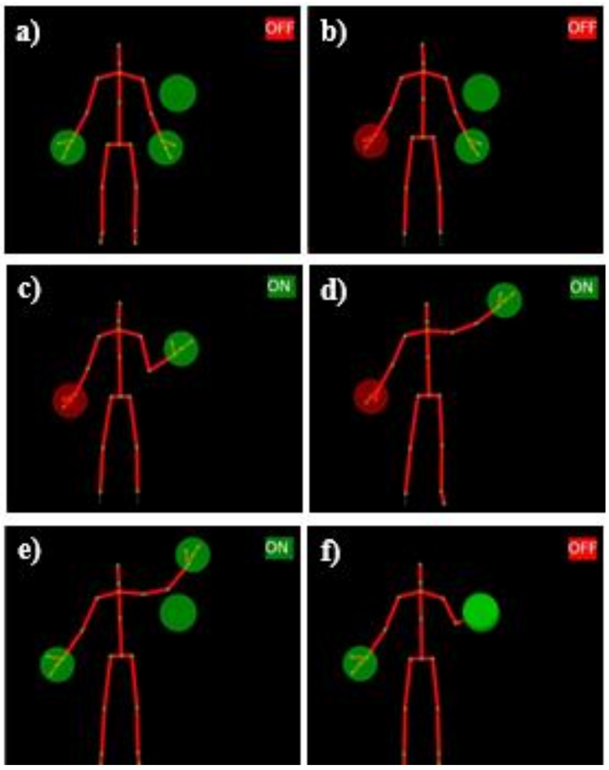


Figure 10 a) User is in front of the Kinect with both hands open, the sensor only displays the skeletal tracking, but the robotic arm is turned off. b) When the user closes the left-hand, c) the user can activate the robotic arm guiding the right-hand position at screen over a green circle, d) once they match the device is activated. e) When the user wants to quit, he opens the left hand and f) guides the right-hand position over the green circle and the robotic arm is deactivated

Conclusions

According to the objectives set in this project, a natural user interface for controlling a robotic arm in real time, without the requirement of a controller or any contact device, but simply by the motion or gesture from the user's arm was successfully designed. The fast response and easy operation in the control of a robotic arm, give our natural user interface important advantages over many proposed prototypes under the same budget. Our prototype can be easily escalated to a more complex natural user interface, which would let us to control a robotic arm with more degrees of freedom, fingers control and haptics, emulating the movement of a real arm more closely. Besides that, applying a remote control using a node to node communication in an Internet of Things (IoT) application, could transform this prototype in a tool for telemedicine, in areas such as tele-operation, tele-rehabilitation, telehealth nursing, assistive and therapeutic robotic devices, which are the last tendency in medicine at this time.

References

Al-Ammri, A. S., Ahmed, I. (2018) Control of omni-directional mobile robot motion. Al-Khwarizmi, Eng. J. 6(4), 1-9.

Ahmed, F. Paul, P. and Gavrilova, M. (2015) Kinect-Based Gait Recognition Using Sequence of the Most Relevant Joint Relative Angles, Journal of WSCG, 23(2), 147-156.

Blake, J. (2012) The natural user interface revolution. In Natural User Interfaces in .NET. Manning Publications.
Chang, D., Huiwen, X., Reabaza, A., Sharma, L., Dela Cruz, C. (2020) Protecting health-care workers form buclinical coronaviros infection. Correspondence 8-3, E13. Published: March 01, 2020.

Hsia, W. (2020) Emerging new coronavirus infection in Wuhan, China: situation in early 2020. Case Study Case Rep 2020; 10:8e9.

Joob, B., Wiwanitkit, V. (2020) COVID-19 in medical personnel: observation from Thailand. Journal of Hospital Infection 104(4).

Sommerer, C., Jain, L. C., Mignonneau, L. (2005). The art and science of interface and interaction design / (ed. Springer).

Wang, X.L., Stolka, P.J., Boctor, E., Hager, G.D., & Choti, M.A. (2012). The Kinect as an interventional tracking system. Medical Imaging: Image-Guided Procedures. Robotic Interventions and Modeling, 8316, 1-6.

Vega A. and Kurita, Y. (2018). A soft exoskeleton jacket for human motion interaction. The 36th Annual Conference of the Japanese Robotics Society, At Chubu University.

Microsoft a (2014) Download the updated Kinect for Windows SDK public preview. URL: https://blogs.windows.com/windowsdeveloper/2014/08/26/download-the-updated-kinect-for-windows-sdk-2-0-public-preview/ Updated: August 26, 2014.

Microsoft b (2014) Kinect for Windows SDK v1.8 URL:https://www.microsoft.com/en-us/download/details.aspx?id=44561 Updated: October 21, 2014.

World Health Organization, WHOa (2020) WHO Coronavirus disease (COVID-19) dashboard. https://covid19.who.int/ Updated: May 8, 2020

World Health Organization, WHOb (2020). Coronavirus disease (COVID-19) advice for the public. https://www.who.int/emergencies/diseases/novel-coronavirus-2019/advice-for-public
Updated: April 29, 2020.