IAEA
International Atomic Energy Agency

International Conference on Physical Protection of Nuclear Material and Nuclear Facilities
13–17 November 2017, Vienna, Austria

IAEA-CN-254-111

Vision-based Hand Motion Recognition for Insider Sabotage Detection using Deep Learning

Shi CHEN, Kazuyuki DEMACHI
Department of Nuclear Engineering and Management
School of Engineering
The University of Tokyo

東京大学 THE UNIVERSITY OF TOKYO

nem 原子力国際専攻 Nuclear Engineering and Management

Contents

1. Introduction

2. Hand Motion Capture

3. Behavior Recognition

4. Robustness Verification

5. Conclusion & Future Work

1.1. Significance of Nuclear Security

Increasing threats of terrorism after Fukushima Daiichi Accident

- Increasing attention towards the site of nuclear facilities;
- The important functions of nuclear facilities are opened to public through media and internet.

"What can happen by natural disaster also can be made to happen by human design."

(http://www.mofa.go.jp/mofaj/dns/n_s_ne/page22_000968.html)

1.2. Importance of Insider Sabotage Detection

Sabotage Type

Outsider

Insider

- Two-man rule
- Trustworthiness confirmation
- Physical Protection System (PPS)

Difficulties in Insiders' Sabotage Prevention

- ✓ Deterrence is **invalid** for insider;
- ✓ **Difficult to distinguish** malicious insider from ordinary worker and monitor everyone in nuclear facility;
- ✓ **Difficult to distinguish** sabotage behaviors from maintenance behaviors.

Time Limitation

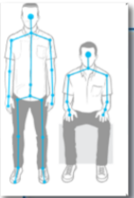
PPS: Deterrence → Detection → Delay → Response

If detection was failed, delay and response would not be activated

Detection technology should be enhanced!

1.3. Proposal of Hand Motion Analysis





1.A. Body Behavior
(Detection of Specific Motion)

1.B. Hand Behavior
(Detection of Specific Motion)

1.C. Tool Tracking
(Malicious or not)

1.D. Hidden Objects
(Disappearance of objects)

- Connecting / Disconnecting a Cable
- Turning on / off a Switch
- Turning a Screw using a Screwdriver
- Inserting a USB Device
- Putting / Taking a Suspicious Object

⋮

- SBO?
- Reactor Out of Control?
- Loss of Safety Function?
- Personal Injury?

⋮

"Could these sabotage motions be distinguished by only body motion analysis?"

1.3. Proposal of Hand Motion Analysis

- ✓ Hand motion analysis is necessary.
- ✓ 3D fingertip position is more essential in order to determine the hand motion.

Wide Range of Applications

- Criminal Detection
- Sign language Communication
- Human-computer Interaction for AR, VR, IoT……

⋮

1.4. Proposal of Time-Series Data Analysis

Challenge of Distinguish Sabotage Behaviors from Maintenance Behaviors

Ordinary Maintenance Behavior

Sabotage

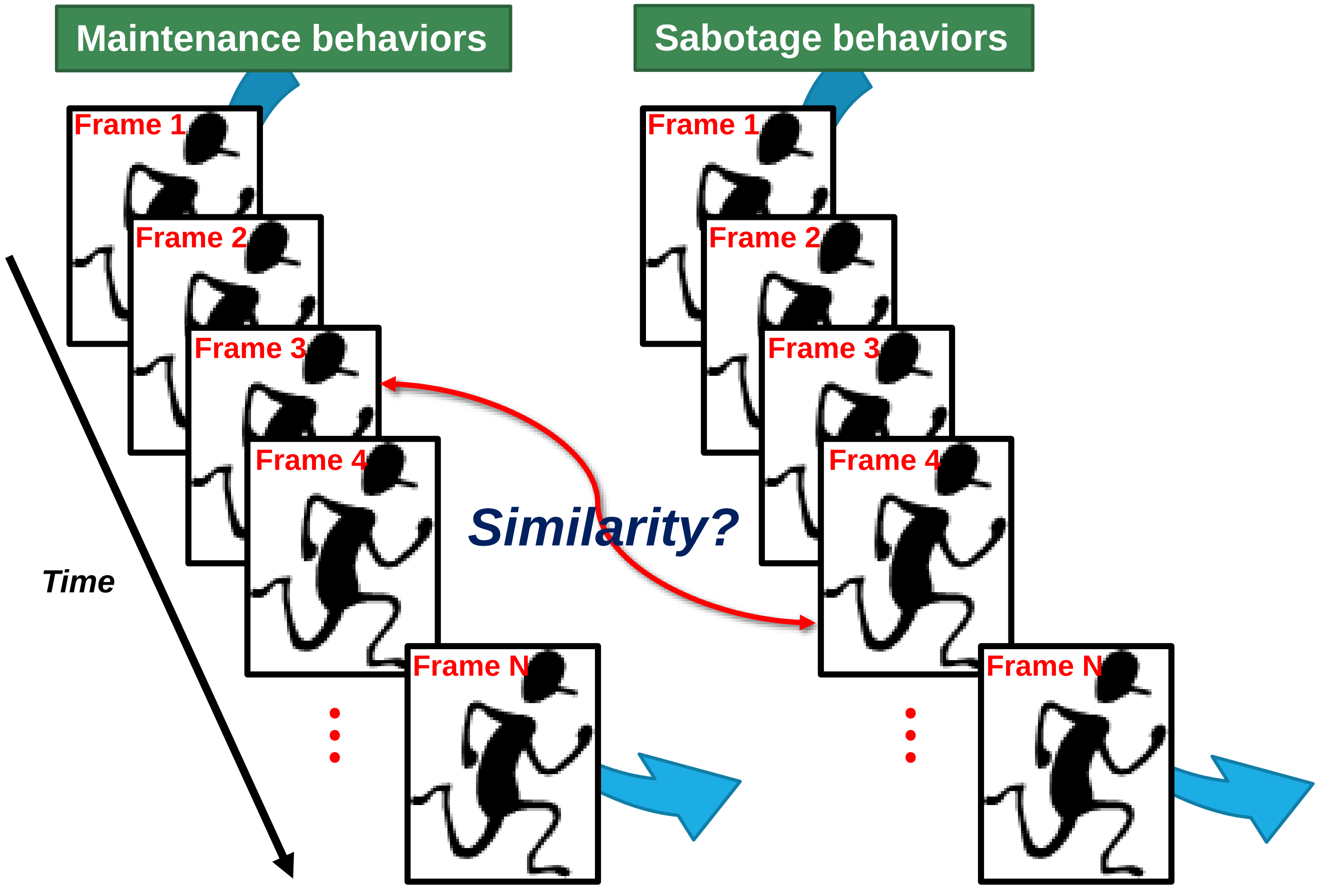
Hidden in Ordinary Maintenance Behavior

Normal behaviors of ordinary maintenance are complicated and diverse

1.4. Proposal of Time-Series Data Analysis

Coventional Research

Static Image Analysis



Time Series Data Analysis

- ✓ ***More scenes;***
- ✓ ***More detail information;***
- ✓ ***Time variation information can be detected;***
- ✓ ***Reduced calculated amount.***

By feature extraction using Deep Learning, data compression can be proceed and calculated amount can be reduced.

Distinguish of malicious behaviors and ordinary maintenance behaviors is possible.

1.5. Research Objectives

Detection of Insiders' Sabotage for Nuclear Security.

Challenge!

To distinguish sabotage behaviors from ordinary maintenance behaviors

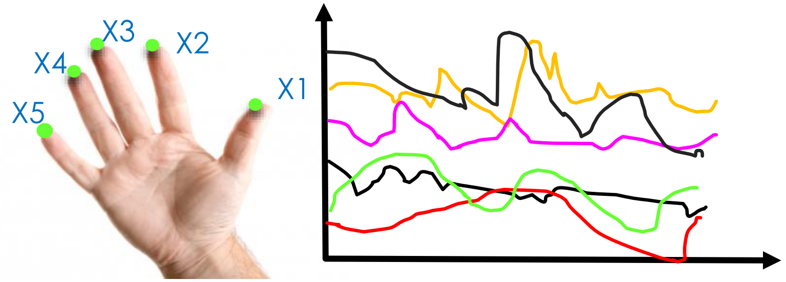
Hand Motion Analysis

Objective1: Hand Motion Capture
- *New Vision-based Algorithm*

Objective2: Behavior Recognition
- *Time-Series Data Analysis*
- *Deep Learning*

Objective3: Robustness Verification

Hand Motion Time Series Data



Contents

1. Introduction

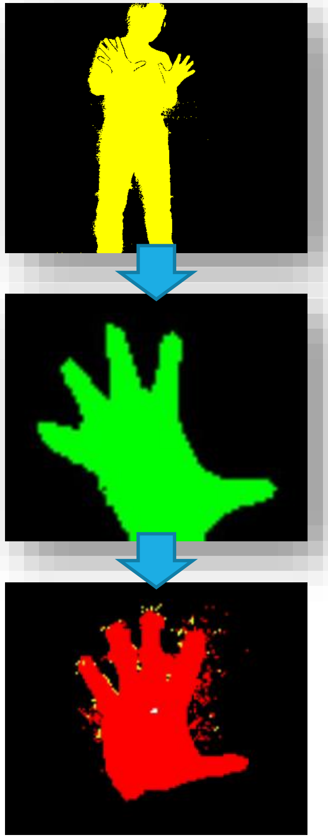
2. Hand Motion Capture

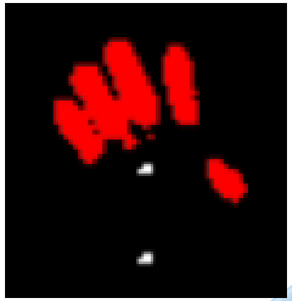
3. Behavior Recognition

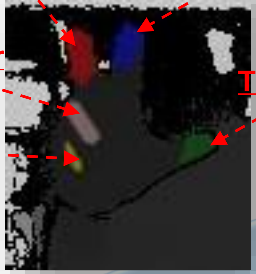
4. Robustness Verification

5. Conclusion & Future Work

2.1. New Algorithm of Hand Motion Capture







Visual Studio C#

Developing hand motion capturing system by Microsoft Visual Studio (coding by C#)

4. Real-time Hand Motion Capture

3. Fingers Identification

Identifying each finger name.

2. Fingers Segmentation

Segmenting fingers from palm.

