International Journal of Inventive Engineering and Sciences (IJIES)
ISSN: 2319–9598, Volume-4 Issue-1, February 2016

Vector Multiplication Approach for Point of View Variations in a Mimicking Robotic Shoulder using Microsoft Kinect

Rollie Jay R. Ortega, Norm Allen T. Siron, Kevin T. Mapalad, Heidrick S. Emano,
Roselito E. Tolentino

Abstract*— *This study aimed to manipulate a robotic shoulder even while the user is varying its point of view by applying a vector multiplication and controlling it using a camera with depth sensor. The system acquired the motion of the user's arm using Kinect sensor. The position of the user's joints was obtained using the Kinect Skeletal Tracking of Kinect SDK. Through the use of Visual Studio, we used C# and create a program to acquire the values of the skeletal coordinates and that was used to calculate the vectors using Cross Product and then the angles using Dot Product of the Vector Multiplication. The angles obtained were sent to the microcontroller through serial communication and then converted to signals for the movement of servo motors of the robotic shoulder. The rotation of the servo motors was according to the angles given as input. The researchers concluded that the system is effective in acquiring the user's shoulder angle for the mimicking of robotic shoulder for different point of views. Likewise, the researchers considered that the user's actual shoulder angle is close to the robotic shoulder prototype angle.

Index Terms*—*Kinect, mimicking, robotic shoulder, vector multiplication.

I. INTRODUCTION

In recent years, the development of interaction between humans and robots has become a key research topic in the field of robotics. Robot mimicking is one of the important application for the field of robotics, where the robots are being controlled by human movements. Different methods and techniques have been develop and different sensor devices have been use to capture human motion to manipulate the movements of the robot. One of the most reliable sensor device is the Microsoft Kinect, a 3D camera sensor composed of a RGB camera and a depth sensor that is widely used in motion capture system because of its relative low price.

Various groups have attempted to create a solutions to accurately translate human motions in a robot using Kinect Sensor. Such works include a research work at HuT Labs which is a Kinect Based Gesture Controlled Robotic Arm that used a Coordinate Geometry to calculate angles is presented in [1]. Another work a Kinect-Based Humanoid Robotic Manipulator for Human Upper Limbs Movements Tracking which implemented an inverse kinematics and they used directional angels to get the angles needed to control their robot is presented in [2]. Another study only implemented a simple vector approach where they just get the angle between the two vectors that is facing the sensor to control the robot by using Kinect Sensor [3].

A lot of studies in the field of robotics used Kinect sensor in capturing the human motion [4-9]. But different problems occurred, one of these problems is when the user change its point of view, the acquired data became unreliable for their system. It means that the user must be parallel in the front of the sensor always to have an accurate acquisition of the angles needed. If the user changed its point of view, the angle acquired by the sensor will not be accurate.

In this paper, our main objective is to show that the robotic shoulder can mimic the movement of the human shoulder even the user is varying its point of views. In order to do that, we designed and implemented a new algorithm base on Vector Multiplication Approach. This algorithm used the concept of Euclidian Distance, Cross Product and Dot Product. The human shoulder movement was captured in real time by the used of Kinect as a sensor which is connected to the computer to be processed and analyses the motion of the user then sends the data to the microcontroller that generates signals for the movement of the robotic shoulder.

II. METHODS

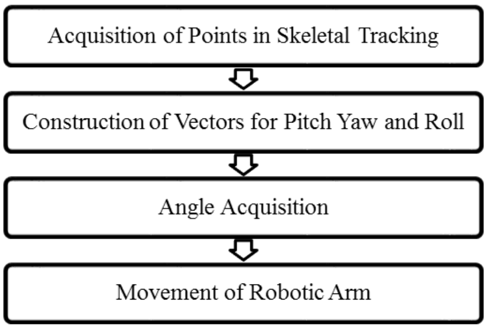


Fig. 1. System Block Diagram

Figure 1 shows how the system works. It consist of acquisition of points in Skeletal Tracking, Construction of Vectors, angle acquisition and the movement of the robotic arm. First, with the use of Skeletal Tracking of Microsoft Kinect's feature, the system acquires the 3D coordinates of the required joints.

Revised Version Manuscript Received on March 12, 2016.

Rollie Jay R. Ortega, Are Graduates of Polytechnic University of the Philippines Santa Rosa Campus, City of Santa Rosa Laguna, Philippines.

Norm Allen T. Siron, Are Graduates of Polytechnic University of the Philippines Santa Rosa Campus, City of Santa Rosa Laguna, Philippines.

