

## 3-D Interface for Humanoid Robot Operation

Jacobo E. Cruz-Silva, J. Yaljá Montiel-Pérez, J. Humberto Sossa-Azuela

Instituto Politécnico Nacional,  
Centro de Investigación en Computación,  
Mexico

jacobo8806@hotmail.com, yalja@ipn.mx, humbertosossa@gmail.com

**Abstract.** The study of human body motions, especially upper and lower limbs motions, help to understand how the human body works. This paper presents a method to obtain the human limb angles through a kinematic model depicted by Roll-Pitch-Yaw rotation matrix and subsequently control the humanoid robot motions. The main advantage of this model is the detailed representation of each joint motion expressed in the coordinate axes  $(x, y, z)$ . The estimation of human limb angles is performed with information obtained by algorithms of artificial vision and artificial intelligence. In order to reduce the latency between the human motion capture and robot motions, a fuzzy logic controller is implemented to control each robot joint due the less complexity of these kind of algorithms. The final robot limbs angles are compared with the human limbs angles to obtain a final error between those measurements. This method shows a similar results on the arms posture regarding previous works.

**Keywords.** Fuzzy control, human body, kinematics, modelling, NAO robot, roll-pitch-yaw.

### 1 Introduction

The knowledge of human limb motion has an important relevance in robotics when a robot is wanted to perform human-like behaviours. The robotic teleoperation is the action to control a robot at a distance.

One of the most important method to teleoperate robots is through haptic devices. These devices can be joysticks or special suits like gloves or full-body suits that are equip with inertial measurement unit (IMU) and are able to transfer force and positions to a robot in order to perform an specific motion [15, 14, 12, 9, 19, 10, 1, 16, 18].

The disadvantage of these methods is the sensors noise that can carry out low precision on the final motion of the robot. Other remarkable disadvantage is the suits are made for a fixed size and not all person can use them.

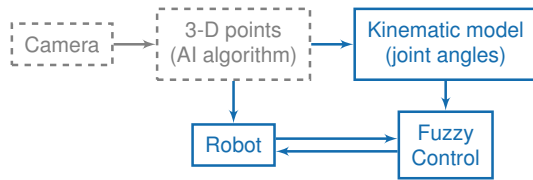
Another method is capture specific points of the body through artificial vision and artificial intelligence, afterwards use them as the set-points for the robot. The motion capture can be with an algorithm that search for a specific color in an image, for example a mark on person's clothes.

Also, the augmented reality, that is the combination of real word images and information generated by a computer, in combination with haptic devices improves the teleoperation capabilities [8, 17, 11].

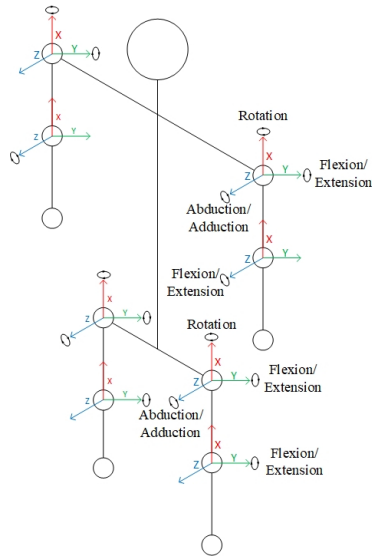
In Health Sciences, the robots teleoperation is applied to rehabilitation and therapy techniques such as therapies for children with autism [3, 5, 4]. In the present paper, a method to mimic the human limbs motions using three-dimensional kinematic model and a fuzzy logic control algorithm.

The kinematic model receives as input the points array from an artificial intelligence algorithm [2] which correspond to the locations of human joint, this array contains the points obtained by three-dimensional analysis of 4 plain images that are taken from 4 different points of view (front, back, right side, left side) instead of a 3-D approximation from a single point of view of a bi-dimensional image [20].

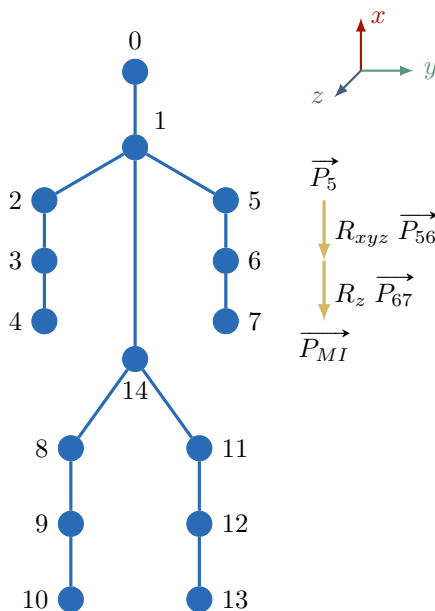
Then, the angles of limbs are obtained by three equations which are derived from kinematic model. These angles correspond to the motions that human limbs can perform on the different axis and are the set point of the fuzzy control algorithms of



**Fig. 1.** NAO robot teleoperation method



**Fig. 2.** Upper and lower limb joints



**Fig. 3.** Key points of human body

the robot. The fuzzy logic algorithms are Mamdani fuzzy inference systems with “if, else” rules engine.

