Human Assistive Robotic System

Futoshi Kobayashi

Kobe University

Outline

1. Human Assistive Robotic System
2. Measurement of Human Motions

   2.1 Motion Capture with Multiple Depth Sensors

   2.2 Measurement of Human Hand Position and Posture
3. Motion Generation for Handover with Hand/Arm Robot
4. Summary

Human Assistive Robot

Humanoid robots are developed in order to assist and help humans in their daily activities.

Cooperative works with humans are divided into

- ✓ confrontive assistance which the robot shares works with humans, and
- ✓ indirect assistance which the robot supports humans.

Requirements for Human Assist

In the human assistive robotic system, the system has to make assisted motions without

- ✓ not only hurting a human,
- ✓ but also disturbing human's works.

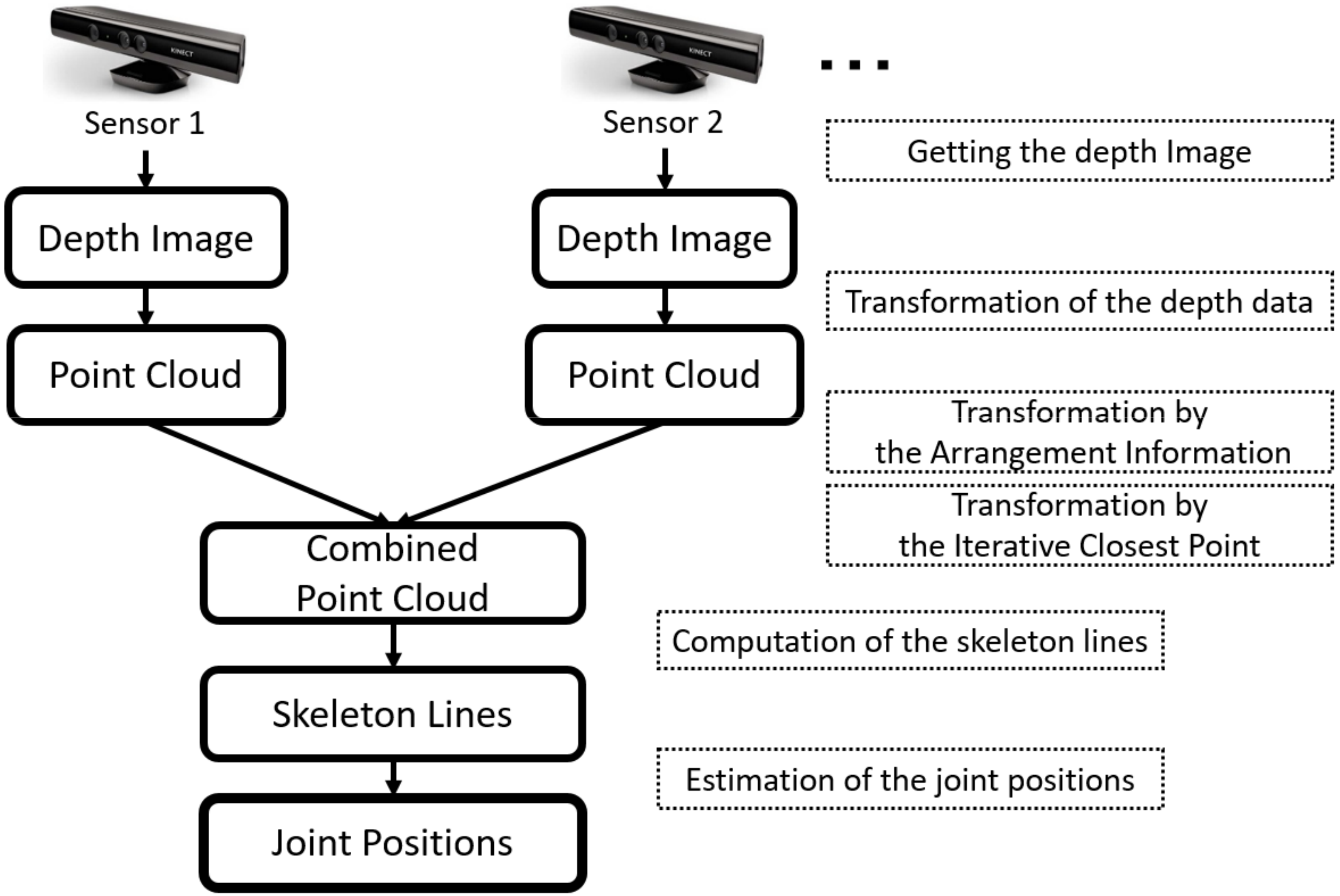
For satisfying these requirements, the human assistive robotic system carry out

- measurement of human motions,
- and motion generation according to human motions.

Measurements of Human Motions

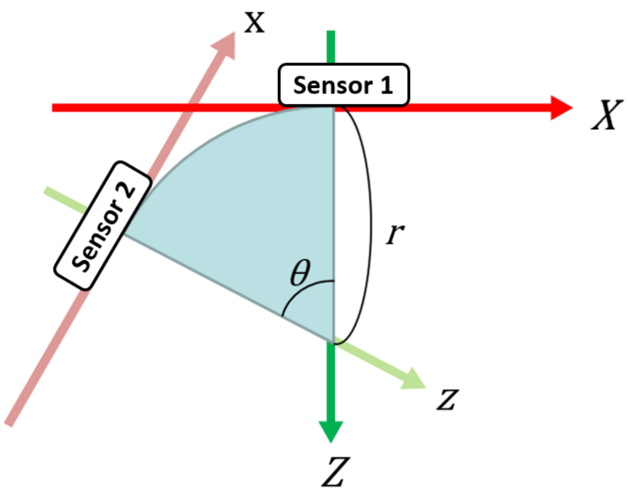
- ✓ Motion Capture with Multiple Depth Sensors
- ✓ Measurement of Human Hand Position and Posture

Motion Capture System with Multiple Depth Sensors



Coordinate Transformation of Point Cloud with Sensor Arrangements

- ✓ Multiple point clouds have own the local coordinate systems based on multiple depth sensors.
- ✓ To combine these multiple point clouds, these coordinate systems need to be unified to the global coordinate system.
- ✓ To unify the coordinate systems, the system uses the affine transformation around Y axis.



$$\vec{X} = R\vec{x} + \vec{t}$$

$\vec{X}$: a point after transformation
$\vec{x}$: a point before transformation
R: rotation matrix
$\vec{t}$: translation vector