Kevin T. Mapalad, Are Graduates of Polytechnic University of the Philippines Santa Rosa Campus, City of Santa Rosa Laguna, Philippines.

Heidrick S. Emano is Currently Studying in Polytechnic University of the Philippines Santa Rosa Campus, City of Santa Rosa Laguna, Philippines.

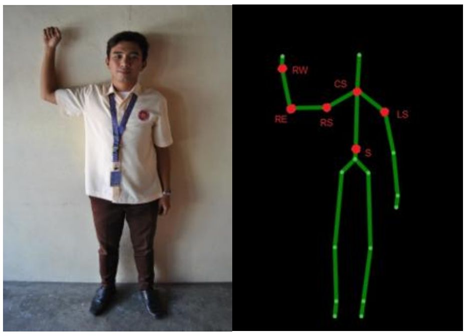
Roselito E. Tolentino is with Polytechnic University of the Philippines – Santa Rosa Campus and De La Salle University – Dasmarinas as part time Instructor.

Then by applying the concept of Vector Multiplication, using that coordinates, the vector needed for the pitch, yaw and roll will be constructed. After that, the acquisition of angles will be next by the use of Dot Product of the two vectors. The data angles are processed by the computer and then sent to the microcontroller and the microcontroller input the required angle for the servo motor for the movement of the robotic arm.

A. Acquiring Shoulder Angle

1. Acquisition of Points

The first step to acquire the angle for pitch, yaw and roll of shoulder is to use the Microsoft Kinect SDK's Skeletal Tracking feature to obtain the coordinates of the joints of the user that are needed. Through the use of Visual Studio, the researchers used C# and create a program to acquire the values of the skeletal coordinates. Then by integrating the C# code into LabVIEW, the researchers are able to use the coordinates for shoulder angle for pitch, yaw and roll.



(a) (b)

Fig. 2. (a) The User, (b) Required coordinates of 3D skeletal joint.

For the analysis of the shoulder angles for pitch, yaw and roll, the researchers decided to select the spine, center shoulder, left shoulder, right shoulder, and right elbow and designated those as (Sx, Sy, Sz), (CSx, CSy, CSz), (LSx, LSy, LSz), (RSx, RSy, RSz) and (REx, REy, REz) for the coordinates of spine, center shoulder, left shoulder, right shoulder and right elbow respectively.

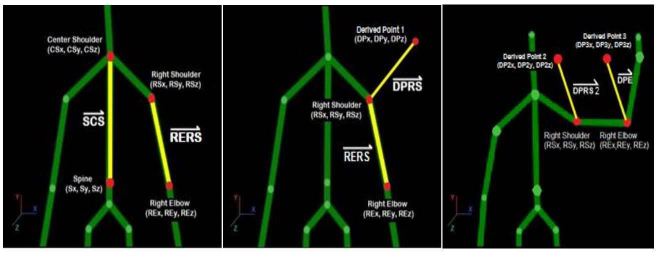
(a) (b) (c) (d) (e)

Fig. 3. Different Point of Views of Kinect to user. (a) POV1: parallel to the Kinect, (b) POV2: 30 degrees of clockwise rotation, (c) POV3:60 degrees clockwise rotation, (d):POV4 30 degrees of counter-clockwise rotation, and (e) POV5: 60 degrees of counter-clockwise rotation against the Kinect.

The researchers made several trials for every point of view to verify that the coordinates resulted to an angle close to the actual user angle. The joint coordinates gathered as data are then used for calculation of lengths from each of the joints.

2. Construction of Vectors for Pitch, Yaw and Roll

By connecting the selected joint coordinates, a vector will be formed. The magnitude of each of the vector produced by joint coordinates must be known to be able to calculate the pitch, yaw and roll angle. For the pitch, these vectors were named as SCS and RSRE for the imaginary lines from the spine to center shoulder and right shoulder to right elbow, respectively. For the yaw, these vectors were named as DPRS1 and RSRE for the imaginary lines from the derived point 1 to right shoulder and right shoulder to right elbow, respectively. For the roll, these vectors were named as DPRS2 and DPRS3 for the imaginary lines from the derived point 2 to right shoulder and derived point 3 to right elbow, respectively.



(a) (b) (c)

Fig. 4. Construction of vectors for (A) pitch, (B) yaw and (C) roll.