On the Fig. 1 is depicted a general flow diagram of this method. The artificial intelligence algorithm [2] obtains the information of four cameras equipped with two different sensors, RGB sensor and depth sensor.

The images captured by the 4 RGB sensors are processed with a Convolutional Neural Network which creates feature maps of input image and infers 2-D key points for person in the image. The 2-D key points are the locations of person’s joints. After that, the 3-D points are calculated making the relation between four set of 2-D points.

Two depth sensors work as security system that activate the RGB data acquisition till the person is in the middle of cameras array. This system ensures that person position in the images is similar in order to minimize error when the 3-D points are calculated.

The artificial intelligence algorithm [2] provides a robust method to detect the joint’s location. The measurements are not affected by the lighting changes and it was not necessary to use special equipment on the body, that is the main advantage of the algorithm.

## 2 Method

### 2.1 Kinematic Model

The kinematic model was based on the five basic motions that the NAO robot [6] and humans can perform [13, 7]. These movements are described below:

- **Flexion.** It is angle decrease at one joint.
- **Extension.** It is angle increase at one joint.
- **Abduction.** It is the motion of a limb away from the midline of the body.
- **Adduction.** It is the motion of a limb toward to the midline of the body.
- **Rotation.** It is the motion around a longitudinal axis of a bone, it can be internal or external.

The motions of upper limbs parts considered in this method were shoulders and elbow motions; the shoulders motions are Flexion / Extension ( $y$ ), Abduction / Adduction ( $z$ ) and Rotation ( $x$ ).

The elbows motions are Flexion / Extension ( $z$ ). Similarly, the motions of lower limbs are hips motions and knees motions.

The hips motions are Flexion / Extension ( $y$ ), Abduction / Adduction ( $z$ ) and Rotation ( $x$ ). The knees motions are Flexion / Extension ( $y$ ) as is shown on the Fig. 2.

The Roll, Yaw and Pitch rotations matrixes were selected to describe the previous motions. The Equation 1 is the rotation matrix on the ( $z$ ) axis (Yaw) and the Equation 2 is the rotation matrix ( $y$ ) (Pitch), these equations are used to represent the elbows, knees, shoulders and hips Abduction / Adduction and Flexion / Extension motions respectively.

The Equation 3 is the rotation matrix on the ( $x$ ) axis (Roll) and represents the Rotation motions performed by the shoulders and hips:

$$R_z = \begin{bmatrix} C_\varphi & -S_\varphi & 0 \\ S_\varphi & C_\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (1)$$

$$R_y = \begin{bmatrix} C_\theta & 0 & S_\theta \\ 0 & 1 & 0 \\ -S_\theta & 0 & C_\theta \end{bmatrix}, \quad (2)$$

$$R_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & C_\psi & -S_\psi \\ 0 & S_\psi & C_\psi \end{bmatrix}, \quad (3)$$

where  $C_\varphi$ ,  $C_\theta$  and  $C_\psi$  represent  $\cos \varphi$ ,  $\cos \theta$  and  $\cos \psi$  respectively and  $S_\varphi$ ,  $S_\theta$  and  $S_\psi$  represent  $\sin \varphi$ ,  $\sin \theta$  and  $\sin \psi$  respectively. The angles  $\varphi$ ,  $\theta$  and  $\psi$  are the rotation angles on the axis ( $z$ ), ( $y$ ) and ( $x$ ) respectively.

Due the shoulders and hips are able to perform rotations on the 3 axis, the 3 previous equations were multiplied to obtain a single rotation matrix, equation 4:

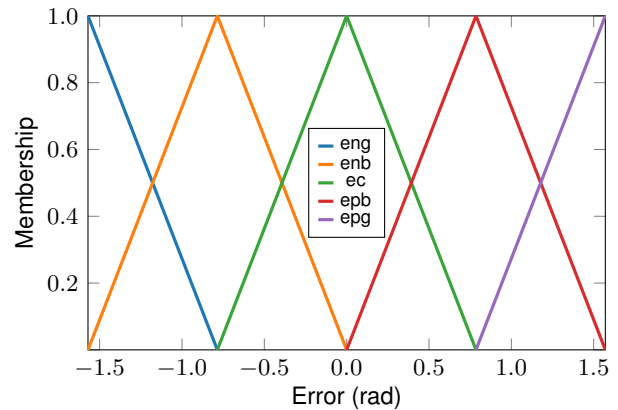
$$R_{xyz} = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix}. \quad (4)$$

**Table 1.** Comfort angles of right limbs

Part	Movement	Axis	ID	Angle Range
shoulder	flex / ext	Y	$q_1$	-150° to 40°
	abd / add	Z	$q_2$	-150° to 20°
	rotation	X	$q_3$	-70° to 60°
elbow	flex / ext	Z	$q_4$	10° to 150°
	flex / ext	Y	$q_6$	-130° to 15°
hip	abd / add	Z	$q_8$	-45° to 20°
	rotation	X	$q_7$	-50° to 45°
knee	flex / ext	Y	$q_8$	0° to 155°