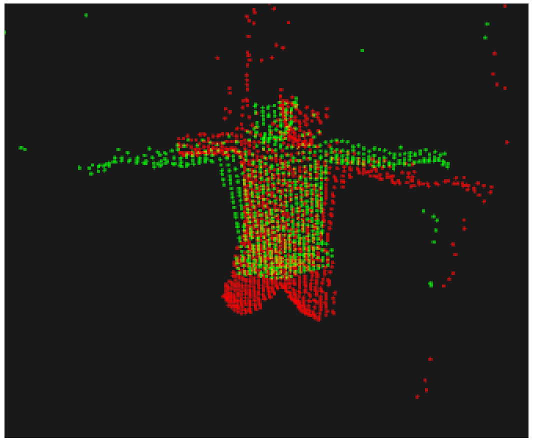
$$R(\theta) = \begin{bmatrix} cos\theta & 0 & -sin\theta \\ 0 & 1 & 0 \\ sin\theta & 0 & cos\theta \end{bmatrix} \quad \vec{t} = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} = \begin{bmatrix} -r\sin\theta \\ 0 \\ r - r\cos\theta \end{bmatrix}$$

Combination of Point Clouds with Iterative Closest Point

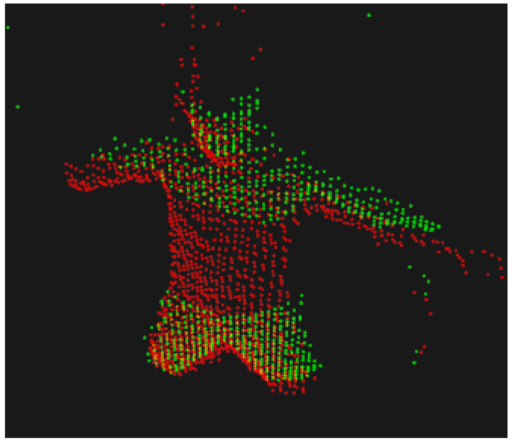
After transforming the point cloud from the depth sensor by the sensor arrangements, the system combines the point clouds by using the ICP.

The ICP algorithm is implemented as the following process.

1. Compute the closest points.
2. Estimate the transformation matrix.
3. Apply the transformation.
4. Judge the terminal condition.



Registration before using the ICP



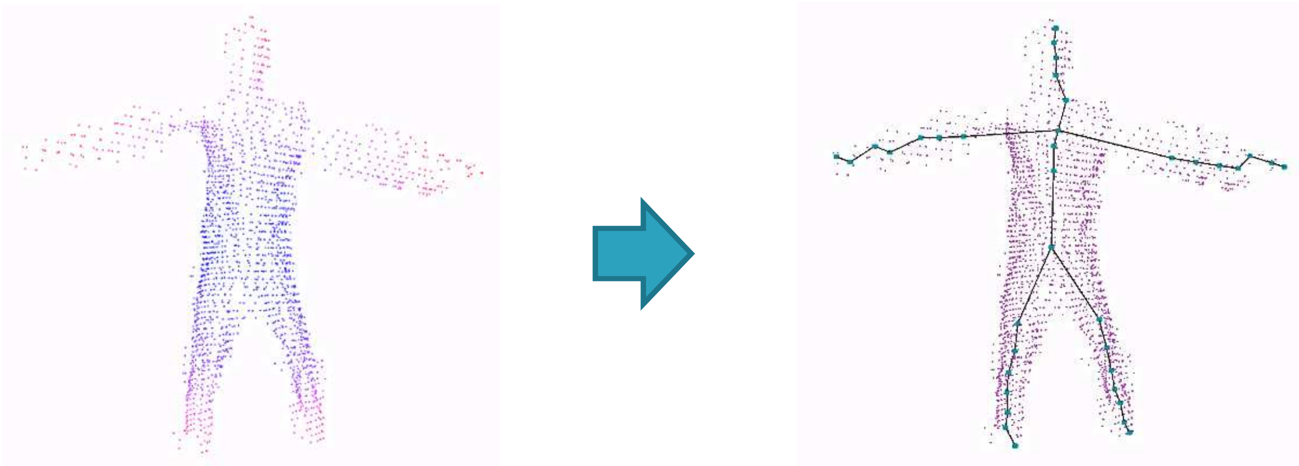
Registration by using the ICP

Computation of Human Skeleton Lines

To estimate the human joint positions, firstly, the system computes skeleton lines from the combined point cloud by using the Reeb graph.

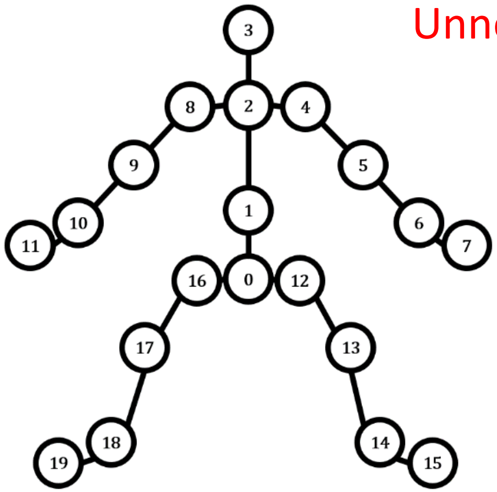
The flow of estimating skeleton lines.

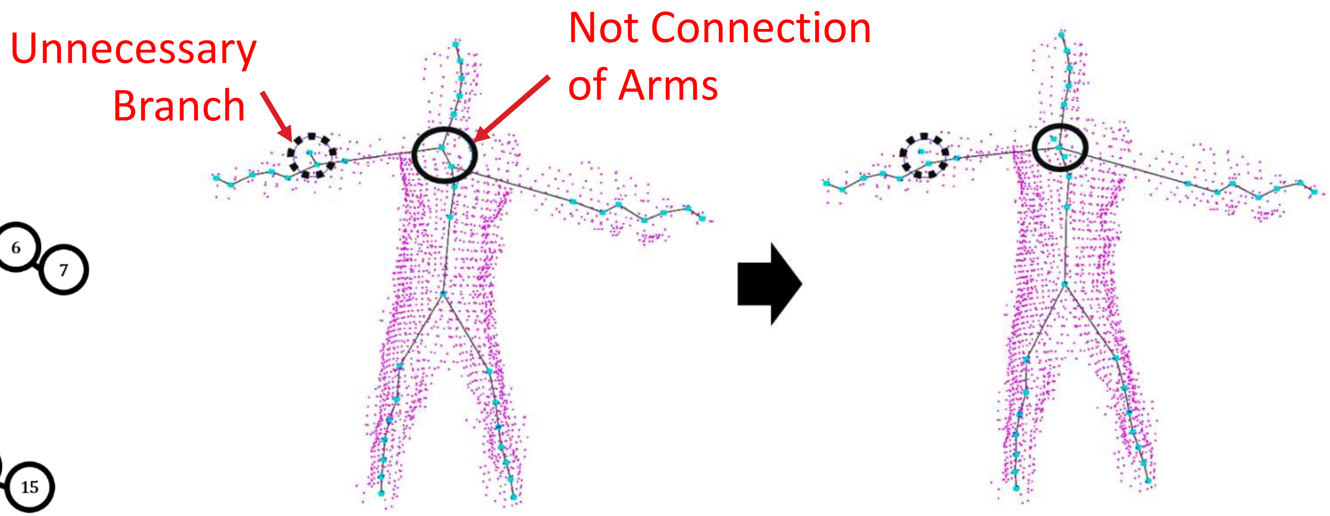
1. The system divides the combined point cloud according to the distance between points in the combined point cloud.
2. The system sets the gravity points in the divided regions as the nodes of the Reeb graph.
3. The system uses the generated Reeb graph as the skeleton lines.