To determine the vector needed for pitch, yaw and roll the following formula is used:

- For Pitch
  Vector SCS and RERS

$$\overrightarrow{SCS}=(S_X-CS_X)i+(S_Y-CS_Y)j+(S_Z-CS_Z)k \tag{1}$$

$$\overrightarrow{RERS}=(RE_X-RS_X)i+(RE_Y-RS_Y)j+(RE_Z-RS_Z)k \tag{2}$$

- For Yaw
  Vector DPRS1 and RERS

$$\begin{aligned}\overrightarrow{DPRS\,1}=&\left[(LS_Y-RS_Y)(S_Z-RS_Z)-(LS_Z-RS_Z)(S_Y-RS_Y)\right]i\\&+\left[(LS_X-RS_X)(S_Z-RS_Z)-(LS_Z-RS_Z)(S_X-RS_X)\right]j\\&+\left[(LS_X-RS_X)(S_Y-RS_Y)-(LS_Y-RS_Y)(S_X-RS_X)\right]k\end{aligned} \tag{3}$$

$$\overrightarrow{RERS}=(RE_X-RS_X)i+(RE_Y-RS_Y)j+(RE_Z-RS_Z)k \tag{4}$$

- For Roll
  Vector DPRS2 and DPE

$$\overrightarrow{DPRS\,2}=\left[(S_Y-RS_Y)(RE_Z-RS_Z)\right]i-\left[(S_Y-RS_Y)(RE_X-RS_X)\right]k \tag{5}$$

$$\begin{aligned}\overrightarrow{DPE}=&\left[(RW_Y-RE_Y)(RS_Z-RE_Z)-(RW_Z-RE_Z)(RS_Y-RE_Y)\right]i\\&+\left[(RW_X-RE_X)(RS_Z-RE_Z)-(RW_Z-RE_Z)(RS_X-RE_X)\right]j\\&+\left[(RW_X-RE_X)(RS_Y-RE_Y)-(RW_Y-RE_Y)(RS_X-RE_X)\right]k\end{aligned} \tag{6}$$

3. Angle Acquisition

With the use of vector Dot Product, the angle of two vectors can be calculated. The acquired angles for pitch, yaw and roll are then used to control the robotic shoulder movement. This is done using the formulas below.

- Pitch

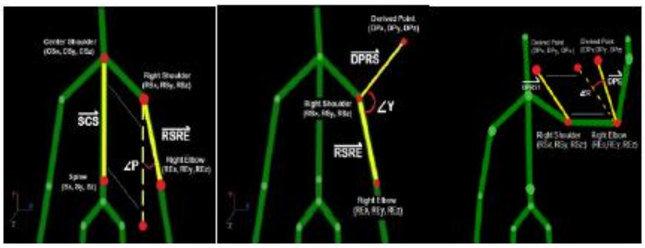
$$\angle P=\cos^{-1}\frac{\left[(S_X-CS_X)(RE_X-RS_X)\right]+\left[(S_Y-CS_Y)(RE_Y-RS_Y)\right]+\left[(S_Y-RS_Y)(RE_Z-RS_Z)\right]}{SCSRERS} \tag{7}$$

- Yaw

$$\angle Y = \cos^{-1}\frac{\begin{bmatrix}(LS_Y - RS_Y)(S_Z - RS_Z)\\ -(LS_Z - RS_Z)(S_Y - RS_Y)\end{bmatrix}[RE_X - RS_X] + \begin{bmatrix}(LS_X - RS_X)(S_Z - RS_Z)\\ -(LS_Z - RS_Z)(S_X - RS_X)\end{bmatrix}[RE_Y - RS_Y] + \begin{bmatrix}(LS_X - RS_X)(S_Y - RS_Y)\\ -(LS_Y - RS_Y)(S_X - RS_X)\end{bmatrix}[RE_Z - RS_Z]}{DPRS1RERS} \qquad (8)$$

- Roll

$$\angle R = \cos^{-1}\frac{[(S_Y - RS_Y)(RE_Z - RS_Z)]\begin{bmatrix}(RW_Y - RE_Y)(RS_Z - RE_Z)\\ -(RW_Z - RE_Z)(RS_Y - RE_Y)\end{bmatrix} + [(S_Y - RS_Y)(RE_X - RS_X)]\begin{bmatrix}(RW_X - RE_X)(RS_Y - RE_Y)\\ -(RW_Y - RE_Y)(RS_X - RE_X)\end{bmatrix}}{DPRS2DPE} \qquad (9)$$



(a) (b) (c)

Fig. 5. Calculation of shoulder angle for (A) pitch, (B) yaw, and (C) roll.