1. Hand Region Classification

- ✓ Body Segmentation
- ✓ Hand Segmentation
- ✓ Hand Pixels Classification

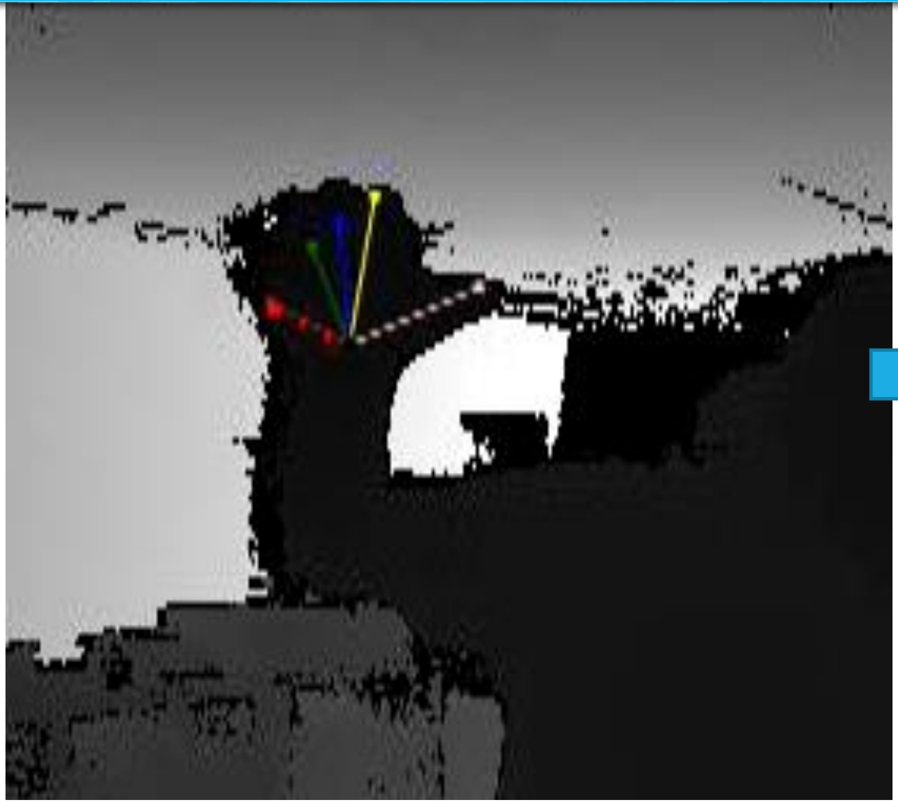
Data Acquisition

Using Kinect v2

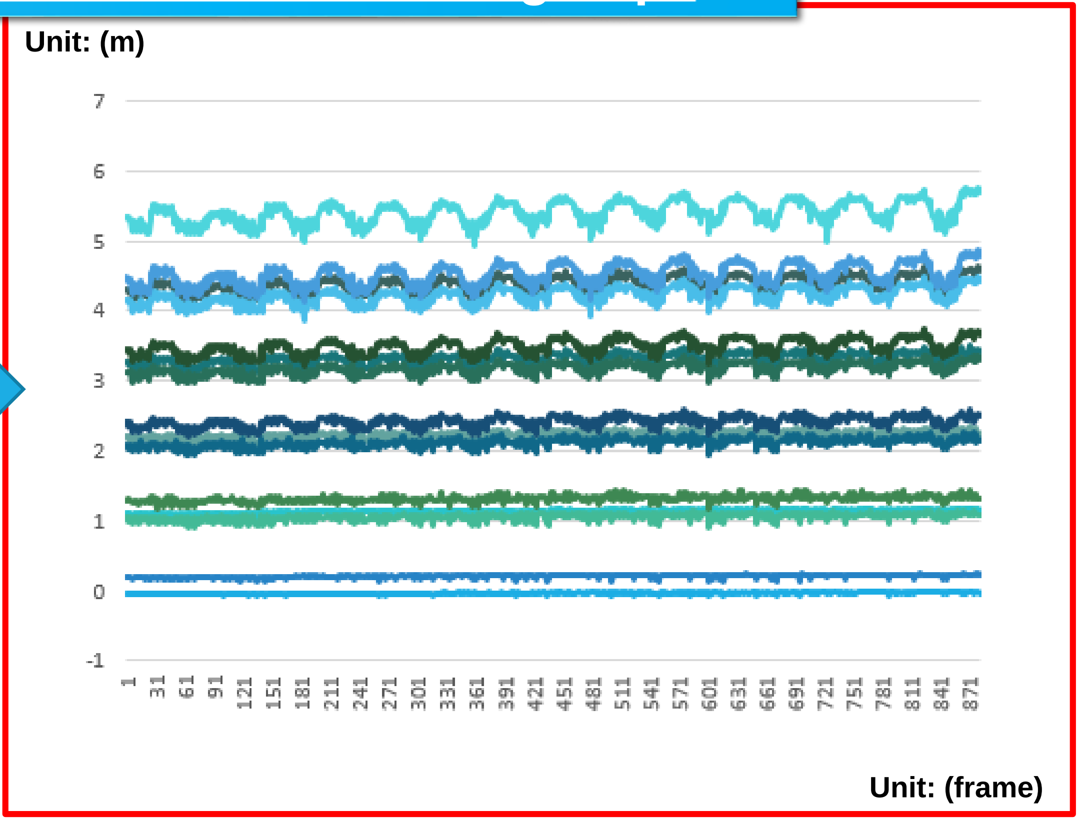
To capture hand motion, development of new algorithm is necessary

2.2. Real-time Hand Motion Capture

Result of Real-time Hand Motion Capture



Time Variation of Fingertips



- ✓ **Positions of each fingertips was successfully obtained;**
- ✓ **The real-time calculating frame rate is about 29.8fps.**

(Shi Chen, Kazuyuki Demachi, Tomoyuki Fujita, Yutaro Nakashima, Yusuke Kawasaki, "Insider Malicious Behaviors Detection and Prediction Technology for Nuclear Security", E-journal of Advanced Maintenance (EJAM), Vol.9, No.1, 2017.)

Contents

1. Introduction

2. Hand Motion Capture

3. Behavior Recognition

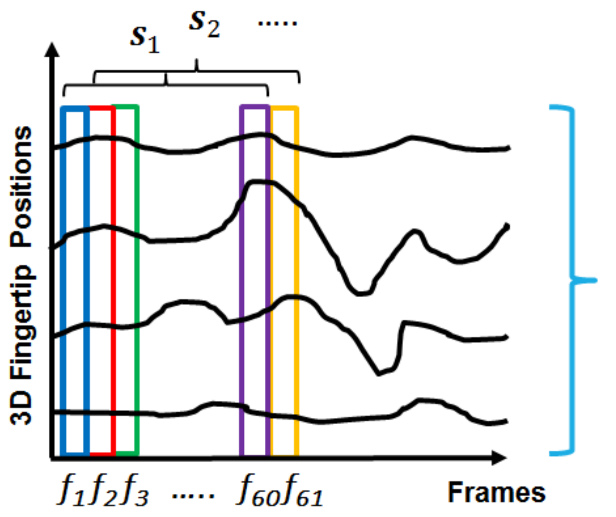
4. Robustness Verification

5. Conclusion & Future Work

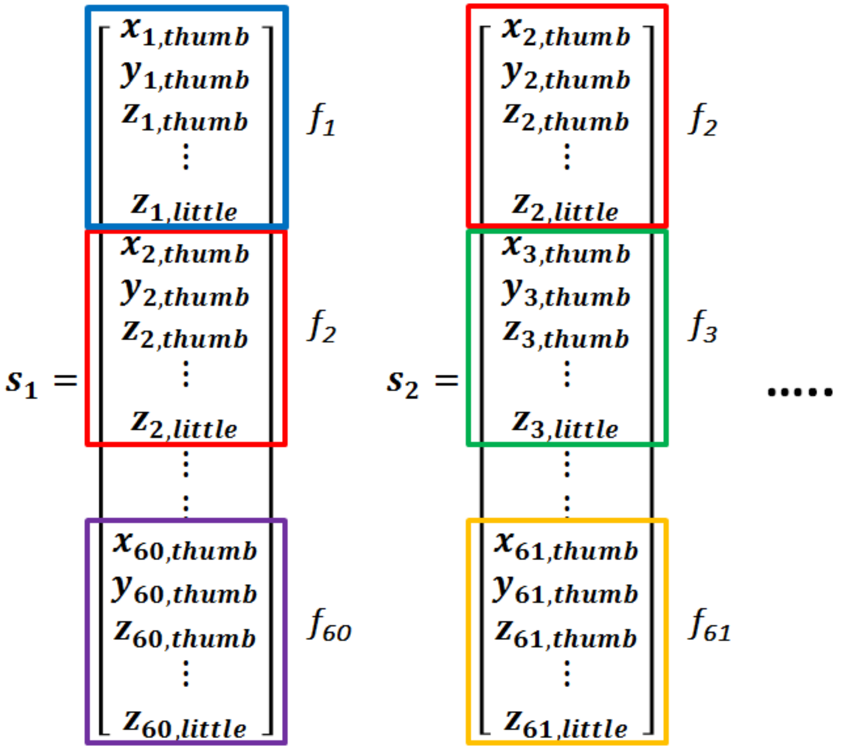
3.1. Time-Series Data Conversion

Assumed Malicious Motions

Motions other than these are considered as ordinary motions



Captured Hand Motion Data



Stacked Auto-encoder Trainset

Captured Hand Motion Data

- *Hand motion captured from 5 person;*
- *Different relative distance and angle to camera.*