Modification of Human Skeleton Lines

- The system estimates the human joint positions from the node of characteristic structure.
- In the system, the human joint positions are defined as the main 20 positions (refer to Kinect Skeletal Tracking).
- At first, the system modifies the skeleton lines by referring to the human joint structure in the following two cases.

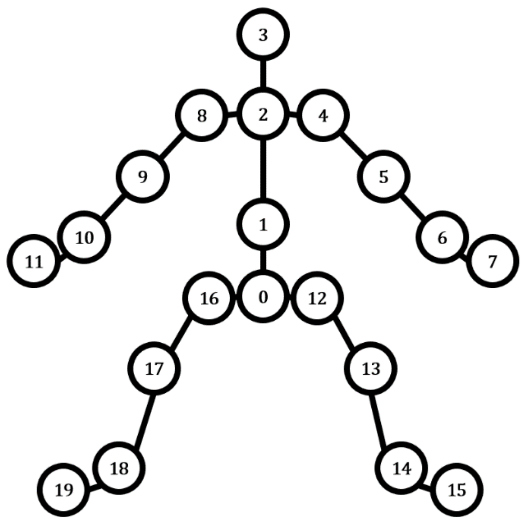


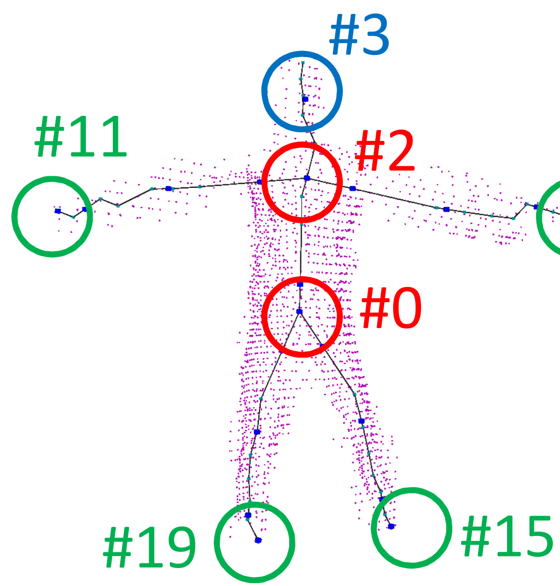


Estimation of Human Joint Positions

Then, the system estimates the human joint positions from the modified skeleton lines.

1. The node with 3 or 4 connections (#0, #2).
2. The gravity point of head direction (#3).
3. The terminal node of arm or leg direction (#7, #11, #15, #19).
4. The position of divided skeleton lines with the specific ratios (#1, #4-#6, #8-#10, #12-#14, 16-#18).

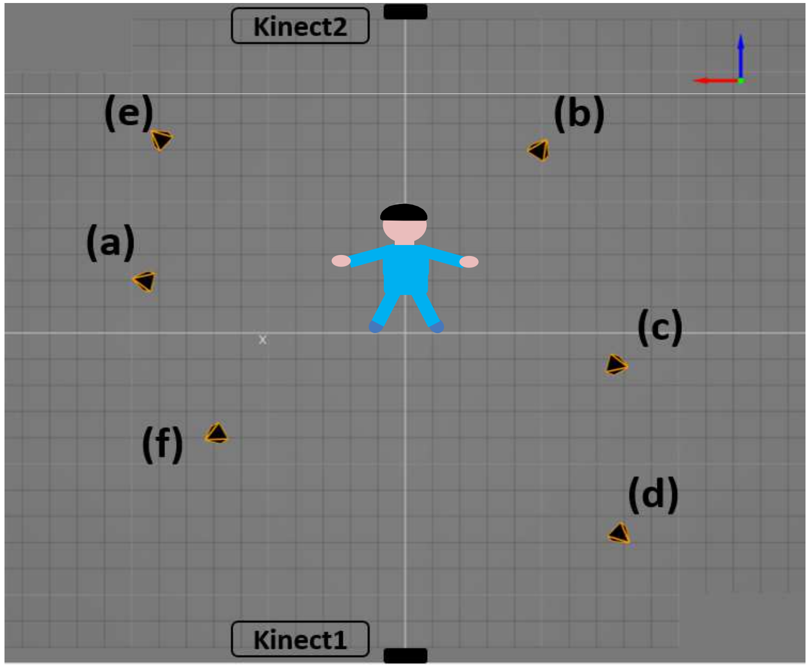


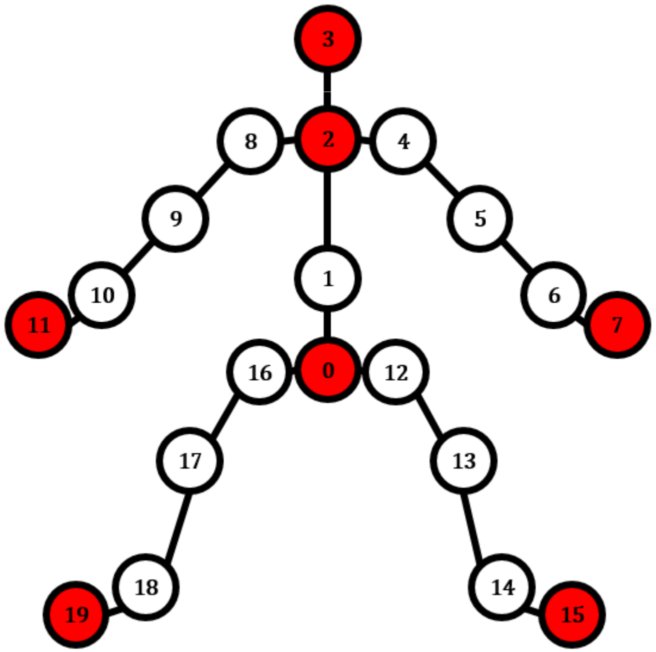


Experiment for Estimating Human Joint Positions

Environment of the Experiment

- ✓ We measured from 4 angles of 0, 30, 60, 90 degrees when the object is facing to the Kinect 1 in 0 degrees
- ✓ We measured 7 positions on 20 frames and compare these average positions of 20 frames.