B. Computation of the significant difference between robotic shoulder angle and human shoulder angle in different variation of point of views of the System

1. Creating of Robotic Shoulder

The researchers created a robotic shoulder to assess the effectiveness of the approach being implemented. In order to mimic the movements of the user as seen from the Kinect sensor, the researchers made use of three high torque servo motors for pitch, yaw and roll. Then the motors were positioned in the robotic shoulder mechanism that the researchers fabricated. These motors were controlled and evaluated through the use of Arduino interfaced with LabVIEW program. In order to acquire the angle that the robotic shoulder will perform, the researchers positioned one potentiometer in each of the pitch, yaw and roll mechanisms. These potentiometers are also evaluated through the use of Arduino interfaced with LabVIEW program.

2. Creating of Wearable Sensor

The researchers assessed the response of the robotic shoulder with respect to the human shoulder's actual value of the user's movement to verify whether the robotic shoulder could still acquire the same shoulder angle even the user is varying its point of view. To acquire the actual shoulder angle of the user, the researchers devised a wearable sensor for the shoulder which comprises of three potentiometer, one each for pitch, yaw and roll that were mounted on a 3-DOF mechanical joint. The potentiometer sends the analog signal as data into the Arduino microcontroller. The acquired data will be converted into angular values, in which these values were inputted into LabVIEW's waveform chart feature. Communication between the LabVIEW and the microcontroller is possible through LabVIEW Interface for Arduino (LIFA).

To acquire the prototype shoulder angle, the researchers devised a potentiometer-mounted robotic shoulder to translate its mechanical movement into analog signals. The analog signals as data are interpreted by the microcontroller and is sent into the computer for the LabVIEW to process. The data are converted into angular values, in which these values are inputted into LabVIEW's waveform chart feature. Communi-cation between the LabVIEW and the microcontroller is po-ssible through LabVIEW Interface for Arduino.

Fig. 6. Process in acquiring data of human shoulder and the robotic shoulder.

In order to evaluate the significant difference, the user performed sets of poses for shoulder pitch, yaw and roll. Then, from each of the pose and movement done, the researchers performed the z-test for pitch, yaw and roll using the recorded actual and prototype angles as data.

Angular data from the robotic shoulder and the user are acquired and are plotted using NI LabVIEW. The plotted response of the system are extracted into an Excel Worksheet where the evaluation of the data are made. To evaluate the response, the researchers applied z-test using the acquired user and robotic shoulder angle for pitch, yaw and roll. In z-test it is necessary to define the null hypothesis (Ho), alternative hypothesis (H1) and the critical value that will prove the hypothesis is true.

To know the critical value for a two-tailed test, the significance level (α) is set to 5%. Setting this significance value will create a confidence of 95% (obtained by 100% - α), the area of the curve as the critical value is 0.975 (obtained by 1 – (α/2)). Knowing the area, the researchers used the z-test table (area under the normal curve) and found the critical value 1.96.

The researchers will obtain the z value by using the z-test equation below:

$$Z = \frac{\overline{X}_1 - \overline{X}_2}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}} \qquad (10)$$

Where:

Z = z-test result
$\overline{X}_1$ = mean of the 1st group
$\overline{X}_2$ = mean of the 2nd group
n_1 = number of samples in the 1st group
n_2 = number of samples in the 2nd group

σ_1 = standard deviation of the 1st group

σ_2 = standard deviation of the 2nd group

III. RESULTS AND DISCUSSION

A. Shoulder Angle Acquisition

For the purpose of gathering data, the user performed a pose where pitch, yaw and roll are at 90, 0 and 90 degrees respectively. These poses are to be performed at five different point of views of the Kinect.

For the different point of views made by the user, the researchers obtained 3D coordinates of the joints essential in getting the pitch, yaw and roll angles. The coordinates that the researchers used are Center Shoulder, Right Shoulder, Spine, Right Elbow and Right Wrist. After getting the coordinates of the needed joints, the coordinates are used to obtain different vectors using cross product approach of vector multiplication. Five vectors were derived from the coordinates of the joints created by the skeletal tracking feature of the Microsoft Kinect. Vector RERS and Vector SCS, Vector DPRS1 and Vector RERS, and Vector DPRS2 and Vector DPE were used in getting the angle of pitch, yaw and roll respectively. The researchers then used the obtained vectors to acquire the computed shoulder angles of the shoulder using dot product of vector multiplication.