3.2. Motion Classification by Deep Learning

Assumed Malicious Motions

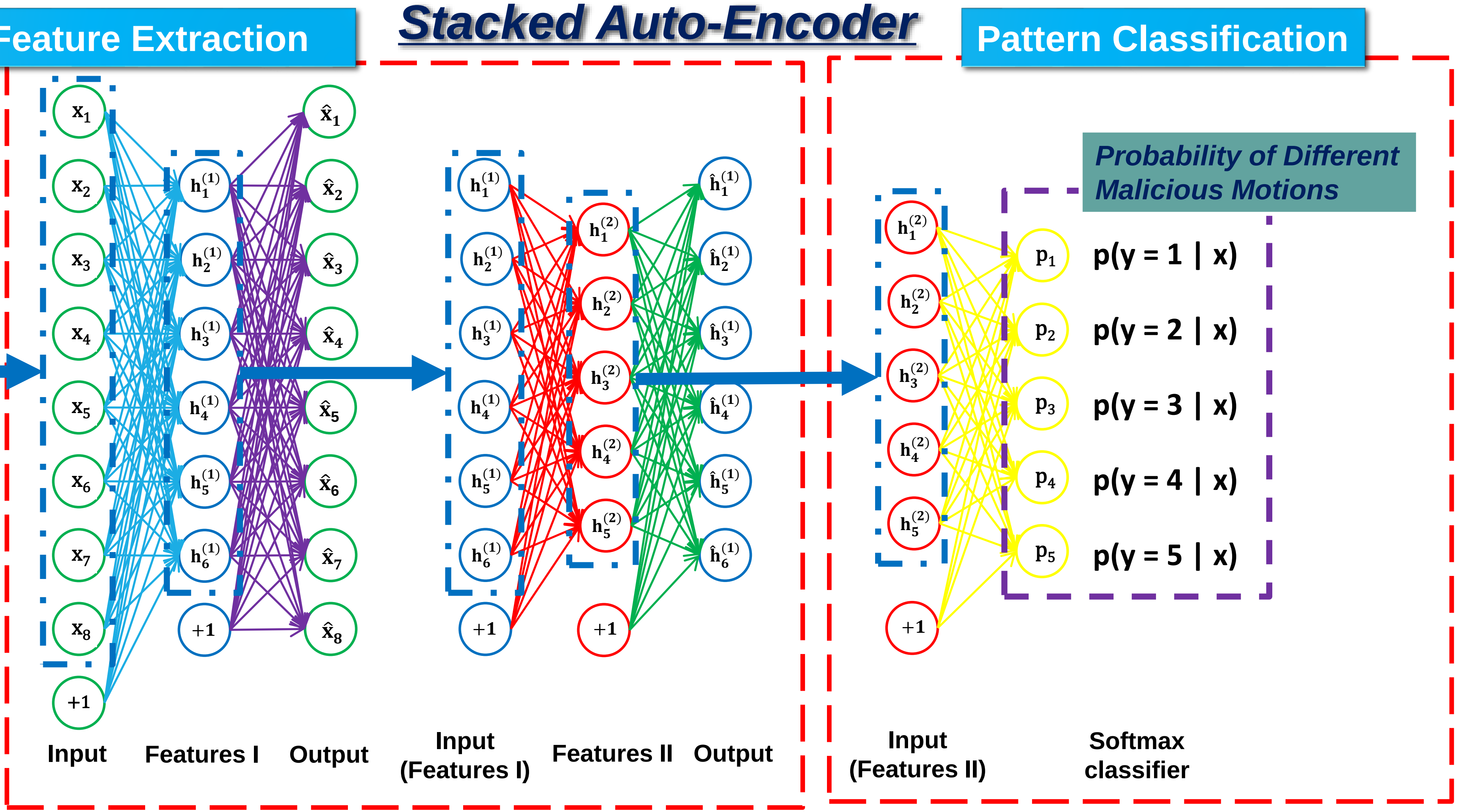
Cutting

Patting

Turning

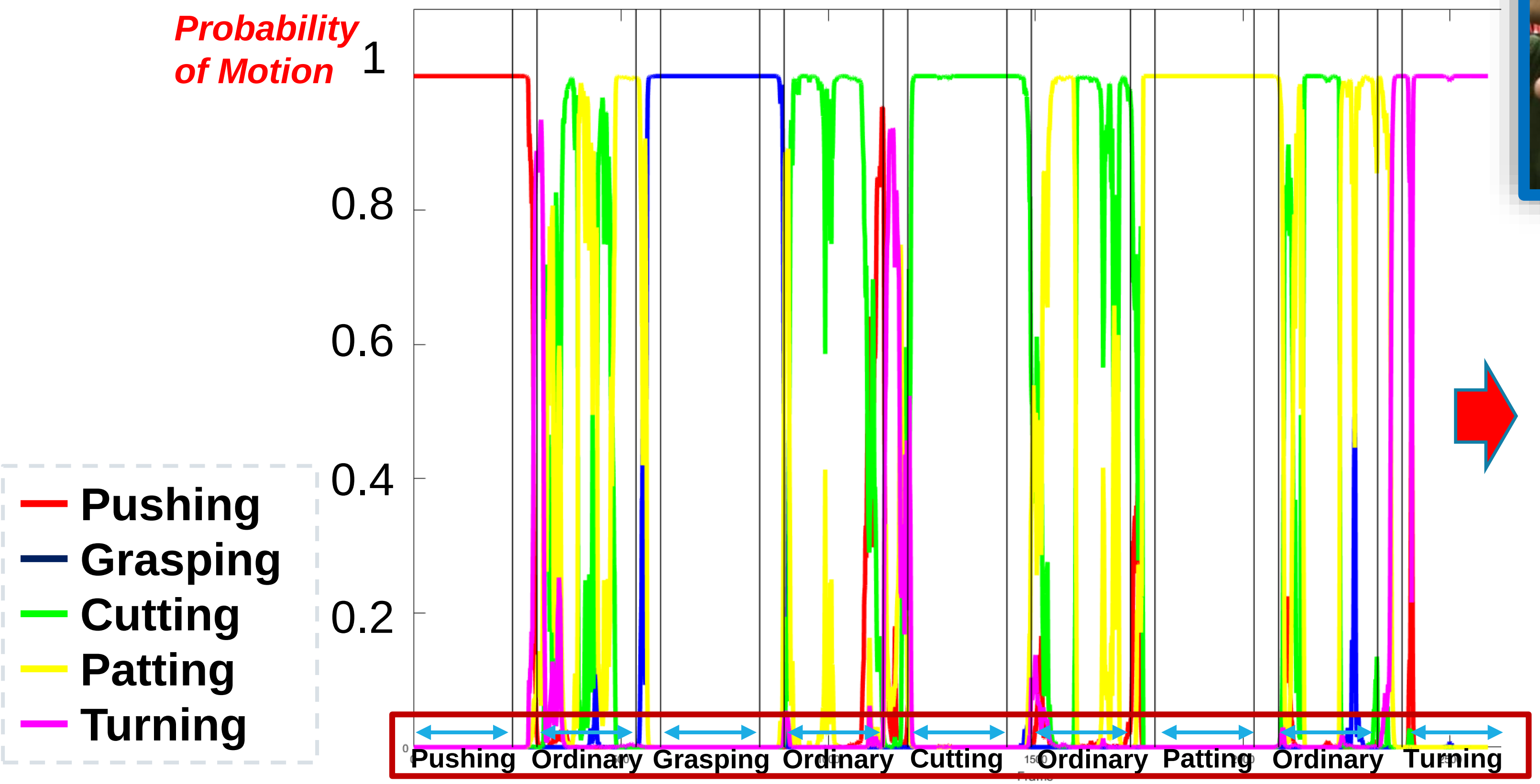
Grasping

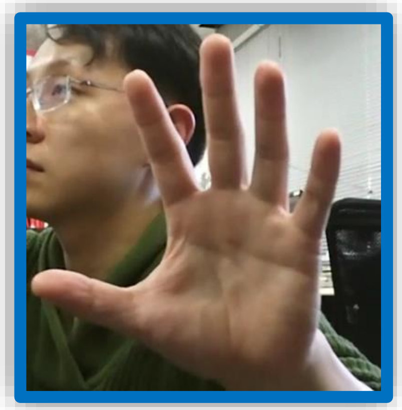
Pushing

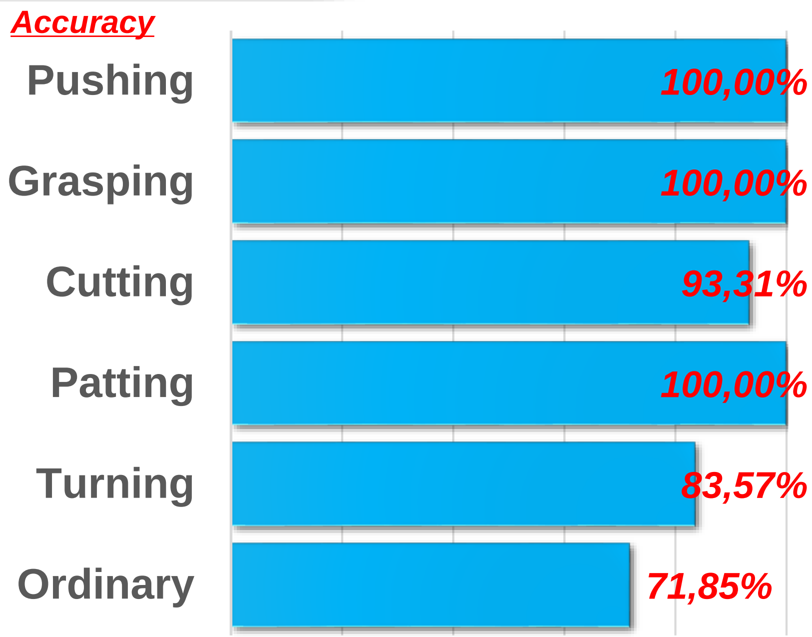


3.3. Behavior Recognition Result

Testing Result







Training Time: ***1217s/43340frame***
Detection Time: ***0.0023s/frame***

Real-time detection is possible.