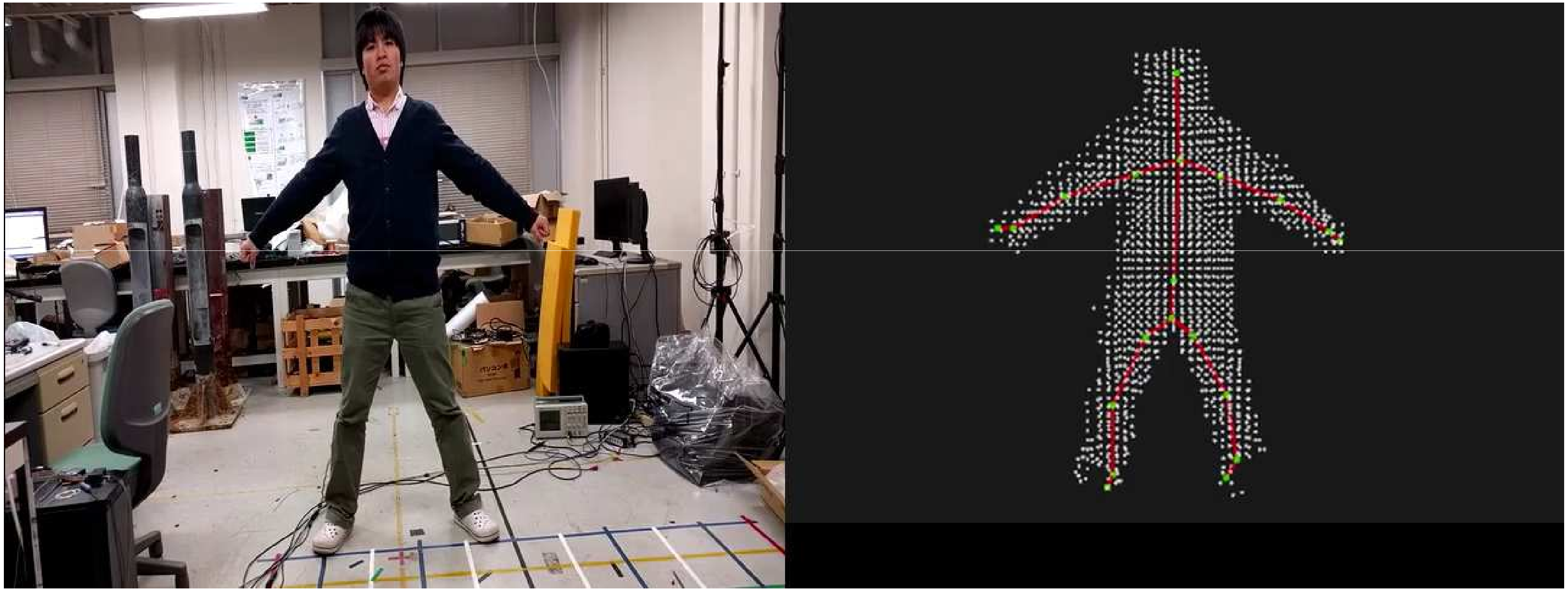
Position Errors According to Sensor Arrangement (1)

	0°		30°	
	Kinect Skeletal Tracking[m]	Proposed method[m]	Kinect Skeletal Tracking[m]	Proposed method[m]
HipCenter (#0)	0.281	0.201	0.272	0.165
ShoulderCenter (#2)	0.326	0.372	0.195	0.179
Head (#3)	0.174	0.219	0.187	0.209
HandLeft (#7)	0.121	0.166	0.097	0.142
HandRight (#11)	0.134	0.154	0.202	0.203
FootLeft (#15)	0.143	0.118	0.160	0.194
FootRight (#19)	0.193	0.189	0.125	0.158

Position Errors According to Sensor Arrangement (2)

	60°		90°		
	Kinect Skeletal Tracking[m]	Proposed method[m]	Kinect Skeletal Tracking[m]		Proposed method[m]
HipCenter (#0)	0.279	0.139	0.222		0.107
ShoulderCenter (#2)	0.244	0.163	0.277		0.099
Head (#3)	0.208	0.210	0.251		0.194
HandLeft (#7)	0.077	0.147	0.181		0.084
HandRight (#11)	0.154	0.178	0.638	>>	0.103
FootLeft (#15)	0.189	0.169	0.201		0.148
FootRight (#19)	0.151	0.218	0.597	>>	0.220

Skelton Tracking



Measurements of Human Motions

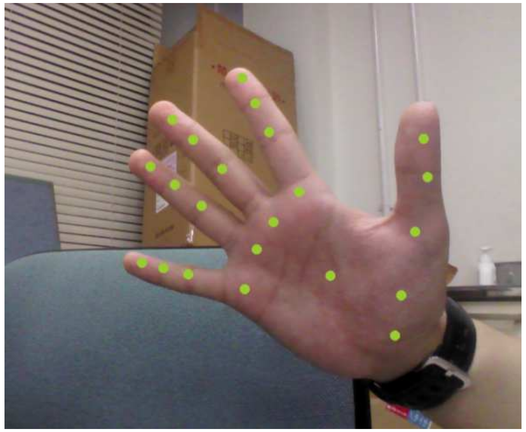
- ✓ Motion Capture with Multiple Depth Sensors
- ✓ Measurement of Human Hand Position and Posture

Measurement of Human Hand Position by RealSense Camera

The RealSense Camera can measure the human hand position.

The RealSense Camera has

- ✓ IR camera for 3-D capture of the human hand, and
- ✓ RGB color camera for confirming the human hand.



Detecting the human hand by the RealSense Controller

RealSense Camera by Intel

Outline

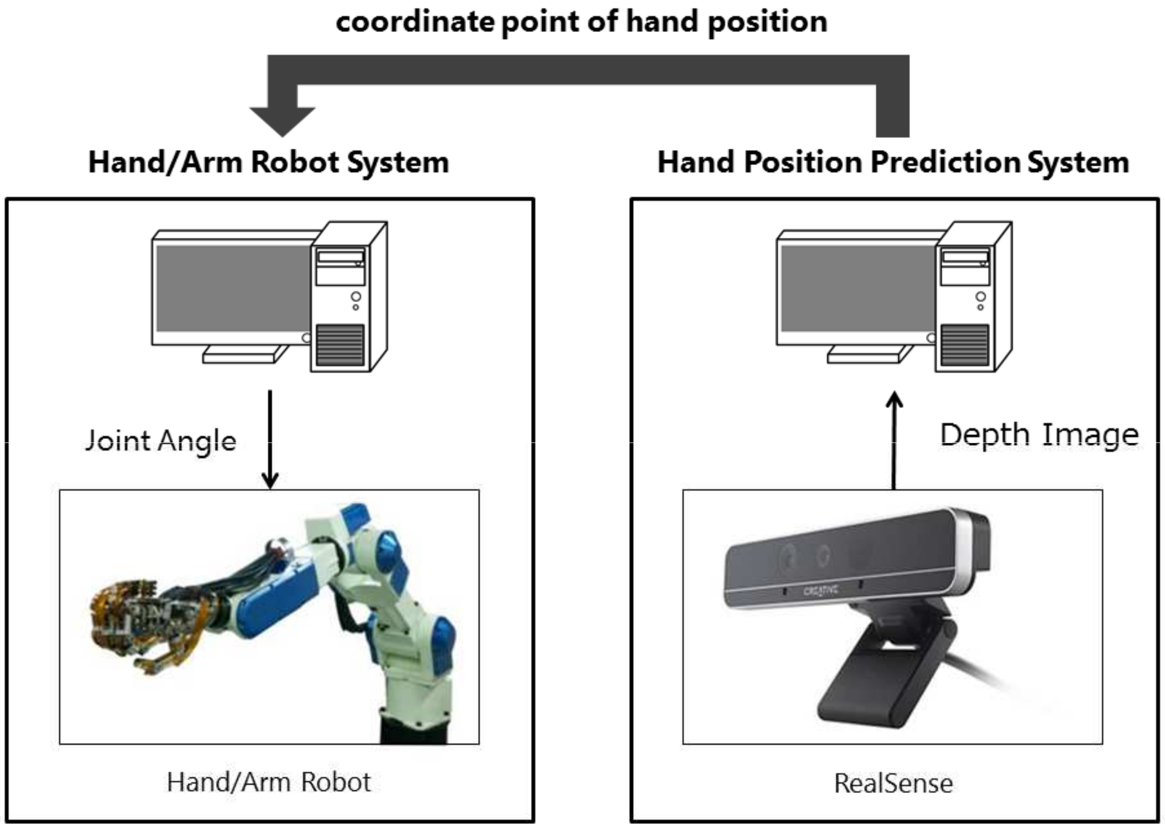
1. Human Assistive Robotic System
2. Measurement of Human Motions