Table I. The Actual and Computed angle, the Angle Difference and the Total Angle Difference for Pitch, Yaw and Roll in Different Point of Views

Point Of View	Actual Angle			Computed Angle			Angle Difference			Total Angle Difference
	Pitch	Yaw	Roll	Pitch	Yaw	Roll	Pitch	Yaw	Roll	
1	90	-2	-87	90	-3	-87	0	1	0	1
2	92	-2	-91	91	-1	-89	1	1	2	4
3	91	-6	-93	89	0	-88	2	6	5	13
4	91	1	-93	90	4	-90	1	3	3	7
5	89	0	-94	86	5	-89	3	5	5	13
Average	90.6	-1.8	-91.6	89.2	0.8	-88.6	1.4	3.2	3	7.6

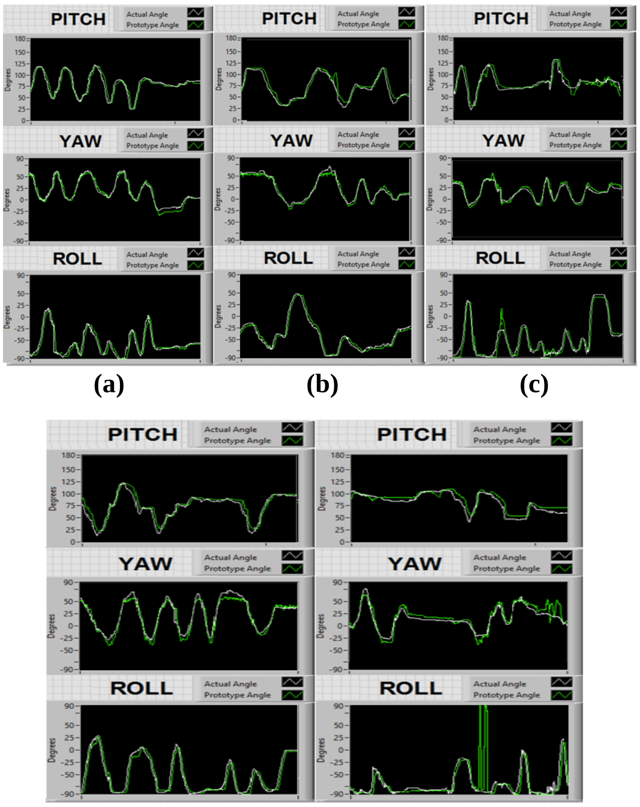
As shown on Table I, the researchers got different angle values for pitch, yaw and roll at different point of views. The values of the angles from the Kinect sensor that the researchers computed are close with the value of user's actual shoulder angles with average angular difference (in degrees) at 1.4, 3.2, and 3.0 for pitch, yaw and roll respectively. The average angular difference of pitch has the smallest value compared with yaw and roll. Therefore, amongst the three, pitch is the most reliable. On the other hand, yaw has the highest value of average angular difference making it the least reliable angle.

Moreover, the total angular difference varies at different point of views. At POV1, the total angular difference is just 1 degree. It is because at POV1 the user is directly facing towards the Kinect sensor and there is a very small chance that the joints will overlap at this point of view, therefore the angular difference is minimized. At POV2, the total angular difference is 4 degrees which is larger than POV1. At 30 degrees of clockwise rotation against the Kinect's point of view, there is a chance that the joints of the user will overlap. At POV3, the total angular difference is 13 degrees, way higher that POV2 and POV1. At 60 degrees of clockwise rotation against the Kinect's point of view, there is a higher chance the joints of the user will overlap because most of the user's body is inclined to the vision of the Kinect. At POV4, the total angular difference is 7 degrees, a little higher than POV2 with the same angle of rotation of 30 degrees against the Kinect's point of view, but counter-clockwise. At POV5, the total angular difference is 13 degrees with the same angle of rotation of 60 degrees against the Kinect's point of view, but counter-clockwise. Amongst the point of views, POV1 has the lowest total angular difference therefore it is the most reliable. While POV3 and POV5 has the highest total angular difference, therefore the least reliable.

Overall, the angular difference between the computed angle and the user's actual angle is at least 0 degrees and at most 6 degrees, meaning there is only a small difference between the two shoulder angles.

B. Computation of the significant difference between robotic shoulder angle and human shoulder angle in different point of views of the System

In the experiment, the user performed random movements to test the capability and the accuracy of the prototype to mimic its pitch, yaw and roll movements in five different point of views.



(a) (b) (c)

(d) (e)

Fig. 7. Labview panel showing the graph of computed and prototype angle for pitch, yaw and roll for the (a) Point of View 1, (b) Point of View 2, (c) Point of View 3, (d) Point of View 4 and (e) Point of View 5.