Overall Accuracy: 82.555%

Contents

1. Introduction

2. Hand Motion Capture

3. Behavior Recognition

4. Robustness Verification

5. Conclusion & Future Work

4. Robustness Verification

Types of data loss problems

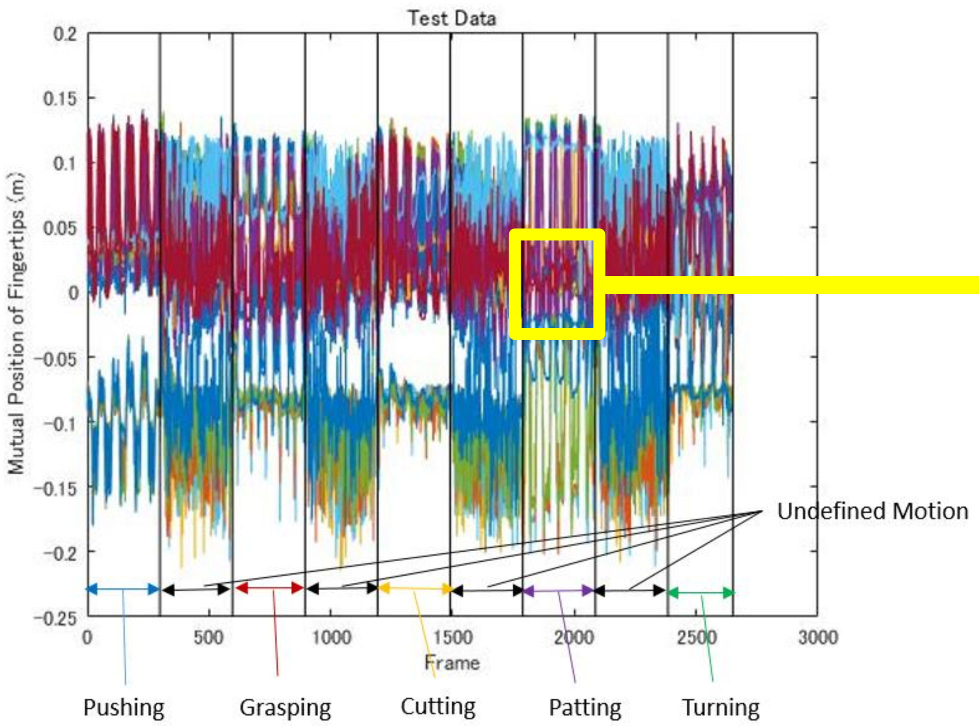
1) Discontinuous frames loss problem

2) Continuous frames loss problem

"Can we still achieve satisfied detection results when some frames of image data are loss?"

4.1. Discontinuous Frames Loss Problem

Verification Method



N continuous frames

……

0.1	0.3	0.4	0.5	0.5
0.2	0.1	0.1	0.3	0.5
0.8	0.8	0.9	1.1	1.2
1.2	0.9	0.8	0.6	0.3
1.6	1.6	1.7	1.8	1.9
1.7	2.1	2.2	2.5	2.5

……

Randomly Selected as data lost frame

Randomly Selection of Lost Data

- ✓ Randomly selecting 1 frame from each 5 continuous frames; → *20% data lost*
- ✓ Randomly selecting 1 frame from each 3 continuous frames; → *33% data lost*
- ✓ Randomly selecting 1 frame from each 2 continuous frames. → *50% data lost*

→ *1000 samples was randomly generated*

4.1. Discontinuous Frames Loss Problem

Verification Results

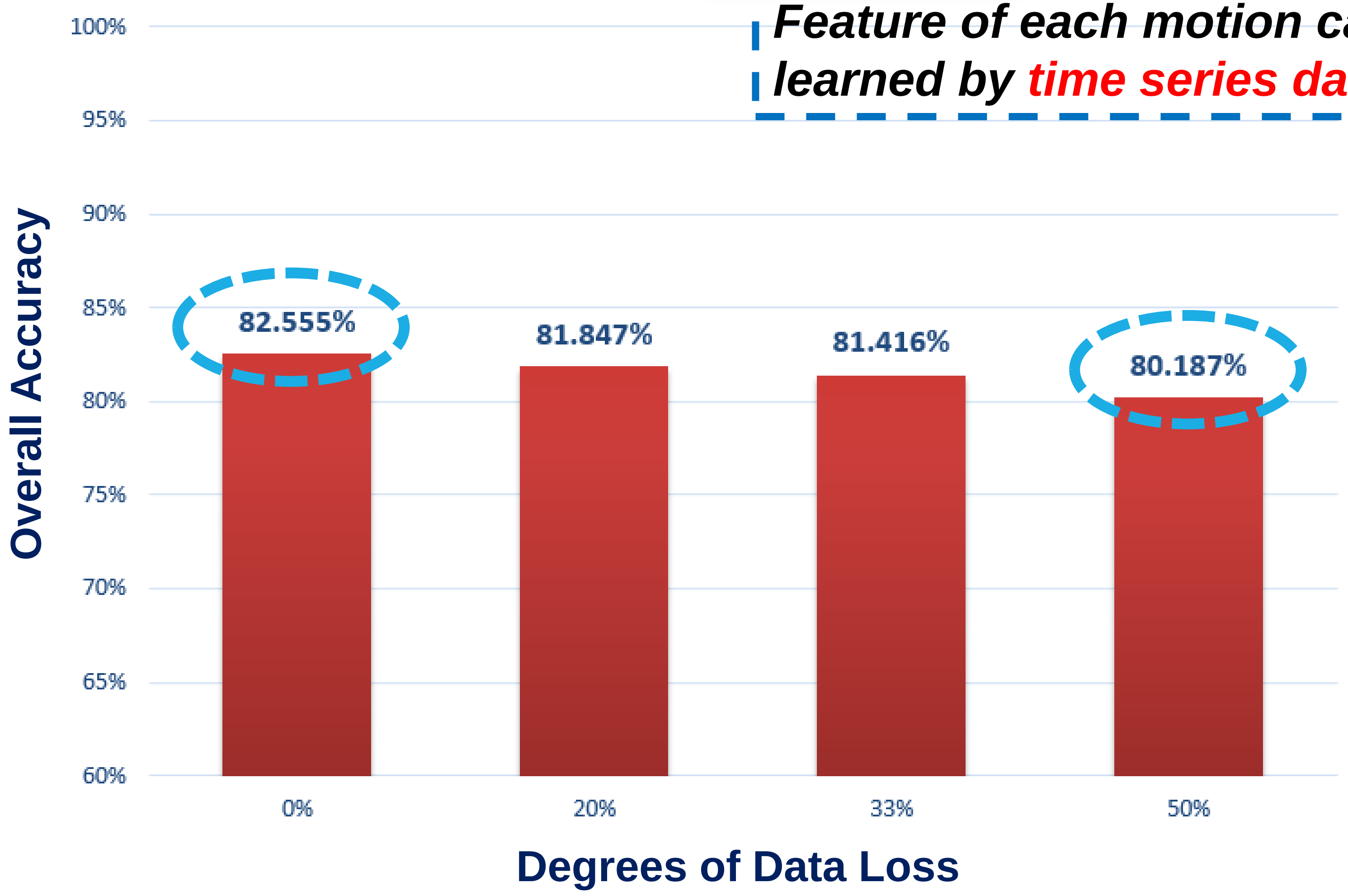
	Pushing	Grasping	Cutting	Patting	Turning	Normal
Original (%)	100	100	93.305	100	83.575	71.849
20% of Data Loss (%)	100	100	91.246	100	83.488	70.922
33% of Data Loss (%)	100	100	90.751	100	83.074	70.282
50% of Data Loss (%)	100	100	87.740	100	81.339	68.822

4.1. Discontinuous Frames Loss Problem

Verification Results

Robustness

Feature of each motion can be learned by time series data analysis



50% data loss

Accuracy: 2.368%

4.2. Continuous Frames Loss Problem

1000 samples was randomly generated

Verification Method

N continuous frames

......

0.1	0.3	0.4	0.5	0.5	0.6	0.6	0.7
0.2	0.1	0.1	0.3	0.5	0.7	0.8	0.8
0.8	0.8	0.9	1.1	1.2	1.2	1.3	1.4
1.2	0.9	0.8	0.6	0.3	0.2	0.2	0.1
1.6	1.6	1.7	1.8	1.9	1.9	2.0	2.0
1.7	2.1	2.2	2.5	2.5	2.6	2.7	2.8

......

Randomly Selected as data lost frames

K continuous data lost frames

Randomly Selection of Lost Data

- ✓ Randomly selecting 10 continuous frames from each 100 continuous frames; → *10% data lost*
- ✓ Randomly selecting 20 continuous frames from each 100 continuous frames; → *20% data lost*
- ✓ Randomly selecting 30 continuous frames from each 100 continuous frames; → *30% data lost*
- ✓ Randomly selecting 40 continuous frames from each 100 continuous frames; → *40% data lost*
- ✓ Randomly selecting 50 continuous frames from each 100 continuous frames. → *50% data lost*

4.2. Continuous Frames Loss Problem

Verification Results

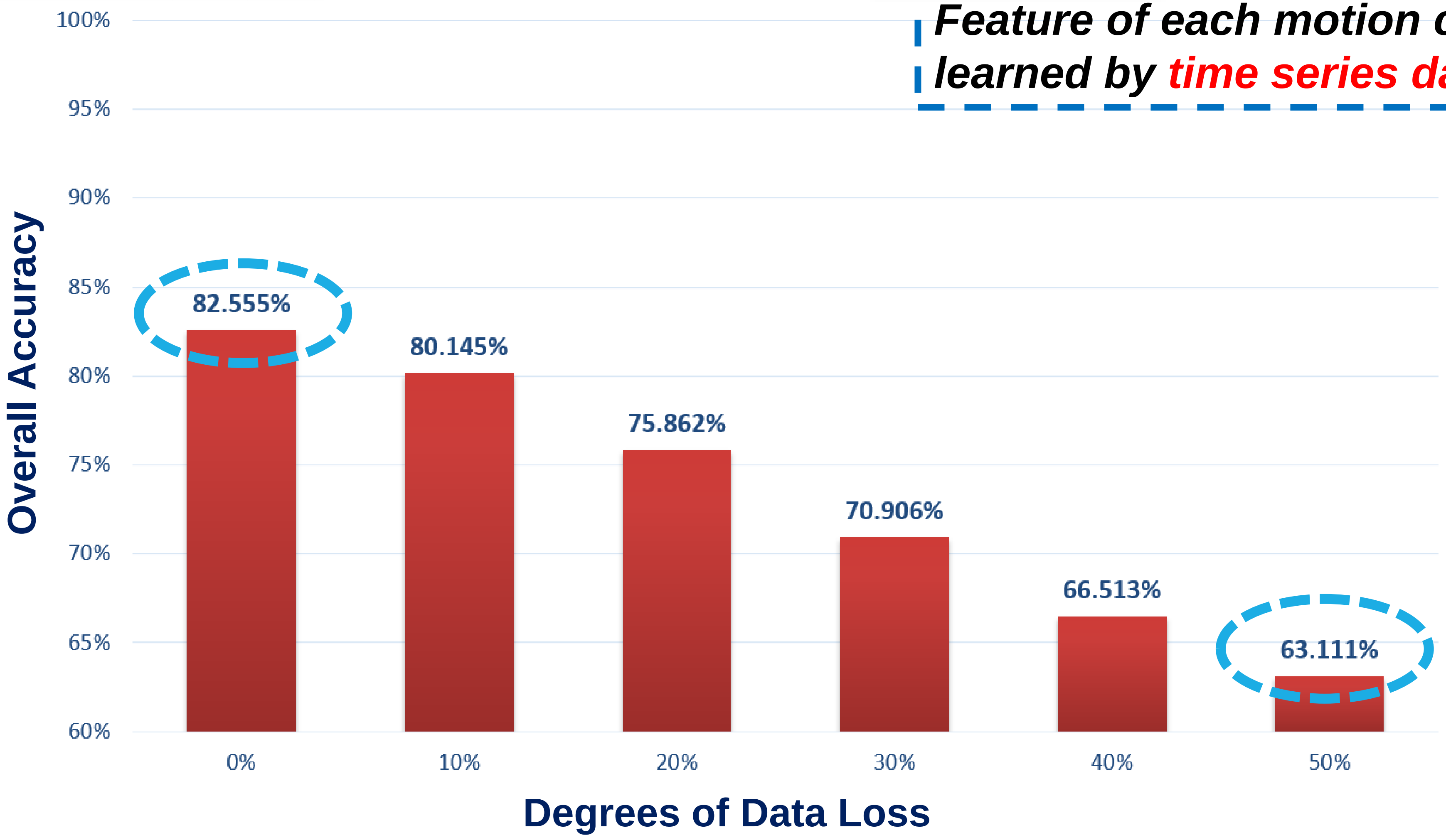
	Pushing	Grasping	Cutting	Patting	Turning	Normal
Original (%)	100	100	93.305	100	83.575	71.849
10% of Data Loss (%)	99.998	99.480	89.302	99.998	80.887	68.624
20% of Data Loss (%)	99.600	93.655	89.302	96.709	74.754	68.624
30% of Data Loss (%)	95.663	84.075	81.857	89.312	68.257	60.027
40% of Data Loss (%)	87.432	76.408	78.909	82.773	63.479	57.001
50% of Data Loss (%)	82.226	72.223	76.368	79.088	59.977	53.949