   2.1 Motion Capture with Multiple Depth Sensors

   2.2 Measurement of Human Hand Position and Posture

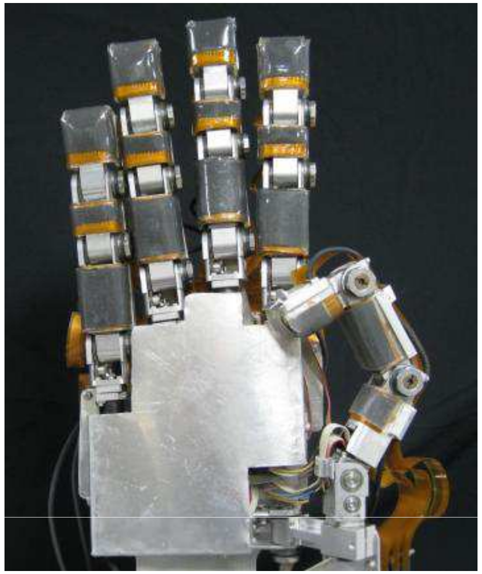
3. Motion Generation for Handover with Hand/Arm Robot
4. Summary

Hand/Arm Robot System with RealSense Camera for Handover



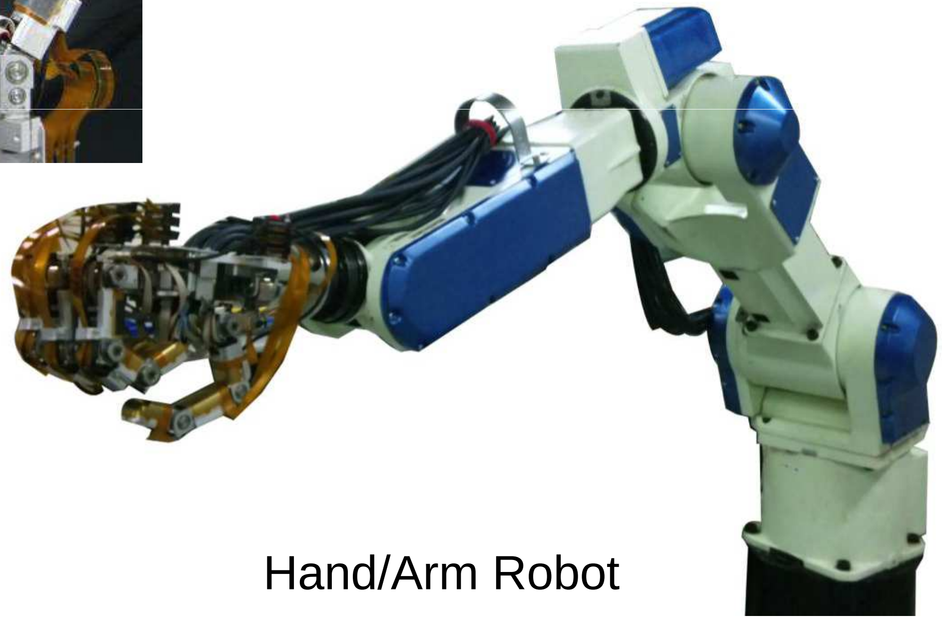
- ✓ The hand/arm robot grips and moves objects.
- ✓ The RealSense Camera measures and predicts the hand position.

Hand/Arm Robot



Universal Robot Hand type II

- 5 Fingers, 20 Joints, 16 Degrees of Freedom
- Connection between DIP and PIP Joints
- 1.6 times height of human hand



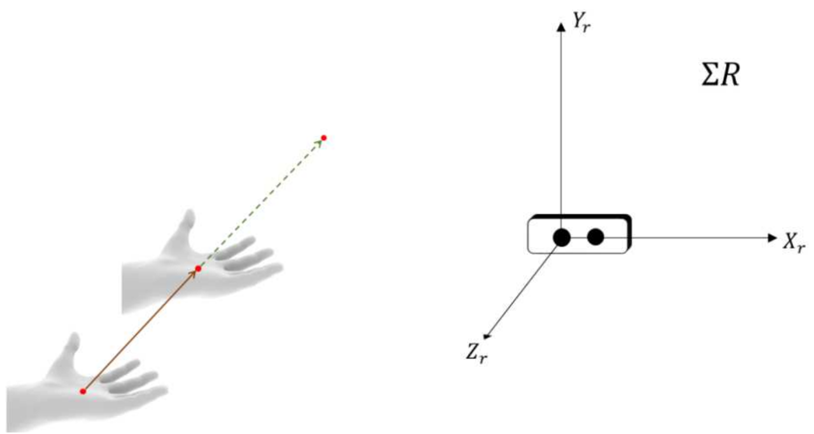
Hand/Arm Robot

6-axis Industrial Manipulator by KHI

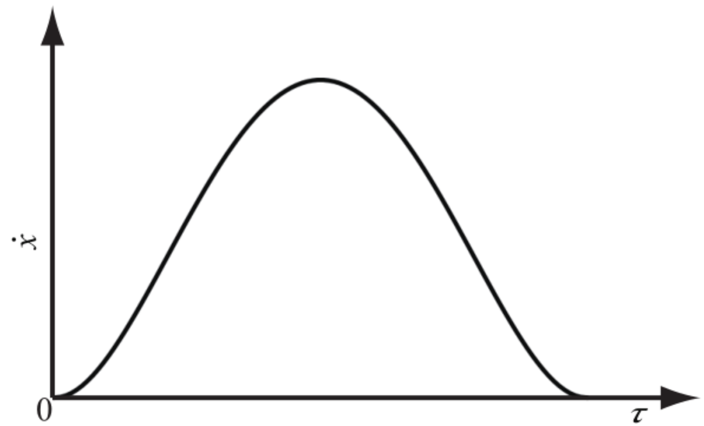
Characteristics of Human Hand Motion

Following facts in the human hand motion is known by previous researches.