Figure 7 shows the graph of the actual and the prototype angle for pitch, yaw and roll for five different point of views. As shown in the graph, for Point of View 1, the computed and the prototype angles were always almost the same to each other. Moreover, at this point of view where the user is directly facing the Kinect sensor, the noise and jitters are at minimum therefore the response of the prototype to the user is realistic.

Likewise, the prototype can easily mimic the user's movements without any unnecessary movements. For Point of View 2 and 4, the values of the actual and the prototype angles were always close to each other even if the user is performing random movements. Also, the noise and jitters are still low therefore the prototype can still mimic the user's movements even there is a few unnecessary movements due to the overlapping of the joints of the user in the Kinect's vision. For Point of View 3 and 5, as shown in the graph, particularly in yaw and roll, the angular difference of the actual and the prototype are close but there was a certain part of the movement of the user that the prototype moved uncontrollably, resulting to the high angular difference. Moving at 60 degrees inclined to the left of the Kinect means a higher chance that several parts of the movements will make the joints overlap, giving the Kinect confusion in where is the joint really at. As a result, noises and jitters were formed in the figure. But overall, the response of the prototype to the user at this point of view was still reliable giving the fact the prototype can still mimic the user's movements. Below is the chart showing in every point of view for every angle and the results of z-test from the actual angle and robotic shoulder angle for pitch, yaw and roll.

Table II. The Z-Test results of every Point of View for Pitch

Point Of View	Prototype Angle			Actual Angle			Results of Z-Test	Comment
	$n1$	$\bar{x}1$	$\sigma1$	$n2$	$\bar{x}2$	$\sigma2$		
1	511	77.1252	24.7924	511	77.0196	24.9489	-0.0679	Null hypothesis is acceptable
2	620	57.6645	27.8096	620	58.6645	30.3121	+0.6053	
3	414	83.8913	17.3169	414	85.1643	18.6944	+1.0176	
4	520	71.3577	19.5917	520	72.2481	20.6597	+0.7131	
5	322	94.4689	11.7177	322	92.9876	11.907	-1.5937	

Table II shows the result of z-test in POV 1 in the value closest to zero, it means that the angular difference of actual and prototype is at minimum, thus POV1 has the most reliable result. In the other hand, result of z-test in POV 3 and 5 have the values closest to the critical value; it means it is in the highest possible acceptable value of z-test. While the result of z-test in POV 2 and 4 have the values not close to zero but also not close to the critical value. Meaning, the response at POV2 and 4 is more reliable than POV3 and 5, but not as reliable as POV1. In comparison, point of views 3 and 5 has the most unreliable z-test result among the five point of views.

Table III. The Z-Test results of every Point of View for Yaw

Point Of View	Prototype Angle			Actual Angle			Results of Z-Test	Comment
	$n1$	$\bar{x}1$	$\sigma1$	$n2$	$\bar{x}2$	$\sigma2$		
1	116	106.793	15.3649	116	107.233	18.5127	-0.1968	Null hypothesis is acceptable
2	103	103.126	16.0065	103	104.845	15.2571	-0.6510	
3	283	121.633	13.5981	283	119.569	18.4964	-1.5121	
4	337	123.991	13.0515	337	123.154	15.0025	-0.7725	
5	160	112.744	13.6425	160	109.575	16.3170	-1.8846	

Table II shows the result of z-test in POV 1 in the value closest to zero, it means that the angular difference of actual and prototype is at minimum. Therefore, POV1 has the most reliable result. On the other hand, the z-test result in POV 3 and 5 have the values closest to the critical value. It means that at POV 3 and 5, the response is reliable but not as reliable as POV1. While the result of z-test in POV 2 and 4 have the values not close to zero but also not close to the critical value. Meaning, the response at POV2 and 4 is more reliable than POV3 and 5, but not as reliable as POV1. Like the result of table 4.4, point of views 3 and 5 has the most unreliable z-test result among the five point of views.