4.2. Continuous Frames Loss Problem

Verification Results

Robustness

Feature of each motion can be learned by time series data analysis



50% data loss

Accuracy: 19.444%

Contents

1. Introduction

2. Hand Motion Capture

3. Behavior Recognition

4. Robustness Verification

5. Conclusion & Future Work

5.1 Conclusion

Objective1:

Hand Motion Capture

- For detection of insiders' sabotage behaviors, a new hand motion detection algorithm was proposed;
- Real-time hand motion capturing system was developed and time series data of each fingertip was successfully obtained with 29.8fps;

Objective2:

Behavior Recognition

- Behavior recognition method was developed by using Time-Series Data Analysis.
- Assumed malicious motions can be classified into different patterns and detected with high accuracy in short time, thus real-time detection is possible.

5.1 Conclusion

Objective3:

Robustness Verification

- Even though dealing with 50% data loss, the accuracy decreases only 2.368% and 19.444% for discontinuous and continuous frames loss problem. Thus, our behavior recognition method can be considered as a robust method.

5.2 Future Work

- The hand motion detection algorithm will be improved to achieve practicality (e.g. recognition finger motion when capturing tool).
- Detailed motion classification and a malicious motion database will be generated;
- Prediction of malicious motions for earlier response.

Thank you for your kind attention!

Q & A Pages

Accurate Tracking of Finger Tip Position

- Marker tracking for raw DB
- Model based feature extraction
- Model based estimation
- Correlation among time-series of each finger tip

Probabilistic Estimation of hidden fingers

- Some finger may not visible
- Need to estimate true position of finger tip
- Probabilistic estimation due to history, restriction, experience

Identification of Finger-Hand-Arm behavior

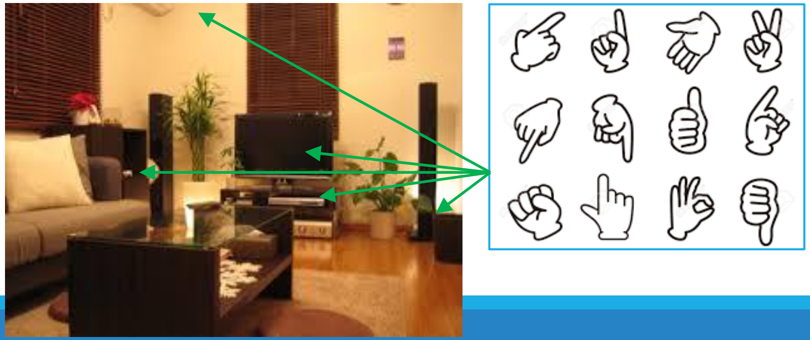
- Database of fingertip time series data in multi-variable space
- Clustering and classification of motion
- Construction of Database
- Identification by this DB


Virtual display

Estimation of hidden fingers

Detection of suspicious behavior

Information Entropy ?

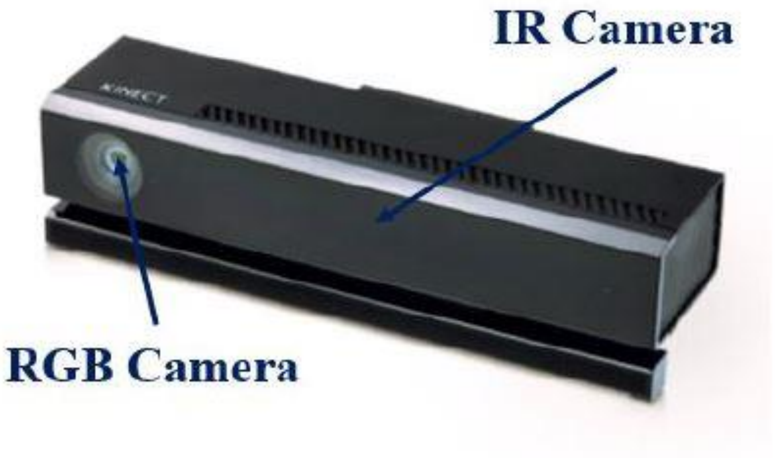


IoT by hand sign (not by voice)


Sign language translation

Kinect v2

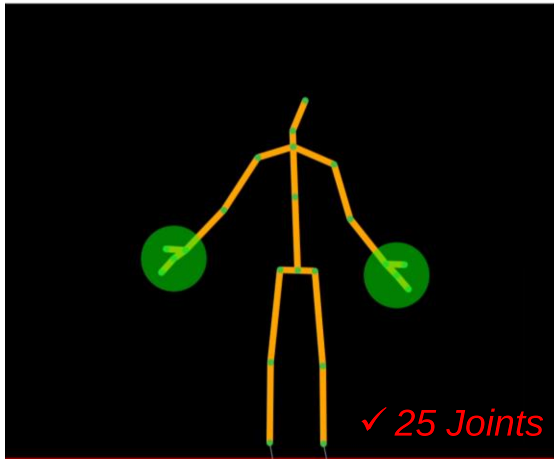
Kinect v2 is the new version of game controller technology introduced by Microsoft.



Key Features

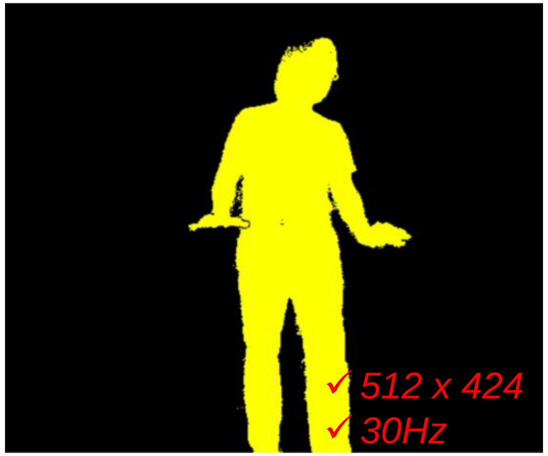
Improved Body Tracking	Tracks as many as **6 complete skeletons** and **25 joints** per person
Depth Sensing	**512 x 424** **30 Hz** FOV: 70 x 60 One mode: 0.5–4.5 meters
1080p Color Camera	**30 Hz** (15 Hz in low light)
New Active Infrared (IR) Capabilities	**512 x 424** **30 Hz**