- ✓ Hand trajectory in the reaching movement becomes linear,
- ✓ hand velocity in the reaching movement becomes the bell type.



Reaching Movement

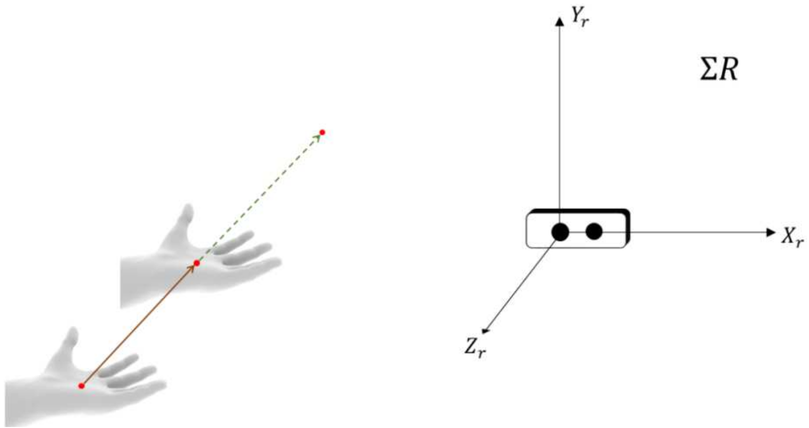


Hand Velocity

Prediction of Direction of Human Hand Trajectory

✓ Hand trajectory in the reaching movement becomes linear.

The direction of human hand trajectory is predicted according to the movement direction of each frame.

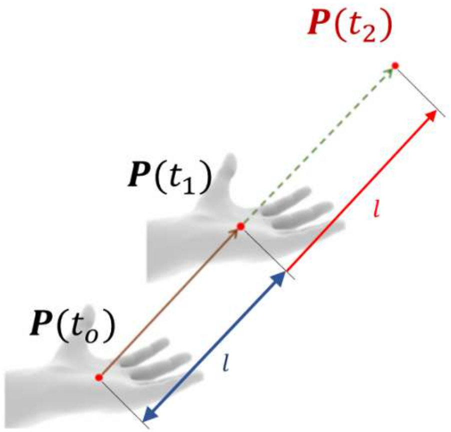


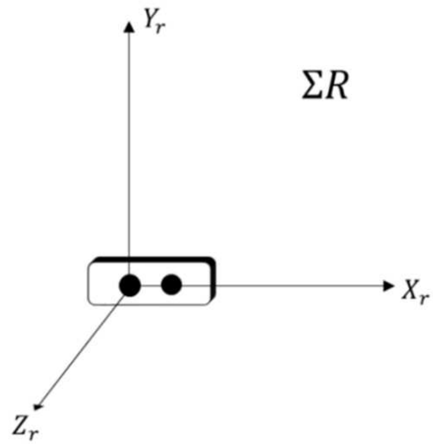
Reaching Movement

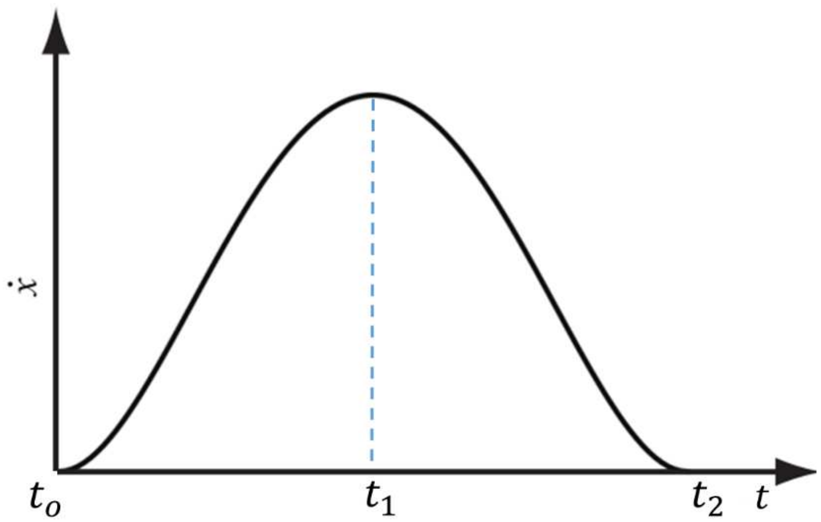
Prediction of Direction of Human Hand Trajectory

✓ Hand velocity in the reaching movement becomes the bell type.

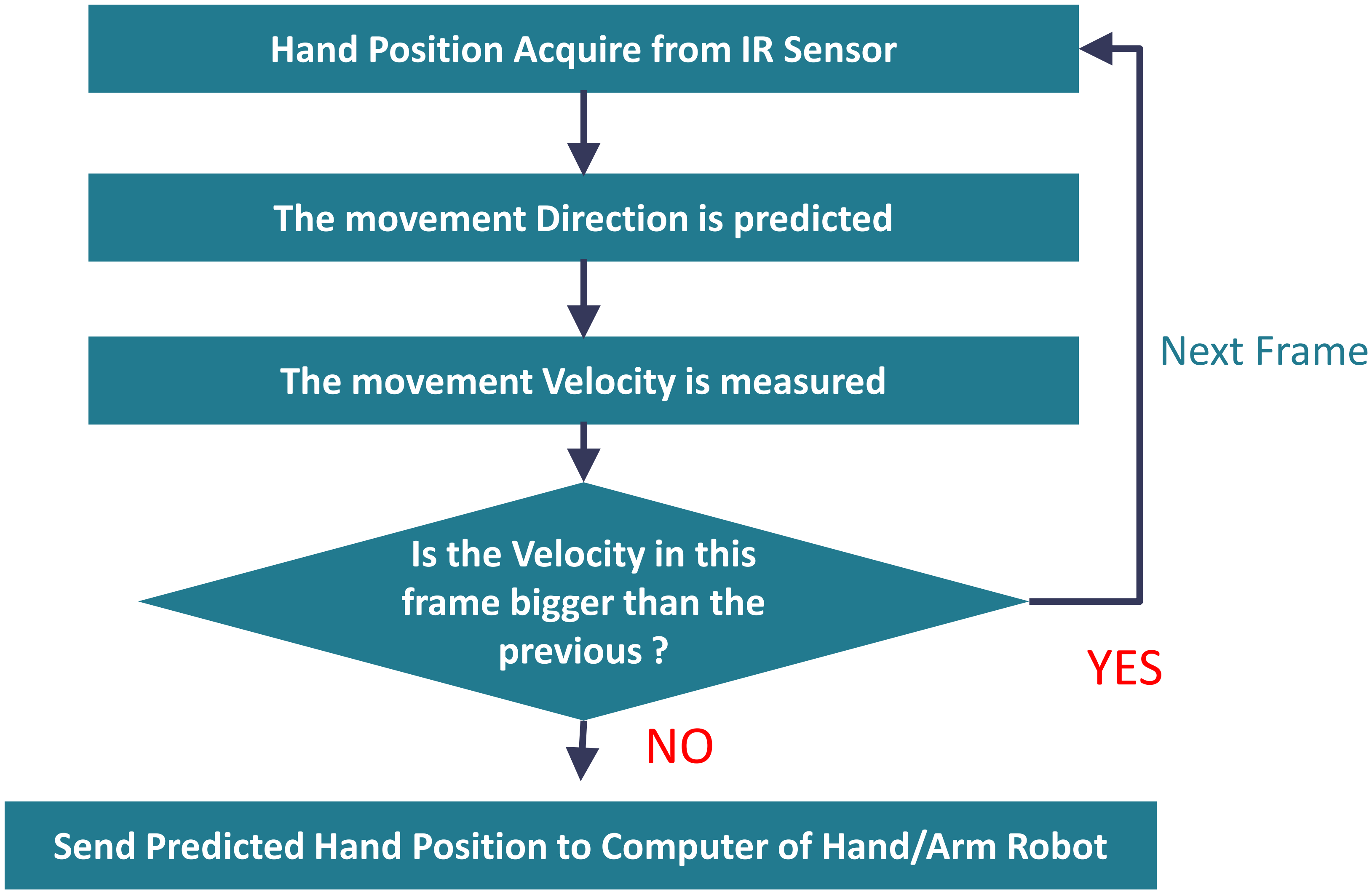
The position of human hand is predicted by doubling the distance from when the hand is started to when the hand speed peaked.