Table IV. The Z-Test results of every Point of View for Roll

Point Of View	Prototype Angle			Actual Angle			Results of Z-Test	Comment
	$n1$	$\bar{x}1$	$\sigma1$	$n2$	$\bar{x}2$	$\sigma2$		
1	128	63.2343	25.6731	128	62.9609	25.657	-0.0852	Null hypothesis is acceptable
2	150	65.9867	20.9128	150	65.7133	22.8086	-0.1082	
3	155	57.2194	25.6989	155	58.7742	23.0613	+0.5606	
4	334	65.3862	20.4887	334	64.5689	19.2355	-0.5315	
5	250	62.488	29.5051	250	65.516	30.3507	+1.1403	

Table IV shows the result of z-test in POV 1 has the value closest to zero, it means that the angular difference of actual and prototype is at minimum. To add, it means that, like the result in pitch and yaw, POV1 has the most reliable result among the 5 point of views. On the other hand, the z-test result in POV 3 and 5 having the value of +0.5606 and +1.1403, have the values closest to the critical value. Thus, at POV 3 and 5, the response is reliable but not as reliable as POV1. While the result of z-test in POV 2 and 4 have the values not close to zero but also not close to the critical value. It means that the response at POV2 and 4 is more reliable than POV3 and 5, but not as reliable as POV1. Like the result of table 4.4 and 4.5, point of views 3 and 5 has the most unreliable z-test result among the five point of views. Overall, all of the z-test results are in the acceptance region. Therefore, there is no significant value between the actual angle and the robotic shoulder angle for pitch, yaw and roll.

IV. CONCLUSION

Based from the summary of the data, the joint coordinates and the value of the vector constructed by the coordinates differ as the user changes his point of view. Even though the value of the vector differs, still the value of the prototype's angle ended up to a result close to the actual angle. Thus, system that the proponents proposed was effective in getting the angle for pitch, yaw and roll. However, POV 3 and 5 had a result having a large value of total angular difference, making it the least reliable point of view. In particular, the yaw of POV3 had an angular difference of 6 degrees, which is the highest among the angular differences presented by the proponents. It is because, at a higher angle of rotation of the user against the Kinect's point of view, there is higher chance that the joints of the user will overlap. This is why POV 1 had the result having a minimum value of total angular difference, because the user is directly facing towards the Kinect.

In terms of the statistical data gathered by the proponents, all of the z-test results for pitch, yaw and roll angles at different point of views were within the range of -1.96 and 1.96. Therefore, the proponents concluded that the prototype can effectively mimic the user's movements even though the user is not directly facing the Kinect.

The data also came up with the result of pitch, yaw and roll showing POV 1 with a z-test result close to zero and POV 3 and 5 with a z-test result almost close to the critical value. This supports the first conclusion that POV 1 is the most reliable and POV 3 and 5 are the least reliable point of views.

Although all of the z-test results for pitch, yaw and roll were within the range of acceptance region, the proponents observed that there are some fluctuations of signals in the graphical response of the system. It is when the coordinates of the joints were overlapping at the Kinect's point of view and also when the Kinect did not recognized the skeletal parts of the user. Because of that, the Kinect cannot detect the actual position of the joints needed to compute the angle. As a result, the prototype does such unnecessary movements and results to an unstable response of the system which leads to unreliable results.

RECOMMENDATION

For future works, the proponents recommended to use the Vector Multiplication Approach in acquiring other human joint angle, such as ankle and neck angle. New methods on how to acquire other body joint angles may be developed by using this approach. In addition, the next researchers may use multi- Kinect to work together in order to compensate the lack of sight of only one Kinect sensor. Thus, it will lessen or eliminate the chance of overlapping joints and minimize the jitters of the response. With this, the future proponents will even have a more reliable result.

REFERENCES

1. Rajesh Kannan Megalingam, Nihil Saboo, Nitin Ajithkumar, Sreeram Unny & Deepansh Menon. (2013). Kinect Based Gesture Controlled Robotic Arm: A research work at HuT Labs. Kollam, Kerala.
2. Mohammed Z. Al-Faiz & Ahmed F. Shanta. (2015). Kinect-Based Humanoid Robotic Manipulator for Human Upper Limbs Movements Tracking. Baghdad, Iraq: Scientific Research Publishing Inc.
3. Mohammed A. Hussein, Ahmed S. Ali, F.A. Elmisery & R. Mostafa. (2014). Motion Control of Robot by using Kinect Sensor. Egypt.
4. Dr.Himani Goyal, Hari Kishore Thadisetty, Samboju Srinath & Poluri Ashish. Motion Controlled Semi Humanoid Robot Using Microsoft Kinect Console. Dundigal, Hyderabad, India, 2014.
5. Rizqa Afthoni, Achmad Rizae & Erwin Susanto. (2013). Proportional Derivative Control Based Robot Arm System Using Microsoft Kinect. Yogyakarta, Indonesia.
6. Van Vuong Nguyen & Joo-Ho Lee. (2012). Full-Body Imitation of Human Motions With Kinect And Heterogeneous Kinematic Structure Of Humanoid Robot. Fukuoka, Japan.
7. Meghan E. Huber, Amee L. Seitz, Miriam Leeser & Dagmar Sternad. (2014). Validity and Reliability of Kinect for Measuring Shoulder Joint Angles. Boston, MA, USA.
8. Michael Waddell, Joel Villasuso, Daniela ChavezGuevera & Jong-Hoon Kim. (2013). Standardized Linearization and Vectorization Algorithm for Arm Motion Control of A Humanoid Telepresence Robot. Miami, FL 33199 USA.
9. Ren C. Luo, Bo-Han Shih & Tsung-Wei Lin. (2013). Real Time Human Motion Imitation of Anthropomorphic Dual Arm Robot Based on Cartesian Impedance Control. Taipei, Taiwan.
10. Goehler, Craig M. (2007). DESIGN OF A HUMANOID SHOULDER-ELBOW COMPLEX. Notre Dame, Indiana.
11. Floyd, R. T. (2012). Manual of Structural Kinesiology (18th ed.). 1221 Avenue of the Americas, New York: McGraw-Hill.
12. Microsoft. (2015). Kinect for Windows. Available from https://msdn.microsoft.com/en-us/library/jj131033.aspx.