**Table 2.** Comfort angles of left limbs

Part	Movement	Axis	ID	Angle Range
shoulder	flex / ext	Y	$q_9$	-150° to 40°
	abd / add	Z	$q_{10}$	-20° to 150°
	rotation	X	$q_{11}$	-70° to 60°
elbow	flex / ext	Z	$q_{12}$	-150° to 10°
	flex / ext	Y	$q_{13}$	-130° to 15°
hip	abd / add	Z	$q_{14}$	-20° to 45°
	rotation	X	$q_{15}$	-50° to 45°
knee	flex / ext	Y	$q_{16}$	0° to 155°



**Fig. 4.** Error membership functions

where:

$$\begin{aligned} A_{11} &= C_\varphi C_\theta, \\ A_{12} &= C_\varphi S_\theta S_\psi - S_\varphi C_\psi, \\ A_{13} &= S_\varphi S_\psi + C_\varphi S_\theta C_\psi, \\ A_{21} &= S_\varphi C_\theta, \\ A_{22} &= C_\varphi C_\psi + S_\varphi S_\theta S_\psi, \\ A_{23} &= -C_\varphi S_\psi + S_\varphi S_\theta C_\psi, \\ A_{31} &= S_\theta, \\ A_{32} &= C_\theta S_\psi, \\ A_{33} &= C_\theta C_\psi. \end{aligned}$$

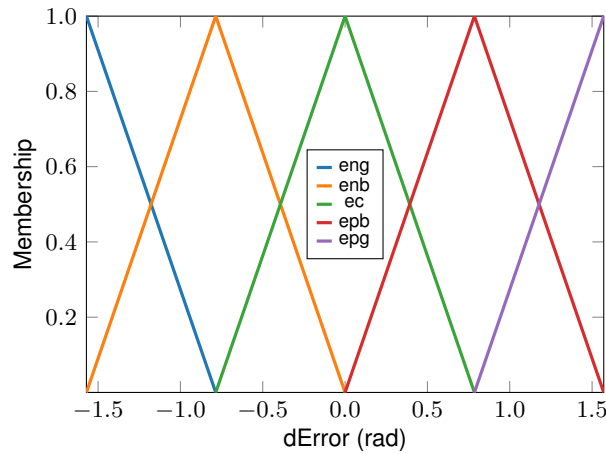


Fig. 5.  $\Delta$ Error membership functions

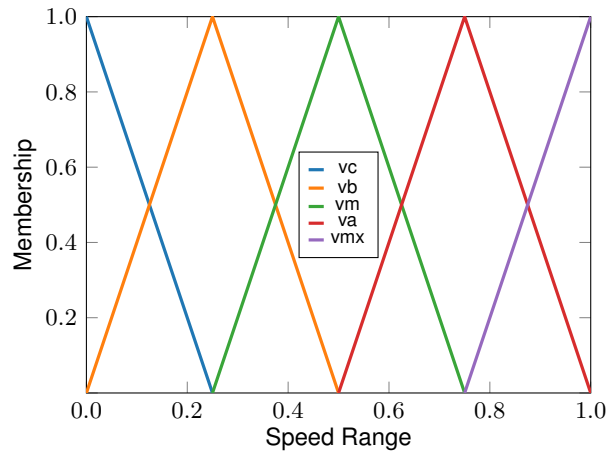


Fig. 6. Output memberships functions

Table 3. Fuzzy inference rules

e / $\Delta e$	eng	enb	ec	epb	epg
eng	vmx	va	vm	va	vmx
enb	va	vm	vb	vm	va
ec	vm	vb	vc	vb	vm
epb	va	vm	vb	vm	va
epg	vmx	va	vm	va	vmx

The artificial intelligence algorithm [2] is able to locate 15 key points. Three of these key points were used as reference point (0, 1, 14) and the others key points were used to estimate the joints angles. These key points are shown on the Fig. 3.

The kinematic model for the upper and lower limbs is described by the following equations:

$$\vec{P}_{MD} = R_z \vec{P}_{34} + R_{xyz} \vec{P}_{23} + \vec{P}_2, \quad (5)$$

$$\vec{P}_{MI} = R_z \vec{P}_{67} + R_{xyz} \vec{P}_{56} + \vec{P}_5, \quad (6)$$

$$\vec{P}_{TD} = R_y \vec{P}_{910} + R_{xyz} \vec{P}_{89} + \vec{P}_8, \quad (7)$$

$$\vec{P}_{TI} = R_y \vec{P}_{1213} + R_{xyz} \vec{P}_{1112} + \vec{P}_{11}, \quad (8)$$

where the Equation 5 describes the final position of the right arm with the vector  $\vec{P}_{MD}$ . The equation is composed of a translation  $\vec{P}_2$  from the point 0,  $P_0 = [0 \ 0 \ 0]^T$ , to the point 2.

A second translation from the point 2 to the point 3 creates the vector  $\vec{P}_{23}$ . This second vector has a rotation described by the matrix  $R_{xyz}$ .

Finally, the vector  $\vec{P}_{34}$  and the matrix  $R_z$  represents the translation from the point 3 to the point 4 and his rotation, respectively.

This procedure was applied to the other limbs, where Equation 6 corresponds to the left arm, Equation 7 to the right leg and Equation 8 to the left leg.