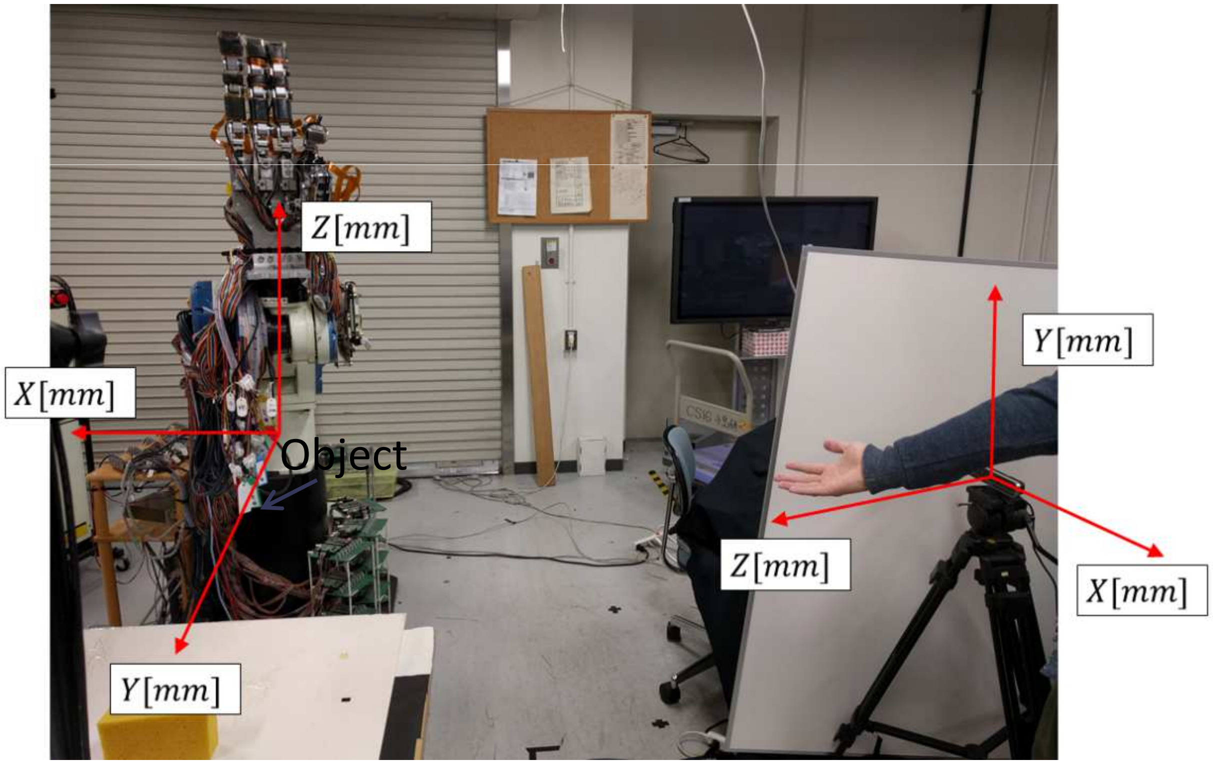


Flowchart of Handover Motion



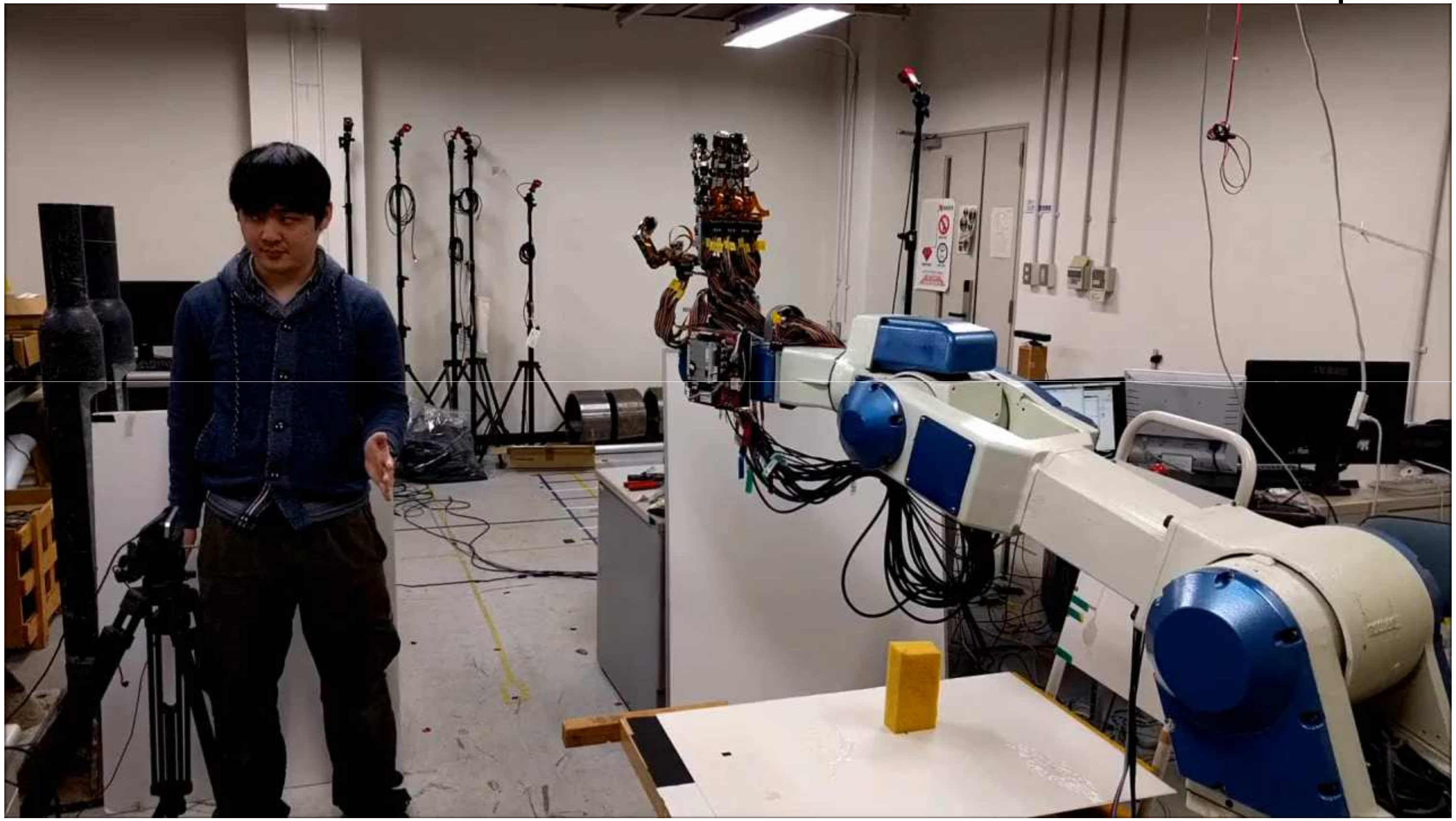
Handover Experiment

The experiments for generating the robot-human handover motion are implemented for showing the effectiveness of the propose method.

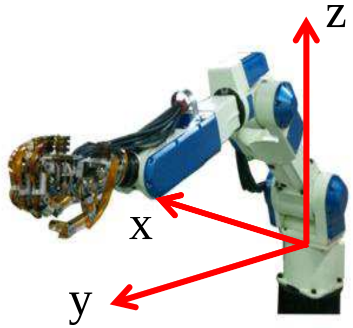


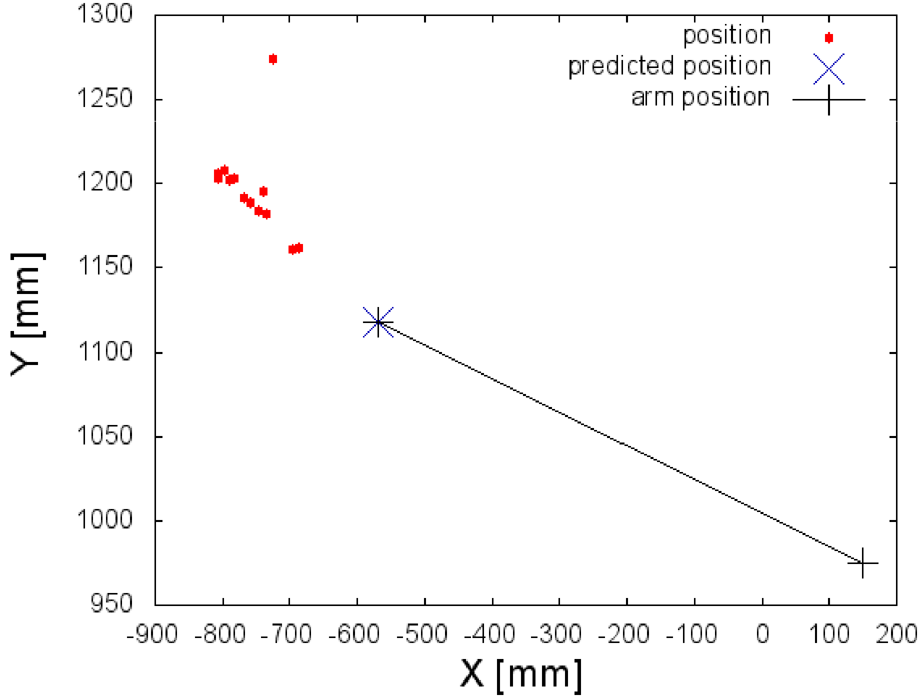
Movie of Handover Experiment