Kinect v2

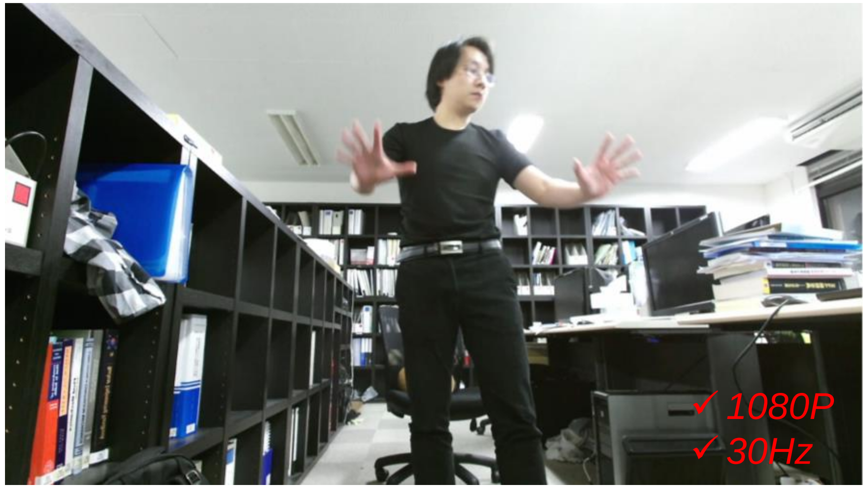


Skeleton Frame

Depth Frame



Body Index Frame



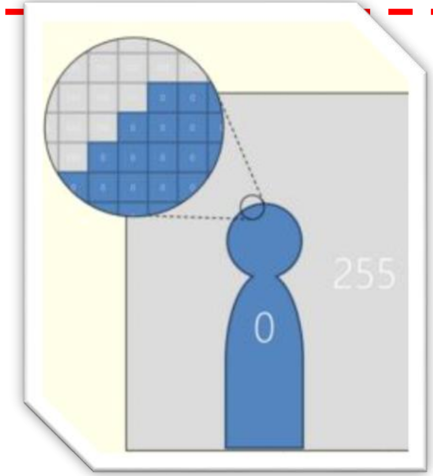
Color Frame

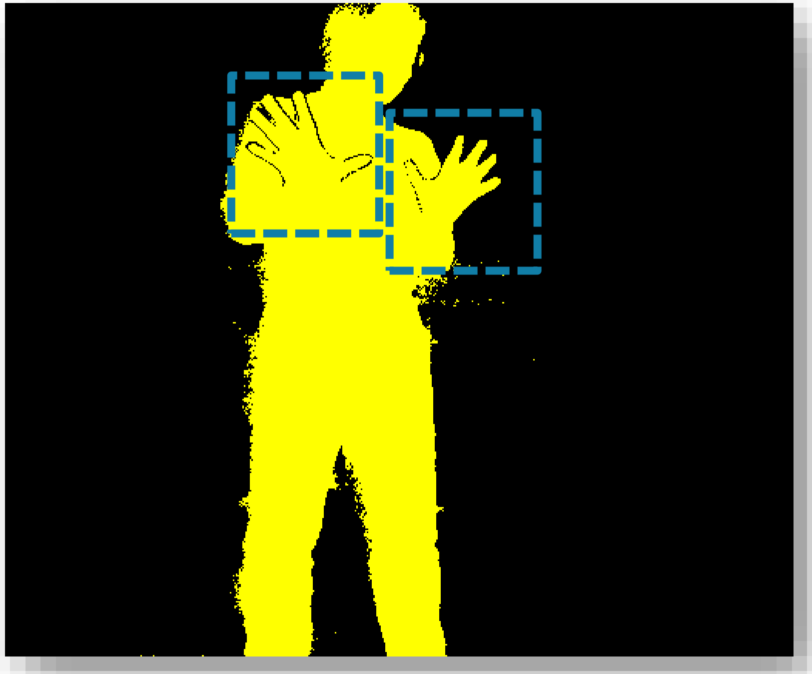
Infrared Frame

New Development for Hand Region Classification

Body Segmentation

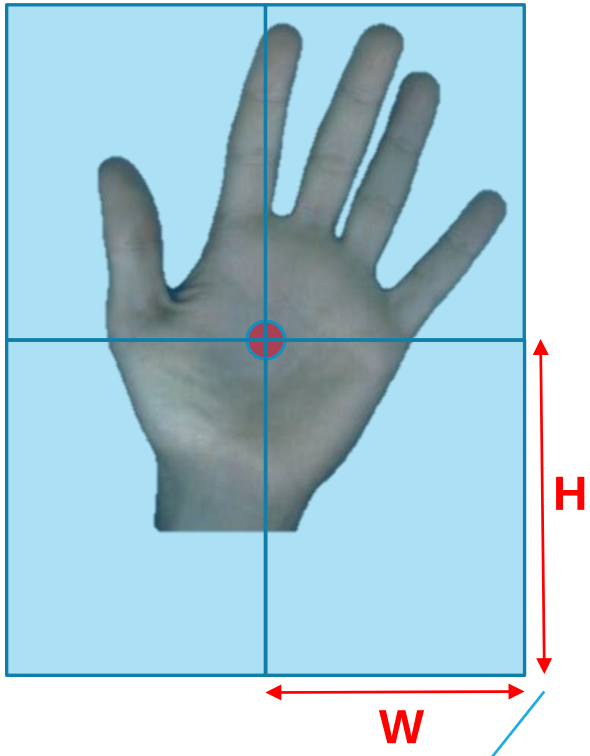
Body Index Frame
- ✓ 512 x 424;
- ✓ 30FPS.





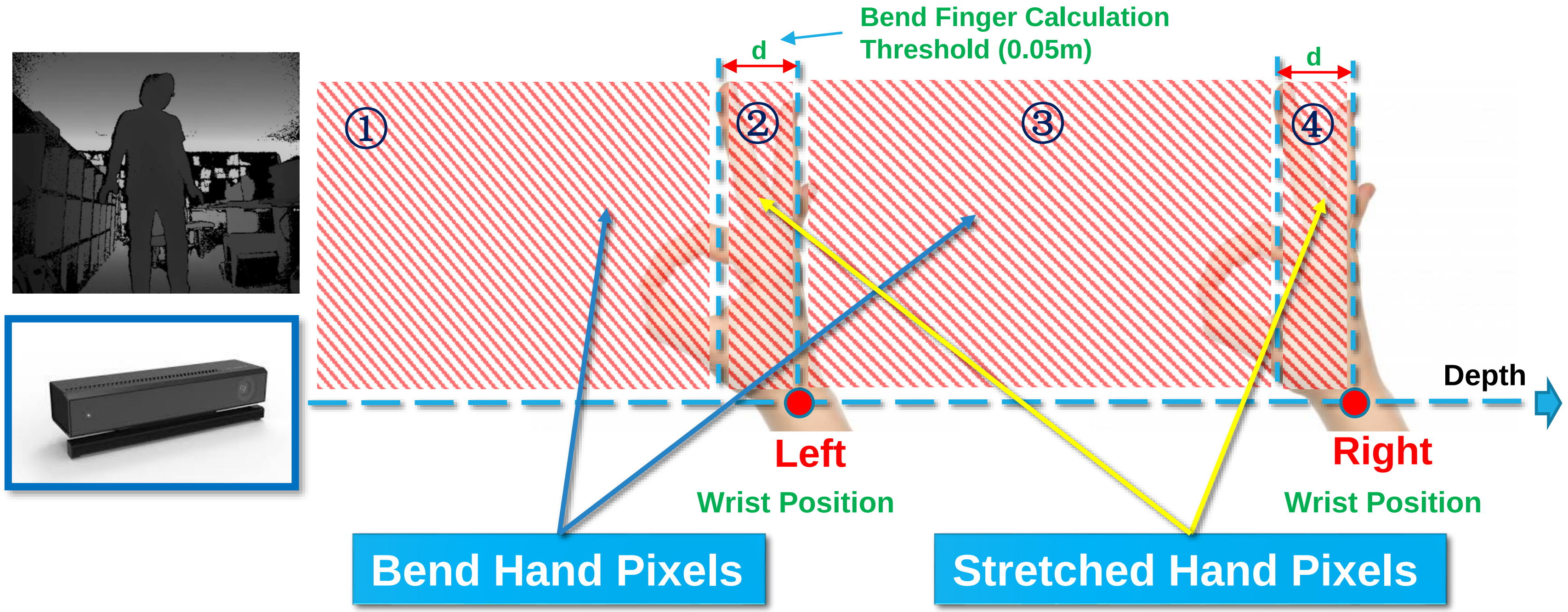
Hand Segmentation

Skeleton Frame
Joint Point



The values of W & H depends on the distance from human body to camera.

New Development for Hand Region Classification



I. Get left & right hand wrist position (using Kinect Skeleton Frame);
II. Distinguish left & right hand;
III. Get each bend fingers part of left & right hands by a threshold.

① Left bend part
② Left stretched part
③ Right bend part
④ Right stretched part

New Development for Hand Region Classification

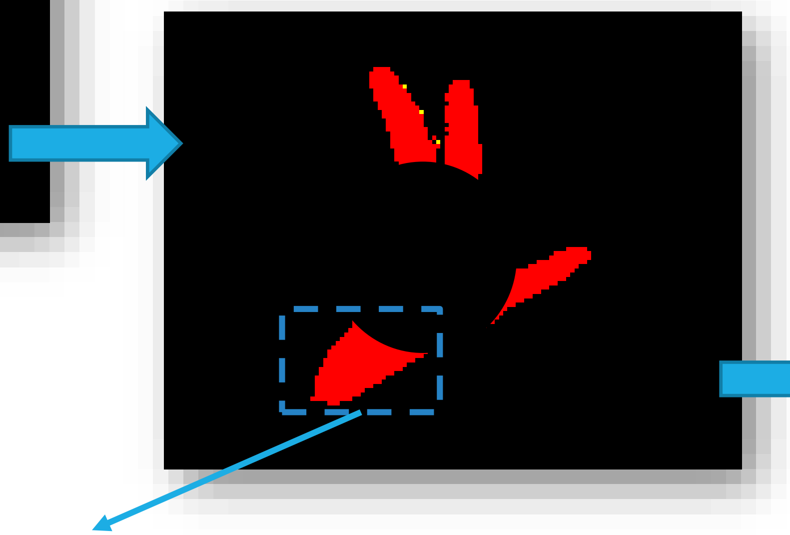
Result of Hand Region Classification

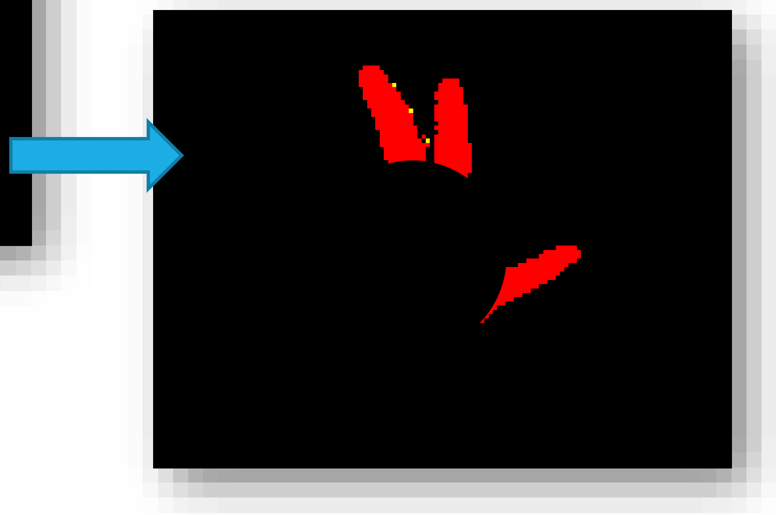
Pixels in hand region was successfully classified into different parts:

- *Red pixels: stretched hand region*
- *Yellow pixels: bend hand region*
- *Black pixels: background*

New Development for Fingers Segmentation

Segmenting stretched fingers from palm.