In Tables 1 and 2 are shown the maximum angle that human limbs can perform before get injured. These angles are know as comfort angles [13, 7]. The vector value of each limb was calculated by the Equation 9:

$$\vec{V}_{ext} = \begin{bmatrix} (X_e - X_0) \\ (Y_e - Y_0) \\ (Z_e - Z_0) \end{bmatrix} = \begin{bmatrix} X_{ext} \\ Y_{ext} \\ Z_{ext} \end{bmatrix}, \quad (9)$$

where  $(X_0, Y_0, Z_0)$  are the limb origin points and  $(X_e, Y_e, Z_e)$  are the end points. For example, the forearm origin point is the elbow an the end point is the wrist, hence  $\vec{V}_{ext}$  is the final vector with the point  $(X_{ext}, Y_{ext}, Z_{ext})$ .

To obtain each motion angle, the directional cosines were selected to estimate the rotation angles  $q_n$  on the different axis, where  $q_n$  is the ID of angles shown in the Tables 1 and 2.

The director cosine for the rotation angle at the  $(x)$  axis is the the Equation 10 and for the  $(y)$  and the  $(z)$  axis are the Equations 11 and 12 respectively:

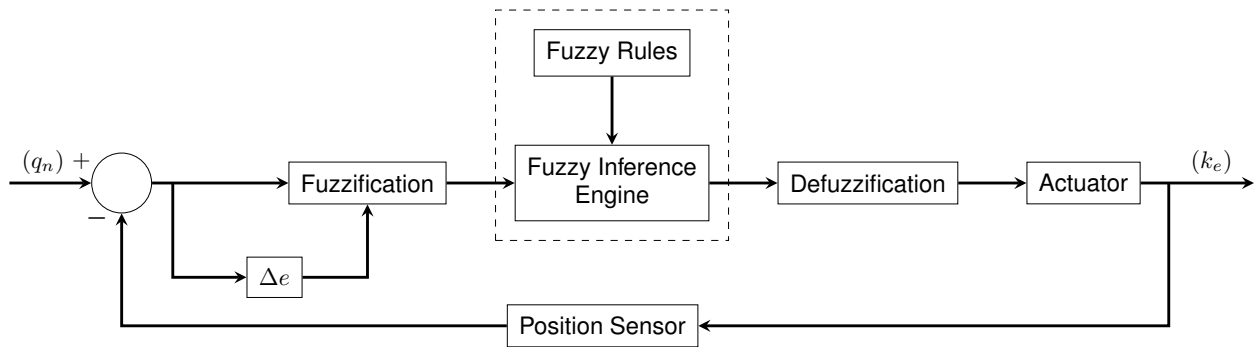


Fig. 7. Fuzzy Controller

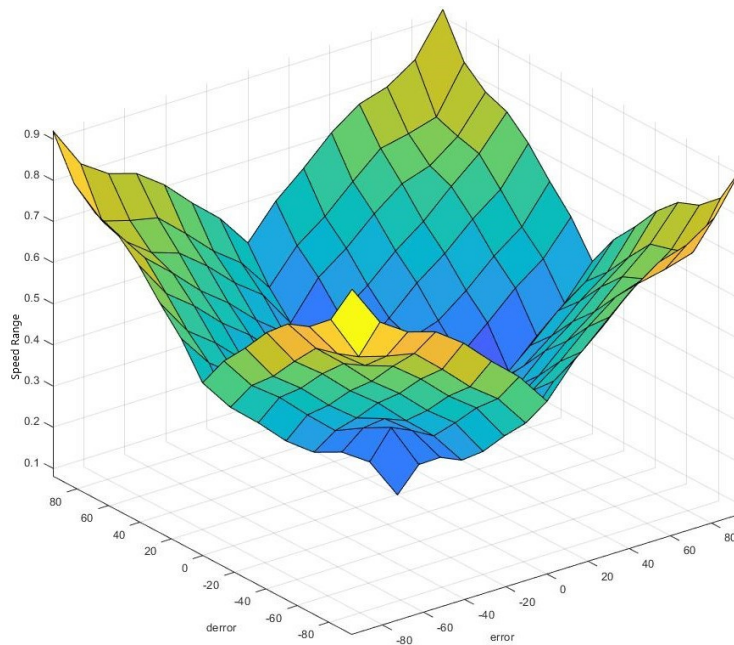


Fig. 8. Control surface

$$q_n = \cos^{-1} \left( \frac{Y_{ext}}{|\vec{V}_{ext}|} \right), \quad (10)$$

$$q_n = \cos^{-1} \left( \frac{Z_{ext}}{|\vec{V}_{ext}|} \right), \quad (11)$$

$$q_n = \cos^{-1} \left( \frac{X_{ext}}{|\vec{V}_{ext}|} \right), \quad (12)$$

where  $Y_{ext}$  corresponds to the  $(y)$  point from the vector  $\vec{V}_{ext}$  and  $|\vec{V}_{ext}|$  is magnitude of that vector. In the same way,  $X_{ext}$  and  $Z_{ext}$  correspond to the  $(x)$  and  $(z)$  points from vector  $\vec{V}_{ext}$  respectively and  $|\vec{V}_{ext}|$  is magnitude of that vector.

### 2.2 Control Algorithm

The control algorithm used to control the robot's joint speed was a Fuzzy PD (Proportional Derivative) controller.