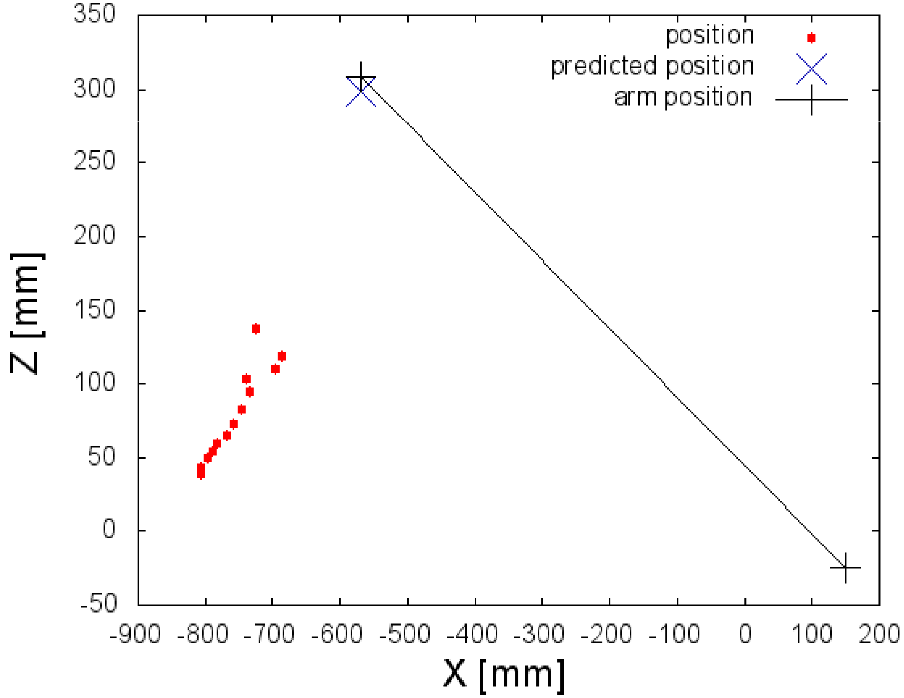
2x speed



Trajectory of Hand/Arm Robot







Summary

Humanoid robots are developed in order to assist and help humans in their daily activities.

Cooperative works with humans are divided into

- ✓ confrontive assistance which the robot shares works with humans, and
- ✓ indirect assistance which the robot supports humans.

Thank you for your kind attention.