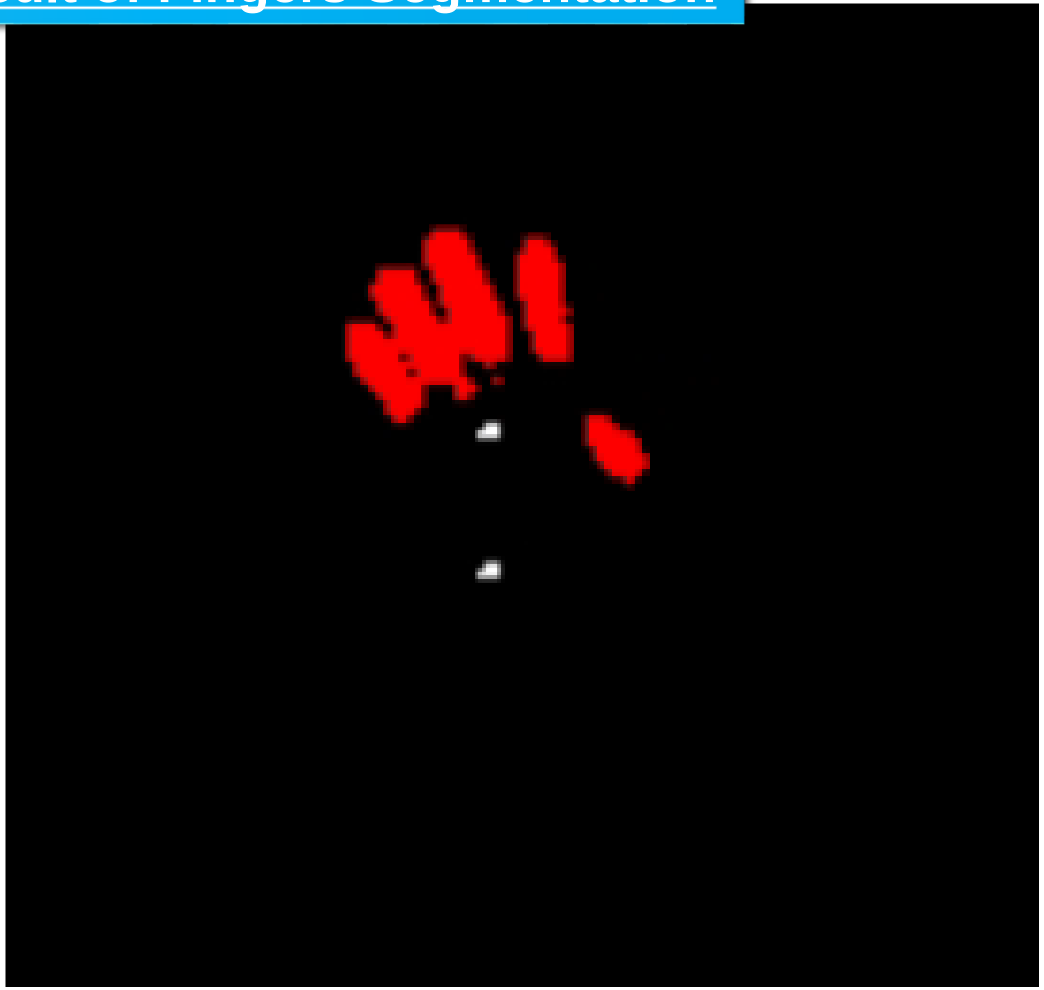


Noises

Can be Filtered by analyzing body index data and depth data of Kinect.

New Development for Fingers Segmentation

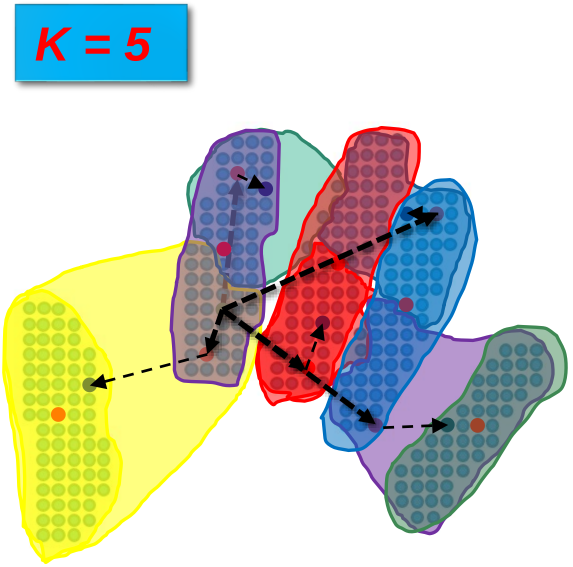
Result of Fingers Segmentation



Fingers was successfully segmented from palm.

K-means Clustering Algorithm

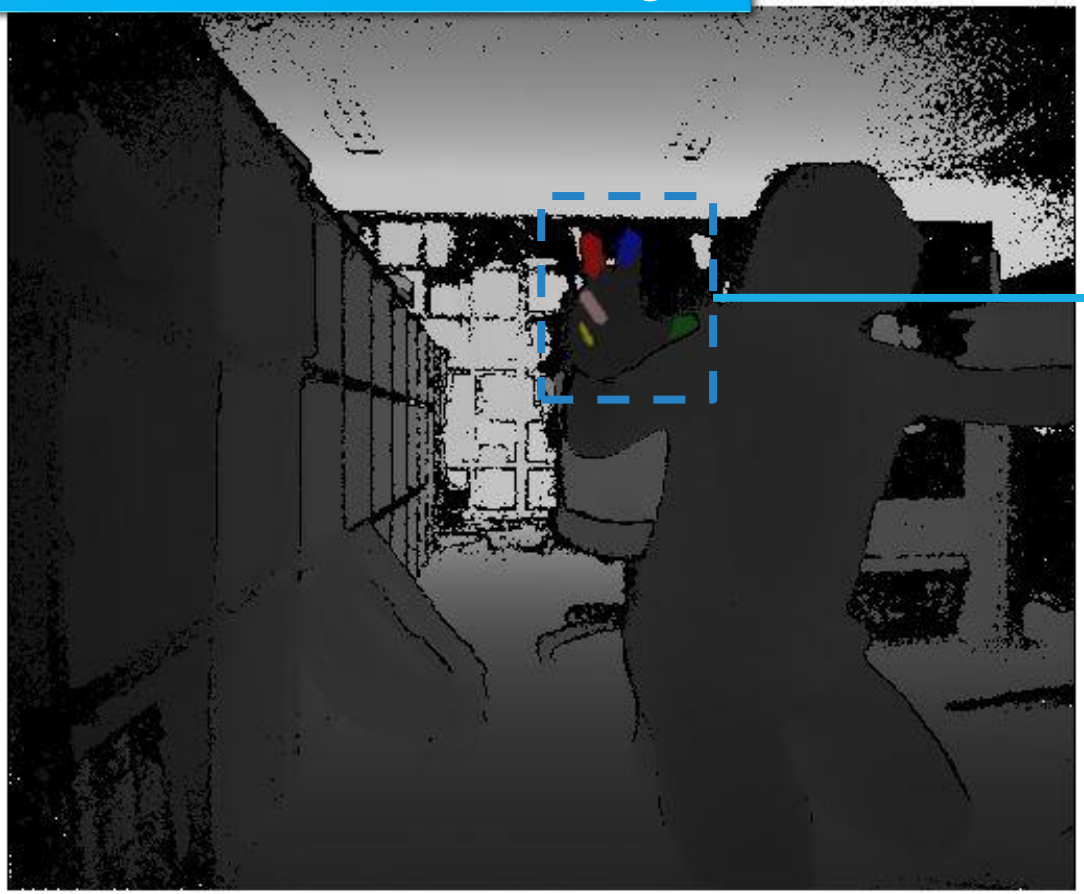
1. K initial "means" (in this case K=5) are randomly generated within the data domain;
2. K clusters are created by associating every observation with the nearest mean;
3. The centroid of each of the K clusters becomes the new mean;
4. Steps 2 and 3 are repeated until convergence has been reached.

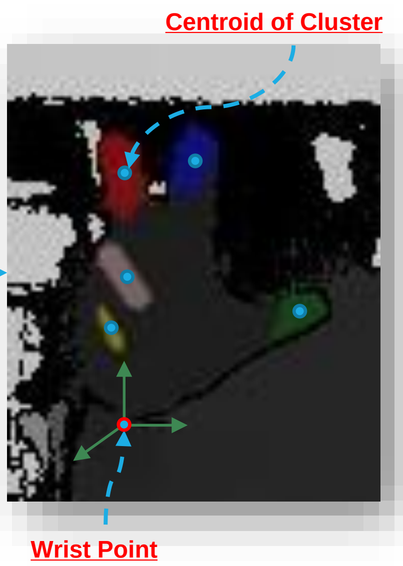


(Ray S, Turi RH, "Determination of number of clusters in K-means clustering and application in colour image segmentation", Proceedings of the 4th international conference on advances in pattern recognition and digital techniques (ICAPRDT'99), Calcutta, India, pp 137–143.)

Fingers Identification

Result of K-means Clustering





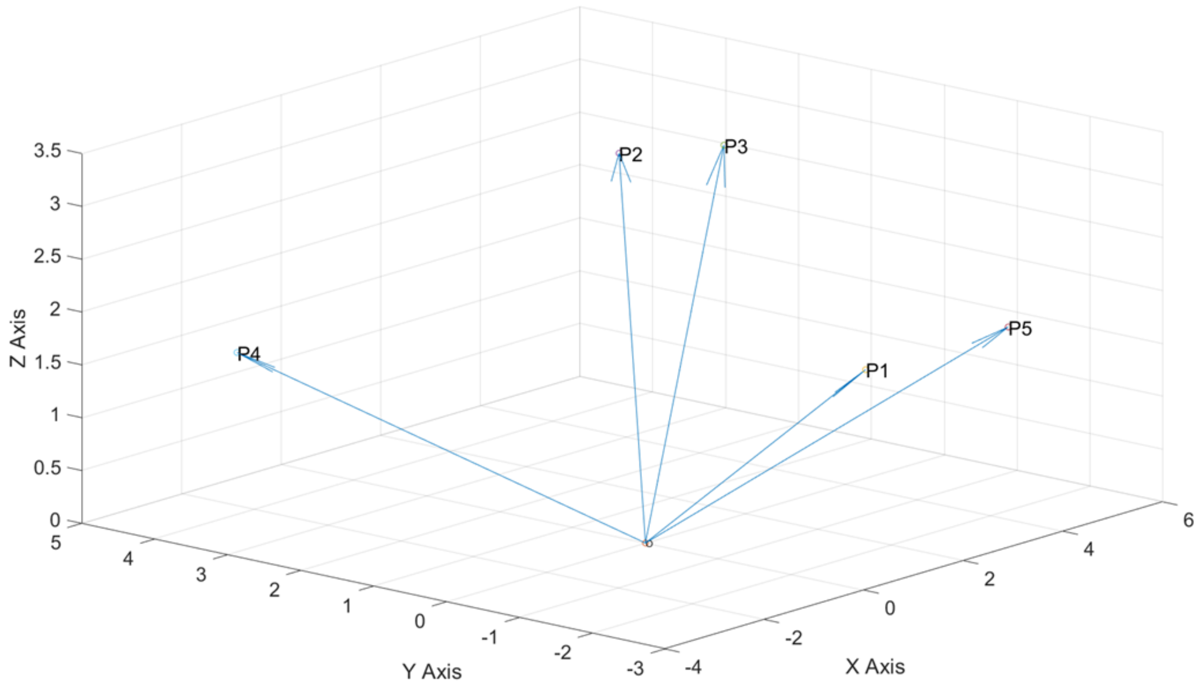
Different initial means can result in different final clusters.