This controller is a Mamdani Fuzzy Inference System which is based on "if-else" inference engine. For this application, the hip rotations were excluded due the NAO robot [6] is not able to perform this motion.

The controller's set-points were bounded to the angles values in the Tables 1 and 2 in order to prevent a damage at the robot joints. In the first step, the controller, Fig. 7, assigned fuzzy values to error and differential error through input membership function.

The error was estimated subtracting the set-point and the angle measured by the sensors in robot's joints at an instance of time ( $e(t)$ ) and the differential error is the error is  $\Delta e(t) = e(t) - e(t-1)$ .

Then, the fuzzy values from the inputs were processed by an inference engine. Finally, the crisp values, that controls the robot's joints, were estimated by the defuzzification block and sent them to robot's actuators.

The error and the differential error were in radian and could be positive and negative, due this, five triangular memberships functions were assigned. For example, the membership functions from the right shoulder are shown on the Fig 4 and 5 .

Each motion angle have five triangular memberships functions for the error and other five for the differential error. The name assigned to the membership functions are the follow:

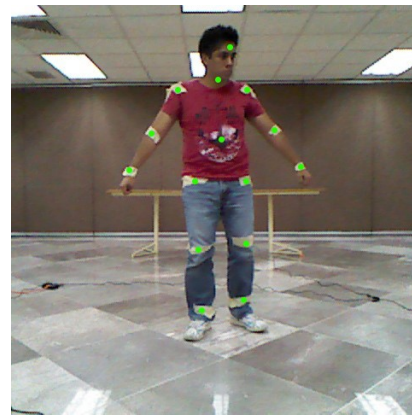
- **eng.** Big Negative Error.
- **enb.** Low Negative Error.
- **ec.** Zero Error.
- **epg.** Big Positive Error.
- **epb.** Low Positive Error.

The robot's speed motion was controlled with values from 0 to 1, which 0 is zero speed and 1 is the maximum speed. Five triangular membership functions were defined in this range.

On the Fig 6 is shown an example from the right shoulder. Similarly to the input, each motion angle has five output membership functions. The name assigned to the membership functions are the follow:

**Table 4.** First stable posture errors

Part	Right ID	P-R	Left ID	P-R
		Error (%)		Error (%)
shoulder	$Y_{q_1} - k_1$	17.3301	$Y_{q_9} - k_9$	12.9054
	$Z_{q_2} - k_2$	7.4301	$Z_{q_{10}} - k_{10}$	53.0586
	$X_{q_3} - k_3$	5.9549	$X_{q_{11}} - k_{11}$	19.4928
elbow	$Z_{q_4} - k_4$	8.4112	$Y_{q_{12}} - k_{12}$	11.8060
hip	$Y_{q_5} - k_5$	22.4334	$Y_{q_{13}} - k_{13}$	14.4368
	$Z_{q_6} - k_6$	19.2864	$Z_{q_{14}} - k_{14}$	42.8673
knee	$Z_{q_8} - k_8$	8.3355	$Y_{q_{16}} - k_{16}$	18.6414



**Fig. 9.** Key points locations on human

- **vc.** Zero Speed.
- **vb.** Low Speed.
- **vm.** Medium Speed.
- **va.** High Speed.
- **vmx.** Maximum Speed.

The inference engine was defined with the rules shown in the Table 3. The fuzzy rules were designed to smoothly decrease the speed when the joint approaches to the set point. These behaviour is shown on the control surface in Fig. 8.

Finally, the crisp value that was sent to the robot was estimated with Center of Gravity (CoG) method. This defuzzification method is shown in the Equation 13:

$$x = \frac{\sum_{i=1}^n \mu_{\bar{c}}(x_i) \cdot x_i}{\sum_{i=1}^n \mu_{\bar{c}}(x_i)}, \quad (13)$$

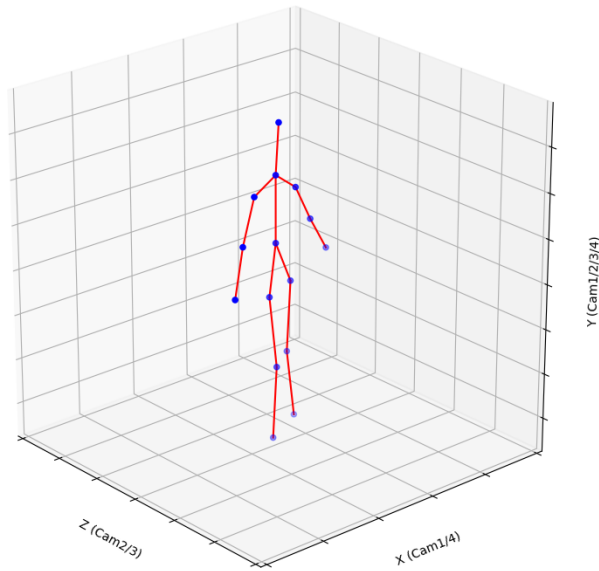


Fig. 10. 3-D Skeleton



Fig. 11. Robot's final posture

where  $x$  is the crisp value,  $\mu_{\bar{c}}$  is the membership function and  $x_i$  output variable.

### 3 Results

The tests were divided into two: stable posture tests and unstable posture test. When the body's center of mass is located at the support base is called as stable posture.