AUTHORS PROFILE

Rollie Jay R. Ortega, was born in San Pedro, Laguna, Philippines on January 29, 1994. He received the degree in Bachelor of Science in Electronics and Communications Engineering from the Polytechnic University of the Philippines Santa Rosa Campus, Laguna in 2016. He was a part of ON Semiconductor Philippines Inc., Carmona Cavite, Philippines as a team member of the ATE Board Repair Area, Engineering Department for his On-the-Job training 1 in 2014. He was also a part of Eat Bulaga! Television And Production Exponents, Inc., Quezon City, Philippines as a team member of the Engineering Department for his On-the-Job training 2 in 2015. Mr. Ortega is currently a member of Institute of Electronics Engineers of the Philippines. And Association of Electronics and Communication Engineering Students.

Norm Allen T. Siron, was born in San Pedro Laguna, Philippines on October 7, 1994. He received the degree in Bachelor of Science in Electronics and Communications Engineering from the Polytechnic University of the Philippines Santa Rosa Campus, Laguna in 2016. He was a part of Global Lighting Phils., Inc., Carmona Cavite, Philippines as a team member of the Engineering Department for his On-the-Job training 1 in 2014. He was also a part of SMART Communications Inc., City of Cabuyao Laguna, Philippines as a team member of the Wireless Access and Transport Operation Department for her On-the-Job training 1 in 2015. Mr. Siron is currently a member of Institute of Electronics Engineers of the Philippines. And Association of Electronics and Communication Engineering Students.

Kevin T. Mapalad, was born in Carmona Cavite, Philippines on May 19, 1995. He received the degree in Bachelor of Science in Electronics and Communications Engineering from the Polytechnic University of the Philippines Santa Rosa Campus, Laguna in 2016. He was a part of Global Lighting Phils., Inc., Carmona Cavite, Philippines as a team member of the Engineering Department for his On-the-Job training 1 in 2014. He was also a part of Eat Bulaga! Television And Production Exponents, Inc., Quezon City, Philippines as a team member of the Engineering Department for his On-the-Job training 2 in 2015. Mr. Mapalad is currently a member of Institute of Electronics Engineers of the Philippines. And Association of Electronics and Communication Engineering Students.

Heidrick S. Emano, was born in Metro Manila, Philippines on June 3, 1995. He received the degree in Bachelor of Science in Electronics and Communications Engineering from the Polytechnic University of the Philippines Santa Rosa Campus, Laguna in 2016. He was a part of ZILOG Electronics Philippines Inc., Taguig City, Philippines as a team member of the Test Department for his On-the-Job training 1 in 2014. Mr. Emano is currently a member of Institute of Electronics Engineers of the Philippines. And Association of Electronics and Communication Engineering Students.

Roselito E. Tolentino, is a registered Electronics Engineer and IECEP-Member. He is a graduate of B.S. Electronics and Communication Engineering at Adamson University in 2004 under the scholarship of DOST-SEI. He finished his Master of Science in Electronics Engineering Major in Control System at Mapua Institute of Technology under the scholarship of DOST-ERDT. He currently takes up Doctor of Philosophy in Electronics Engineering at the same Institute. He is currently working as a part time instructor at Polytechnic University of the Philippines Santa Rosa Campus and De La Salle University – Dasmarinas. His research interest are more on Robotics and Machine Vision.