The human is able to perform this kind of posture without loss the balance. If the body's center of mass is not located at the support base, then it is an unstable posture. For this test, the unstable posture was divided into free unstable posture and intervened unstable posture.

When a person lifts the robot in order to avoid the feet touch the ground is an intervened unstable posture, if the robot's feet touch the ground is a free unstable posture. The errors were estimated subtracting the final angle for each joint of the human limbs and robot limbs. These angles are shown in the Tables 1 and 2:

$$E_{RMA(n)} = \left| \frac{k_n - q_n}{1 - |q_n|} \right|, \quad (14)$$

$$E_{RTP} = \frac{E_{RMA(1)} + \dots + E_{RMA(14)}}{14}. \quad (15)$$

The Equation 14 is the Relative Angle Error  $E_{RMA(n)}$  and the Equation 15 is the Total Position Error  $E_{RTP}$ . At the Equation 14,  $k_n$  is the angle from robot's joints and  $q_n$  is the angle from the human's joints. The Equation 15 is the average of each Relative Angle Error  $E_{RMA(n)}$  from the 14 key point.

On the fists test, the person performed an stable posture. The person only extended his arms and did not move the legs. The Table 4 contains the errors from each limb's angle, in this Table P-R is refereed to Person-Robot.

In the Fig. 9, Fig. 10 and Fig. 11 are show examples of the approximated location of human's key points, the skeleton created by the artificial intelligence algorithm [2] and the final robot's posture, respectively.

The second test was a stable posture too. But in this case, the person extended his arms completely in order to perform a "T" posture, the legs remained at the same position of the previous posture.

The Table 5 contains the errors from each limb's angle, in this Table P-R is refereed to Person-Robot. On the third test, the person raised one leg and extend the arms slightly and the robot stayed on the ground because it was a free unstable posture.

The robot's anti-fall safety system located at the at the feet was activated automatically lowering the legs to the ground in order to avoid a fall. The Table 6 contains the errors from each limb's angle, in this Table P-R is refereed to Person-Robot.

The fourth test was an intervened unstable posture, hence a second person lifted the robot in order prevent the activation of robot's anti-fall safety system.

The first person raised one leg, flexed one arm and raised the same arm in order to locate the wrist near to the chest. The Table 7 contains the errors from each limb's angle, in this Table P-R is refereed to Person-Robot.

On the final test the fist person raised one leg and both arms and the robot stayed lifted by the other person in order to perform an intervened unstable posture. The Table 8 contains the errors from each limb's angle, in this Table P-R is refereed to Person-Robot.

Finally the Average Relative Errors for each test was calculated and shown in the Table 9.

Only the Average Relative Errors of the arms was used to compared with the results of the state of the art, due the total error of the posture was not included as a result on the referenced paper. Both results are shown on the Table 10.

## 4 Conclusion and Future Work

With the detailed information of limbs motions provided by the kinematic model was possible to estimate the joint angles without complex equations. These angles were used as control system's set points that controls the joint's motion speed.

The maximum and minimum joint's angles from human and robots are slightly different, due the maximum angles that the robot can perform were lower than the angles that a human can perform.

Even with this difference, the final posture errors resulted only 1% different than the result obtained in a similar work of the state of the art. The method proposed on the state of art [20] used a standard library to locate the human joints based on the information of the RGB sensor and the depth sensor.

**Table 5.** Second stable posture errors

Part	Right ID	P-R	Left ID	P-R
		Error (%)		Error (%)
shoulder	$Y_{q_1} - k_1$	15.3593	$Y_{q_9} - k_9$	3.6016
	$Z_{q_2} - k_2$	13.1042	$Z_{q_{10}} - k_{10}$	16.9010
	$X_{q_3} - k_3$	4.2484	$X_{q_{11}} - k_{11}$	1.9413
elbow	$Z_{q_4} - k_4$	9.4681	$Y_{q_{12}} - k_{12}$	13.2249
hip	$Y_{q_5} - k_5$	32.2980	$Y_{q_{13}} - k_{13}$	19.3410
	$Z_{q_6} - k_6$	33.6234	$Z_{q_{14}} - k_{14}$	26.9505
knee	$Z_{q_8} - k_8$	4.5920	$Y_{q_{16}} - k_{16}$	16.5909

**Table 6.** Unstable posture errors

Part	Right ID	P-R	Left ID	P-R
		Error (%)		Error (%)
shoulder	$Y_{q_1} - k_1$	22.5545	$Y_{q_9} - k_9$	31.8723
	$Z_{q_2} - k_2$	15.2686	$Z_{q_{10}} - k_{10}$	53.1763
	$X_{q_3} - k_3$	25.7007	$X_{q_{11}} - k_{11}$	21.2021
elbow	$Z_{q_4} - k_4$	17.3055	$Y_{q_{12}} - k_{12}$	63.7488
hip	$Y_{q_5} - k_5$	27.9923	$Y_{q_{13}} - k_{13}$	56.8126
	$Z_{q_6} - k_6$	44.0163	$Z_{q_{14}} - k_{14}$	96.9256
knee	$Z_{q_8} - k_8$	30.0295	$Y_{q_{16}} - k_{16}$	98.2249

**Table 7.** First intervened unstable posture errors

Part	Right ID	P-R	Left ID	P-R
		Error (%)		Error (%)
shoulder	$Y_{q_1} - k_1$	7.5838	$Y_{q_9} - k_9$	37.0683
	$Z_{q_2} - k_2$	30.6896	$Z_{q_{10}} - k_{10}$	16.7383
	$X_{q_3} - k_3$	9.9927	$X_{q_{11}} - k_{11}$	30.1306
elbow	$Z_{q_4} - k_4$	24.8028	$Y_{q_{12}} - k_{12}$	6.7421
hip	$Y_{q_5} - k_5$	8.1788	$Y_{q_{13}} - k_{13}$	14.6641
	$Z_{q_6} - k_6$	8.2292	$Z_{q_{14}} - k_{14}$	21.3018
knee	$Z_{q_8} - k_8$	88.5699	$Y_{q_{16}} - k_{16}$	19.5520

The 3-D location of the joints were estimated only with one plane image. This is the main disadvantage of the method due the algorithm cannot estimate the 3-D position of a joint if the depth sensor cannot measure the complete body of the person, for example, if the RGB sensor and the depth sensor can only measure the left side of the human body, the 3-D location of the joint and the teleoperation cannot be performed due the information of the missing parts cannot be inferred.



**Table 8.** Second Intervened unstable posture errors

Part	Right ID	P-R	Left ID	P-R
		Error (%)		Error (%)
shoulder	$Y_{q_1} - k_1$	13.1688	$Y_{q_9} - k_9$	12.4269
	$Z_{q_2} - k_2$	26.2602	$Z_{q_{10}} - k_{10}$	22.1812
elbow	$X_{q_3} - k_3$	16.9042	$X_{q_{11}} - k_{11}$	2.5673
	$Z_{q_4} - k_4$	12.7722	$Y_{q_{12}} - k_{12}$	10.7212
hip	$Y_{q_5} - k_5$	11.5340	$Y_{q_{13}} - k_{13}$	1.6981
	$Z_{q_6} - k_6$	6.2620	$Z_{q_{14}} - k_{14}$	9.1534
knee	$Z_{q_8} - k_8$	68.4082	$Y_{q_{16}} - k_{16}$	28.0653

**Table 9.** Relative Errors for Real Posture – NAO Robot

Test	Average Relative Errors Person Robot (%)
Test 1	18.74
Test 2	15.08
Test 3	43.20
Test 4	23.16
Test 5	17.43

**Table 10.** Results of the state of the art

Test	Arms average errors (%)
Method [20]	18
Proposed method	19

Additionally, the OpenNI library requires a special illumination because the location of the joints cannot be estimated if the RGB image does not have a specific range of colors.

The proposed method in this paper, the 3-D analysis of the person posture was made with the information of 4 plain images from different points of view. Hence, the 3-D location of the joints are more reliable than the method where the 3-D location is estimates using only one image.

Furthermore, the limitation which the person's joints must be located in front of 2 sensor was eliminated. The artificial intelligence algorithm [2] provided the joints location no matter where the person was located on the range of the 4 cameras. The measurements were not affected by the lighting changes due the artificial intelligence

algorithm was trained with different kind of images, those images included bright and dark images. The robot's final posture from Test 3 was affected by the anti-fall safety system programmed by the manufacturer. This can be seen in the error values from the hip and knee.

Thank of the human intervention, the robot could perform a complex posture avoiding the anti-fall safety system activation getting better errors values of the hip and knee. The human body can keep the balance naturally and unconsciously, but the robot cannot do it. That it is the main reason why the manufacturer included the anti-fall safety system.

The fuzzy control algorithm had a favourable performance moving smoothly the robot's limbs to the set-points. The joints speed was decreased when the angle measured by the joint sensor got close to the set-point.

The fuzzy membership functions were correctly set due the maximum operation range and the speed configuration of the NAO limbs were knew that are included in the technical documentation of the manufacturer.

The future research will be focused on test and develop new deep learning models and fuzzy control algorithms to improve the measurement of human limbs and motion control, also the hands and finger motion will be included on the future 3-D models. The main application to focus will be the medicine and mining.

For medicine, the application to focus will be the surgeon robotics. Nowadays, the surgeon robots are teleoperated through haptic devices and specialist surgeons require many training hours to properly operate it. The motion capture through computer vision to teleoperate a robot will reduce the training time, due the surgeons will need only move their hands as they normally do at surgeon procedure, the haptic devices are not going to be required anymore.

For mining, the research will be focus to eliminate the risk to get injured or dead at work or explore a mine. The labor will teleoperate a robot from a safe distance, the robot in the mine will mimic the labor's movements. If the mine collapse, explode or has a water leak, the human life will be safe and the only lost will be the robot.

## References

1. **Alquisiris-Quecha, O., Maldonado-Reyes, A. E., Morales, E. F., Sucar, L. E. (2018).** Teleoperation and control of a humanoid robot NAO through body gestures. *Seventeenth Mexican International Conference on Artificial Intelligence, IEEE*, pp. 88–94. DOI: 10.1109/micai46078.2018.00022.
2. **Cruz-Silva, J. E., Montiel-Pérez, J. Y., Sossa-Azuela, H. (2019).** 3-D human body posture reconstruction by computer vision. *Advances in Soft Computing, Springer International Publishing, Vol. 11835*, pp. 579–588. DOI: 10.1007/978-3-030-33749-0\_46.
3. **Elbeleidy, S., Jadhav, A., Liu, D., Williams, T. (2022).** Robot teleoperation interfaces for customized therapy for autistic children. *17th ACM/IEEE International Conference on Human-Robot Interaction, IEEE*, pp. 750–753. DOI: 10.1109/hri53351.2022.9889593.
4. **Elbeleidy, S., Rosen, D., Liu, D., Shick, A., Williams, T. (2021).** Analyzing teleoperation interface usage of robots in therapy for children with autism. *Interaction Design and Children, ACM*, pp. 112–118. DOI: 10.1145/3459990.3460715.
5. **Elbeleidy, S., Shick, A., Williams, T. (2021).** Teleoperation interface usage in robot-assisted childhood ASD therapy. *Companion of the ACM/IEEE International Conference on Human-Robot Interaction, ACM*, pp. 162–166. DOI: 10.1145/3434074.3447151.
6. **Europe, S. R., .** Softbank robotics - NAOqi documentation center.
7. **Faller, A., Schuenke, M., Schunke, G. (2011).** *The human body: An introduction to structure and function.* Thieme.
8. **Huang, B., Timmons, N. G., Li, Q. (2020).** Augmented reality with multi-view merging for robot teleoperation. *Companion of the ACM/IEEE International Conference on Human-Robot Interaction, ACM*, pp. 260–262. DOI: 10.1145/3371382.3378336.
9. **Kaushik, R., Simmons, R. (2021).** Perception of emotion in torso and arm movements on humanoid robot quori. *Companion of the 2021 ACM/IEEE International Conference on Human-Robot Interaction, IEEE*, pp. 62–66. DOI: 10.1145/3434074.3447129.
10. **Lin, T. C., Krishnan, A. U., Li, Z. (2020).** Shared autonomous interface for reducing physical effort in robot teleoperation via human motion mapping. *IEEE International Conference on Robotics and Automation, IEEE*. DOI: 10.1109/icra40945.2020.9197220.
11. **M., L. N., Dajles, D., Siles, F. (2018).** Teleoperation of a humanoid robot using an optical motion capture system. *IEEE International Work Conference on Bioinspired Intelligence (IWobi), IEEE*, pp. 1–8. DOI: 10.1109/iwobi.2018.8464136.
12. **Milstein, A., Alyagon, L., Nisky, I. (2021).** Grip force control during virtual interaction with deformable and rigid objects via a haptic gripper. *IEEE Transactions on Haptics, Vol. 14, No. 3*, pp. 564–576. DOI: 10.1109/toh.2021.3060507.
13. **Nordin, M., Frankel, V. H., Forssen, K. (2004).** *Biomecánica básica del sistema musculoesquelético.* McGraw-Hill Interamericana de España.
14. **Puruhita, S. T., Fajar-Satria, N., Hanafi, N., Kusumawati, E. (2020).** Development of teleoperation robot hand mimicking through human hand movement. *International Electronics Symposium*, pp. 277–283. DOI: 10.1109/IES50839.2020.9231584.
15. **Stoddard, B., Cravetz, M., Player, T., Knight, H. (2022).** A haptic multimodal interface with abstract controls for semi-autonomous manipulation. *Proceedings of the 17th ACM/IEEE International Conference on Human-Robot Interaction, IEEE*, pp. 1206–1207. DOI: 10.1109/hri53351.2022.9889349.

16. **Torielli, D., Muratore, L., Laurenzi, A., Tsagarakis, N. (2022).** TelePhysicalOperation: Remote robot control based on a virtual 'marionette' type interaction interface. *IEEE Robotics and Automation Letters*, Vol. 7, No. 2, pp. 2479–2486. DOI: 10.1109/lra.2022.3144792.
17. **Wallace, D., Hang-He, Y., Chagas-Vaz, J., Georgescu, L., Oh, P. Y. (2020).** Multimodal teleoperation of heterogeneous robots within a construction environment. *IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE*, pp. 2698–2705. DOI: 10.1109/iros45743.2020.9340688.
18. **Wang, Y., Zhang, X., Li, K., Wang, J., Chen, X. (2020).** Humanoid robot control system based on AR-SSVEP. *Proceedings of the 2020 6th International Conference on Computing and Artificial Intelligence, Association for Computing Machinery*, pp. 529–533. DOI: 10.1145/3404555.3404625.
19. **Yazdani, A., Novin, R. S. (2021).** Posture estimation and optimization in ergonomically intelligent teleoperation systems. *Companion of the ACM/IEEE International Conference on Human-Robot Interaction, ACM*, pp. 604–606. DOI: 10.1145/3434074.3446350.
20. **Zuher, F., Romero, R. (2012).** Recognition of human motions for imitation and control of a humanoid robot. *Brazilian Robotics Symposium and Latin American Robotics Symposium, IEEE*, pp. 190–195. DOI: 10.1109/sbr-lars.2012.38.

*Article received on 17/04/2023; accepted on 22/09/2023.  
Corresponding author is J. Yajá Montiel-Pérez.*