Disadvantage

Fingertip Calculation

For each cluster of finger pixels, the fingertip is the pixel which has closest distance to camera.

Fingers Identification



Bubble Sort Algorithm

To get relative positions all five vectors of finger.

① (P1, P2, P3, P4, P5) → (P2, P1, P3, P4, P5)
(P2, P1, P3, P4, P5) → (P2, P3, P1, P4, P5)
(P2, P3, P1, P4, P5) → (P2, P3, P4, P1, P5)
(P2, P3, P4, P1, P5) → (P2, P3, P4, P1, P5)

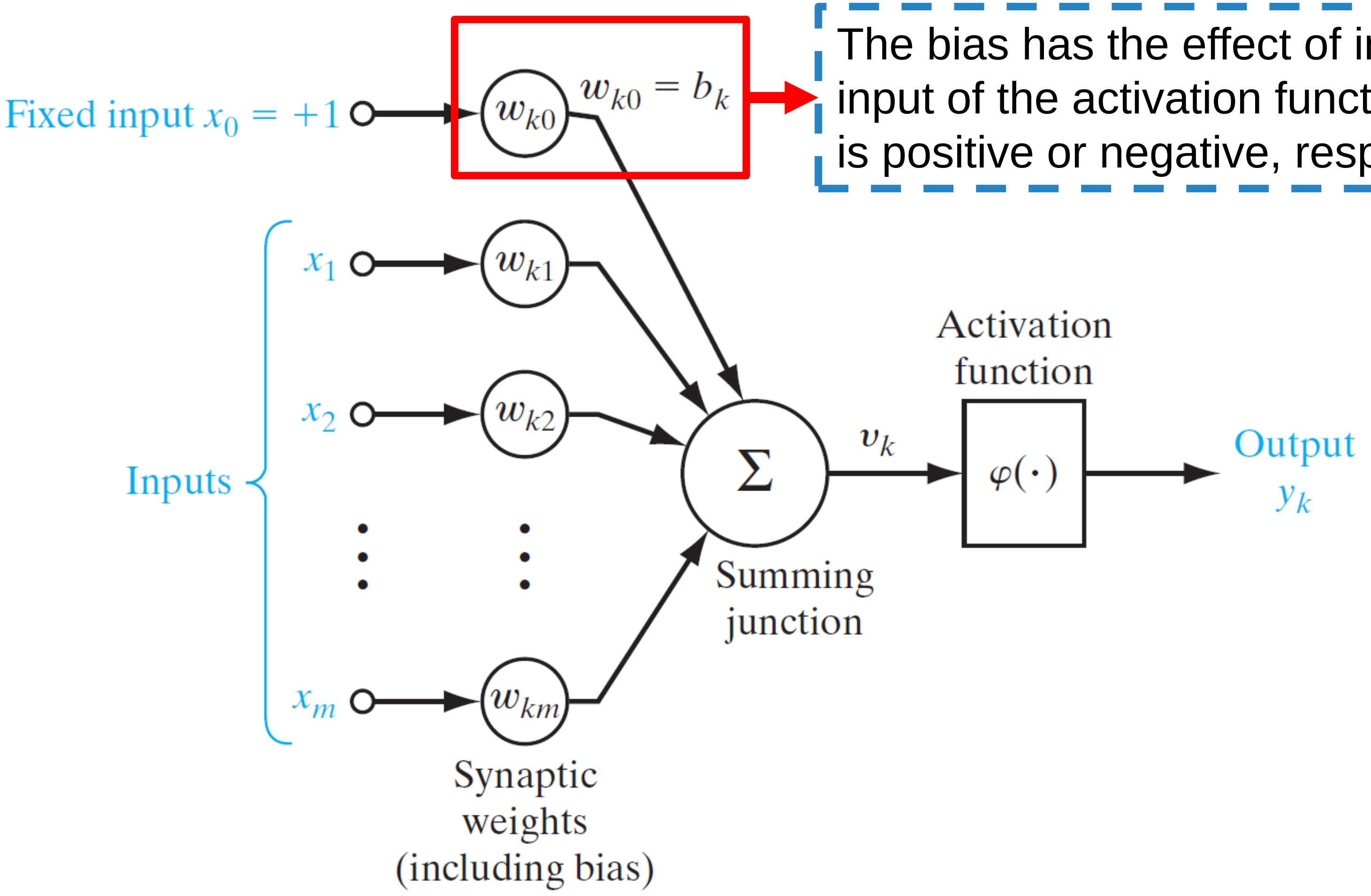
② (P2, P3, P4, P1, P5) → (P2, P3, P4, P1, P5)
(P2, P3, P4, P1, P5) → (P2, P4, P3, P1, P5)
(P2, P4, P3, P1, P5) → (P2, P4, P3, P1, P5)
(P2, P4, P3, P1, P5) → (P2, P4, P3, P1, P5)

③ (P2, P4, P3, P1, P5) → (P4, P2, P3, P1, P5)
(P4, P2, P3, P1, P5) → (P4, P2, P3, P1, P5)
(P4, P2, P3, P1, P5) → (P4, P2, P3, P1, P5)
(P4, P2, P3, P1, P5) → $\boxed{\text{(P4, P2, P3, P1, P5)}}$

Cross Product Calculation

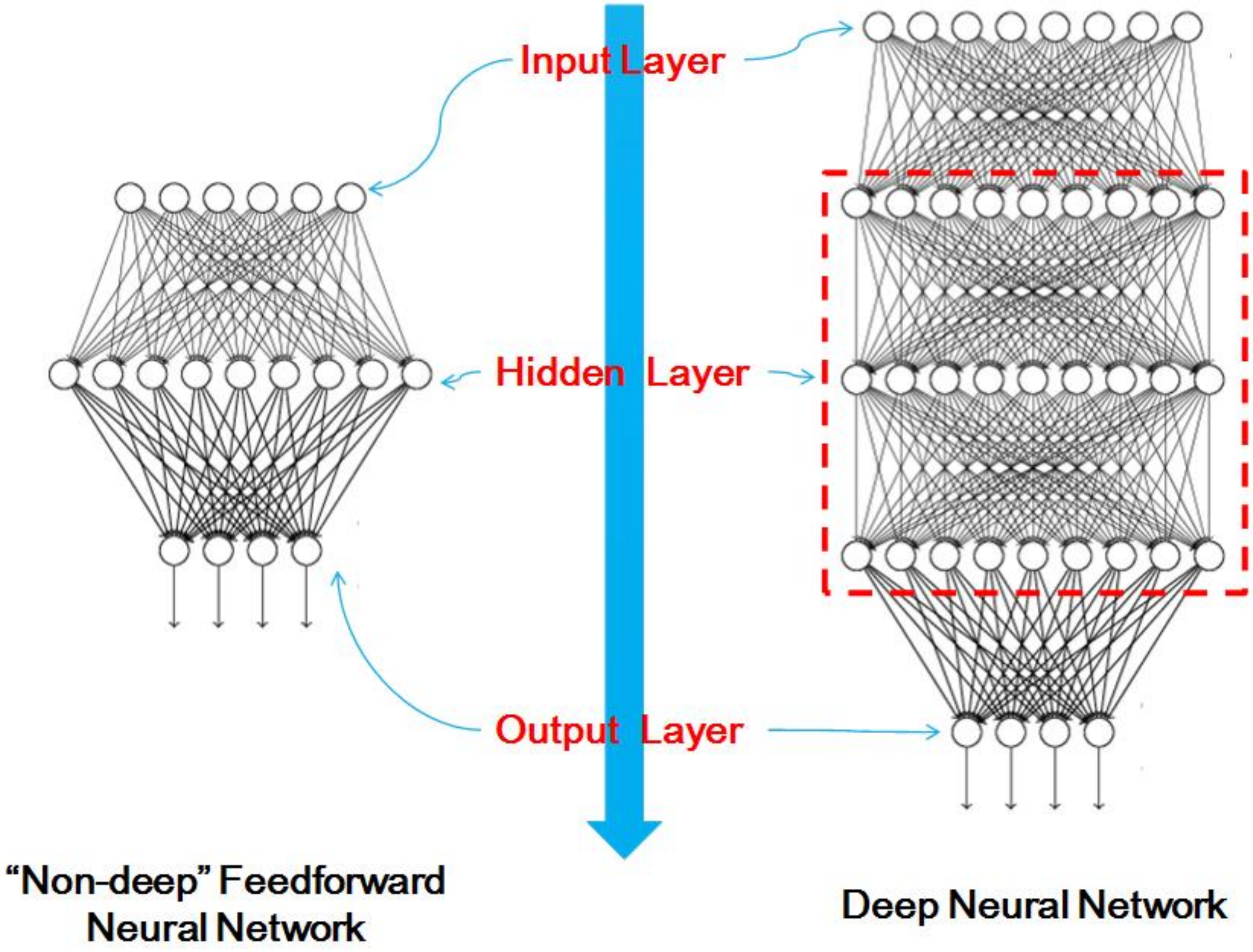
If $P_1 \times P_2 > 0$, then P_1 is in the clockwise direction of P_2, otherwise P_1 is in the counter clockwise direction of P_2.

Neural Networks

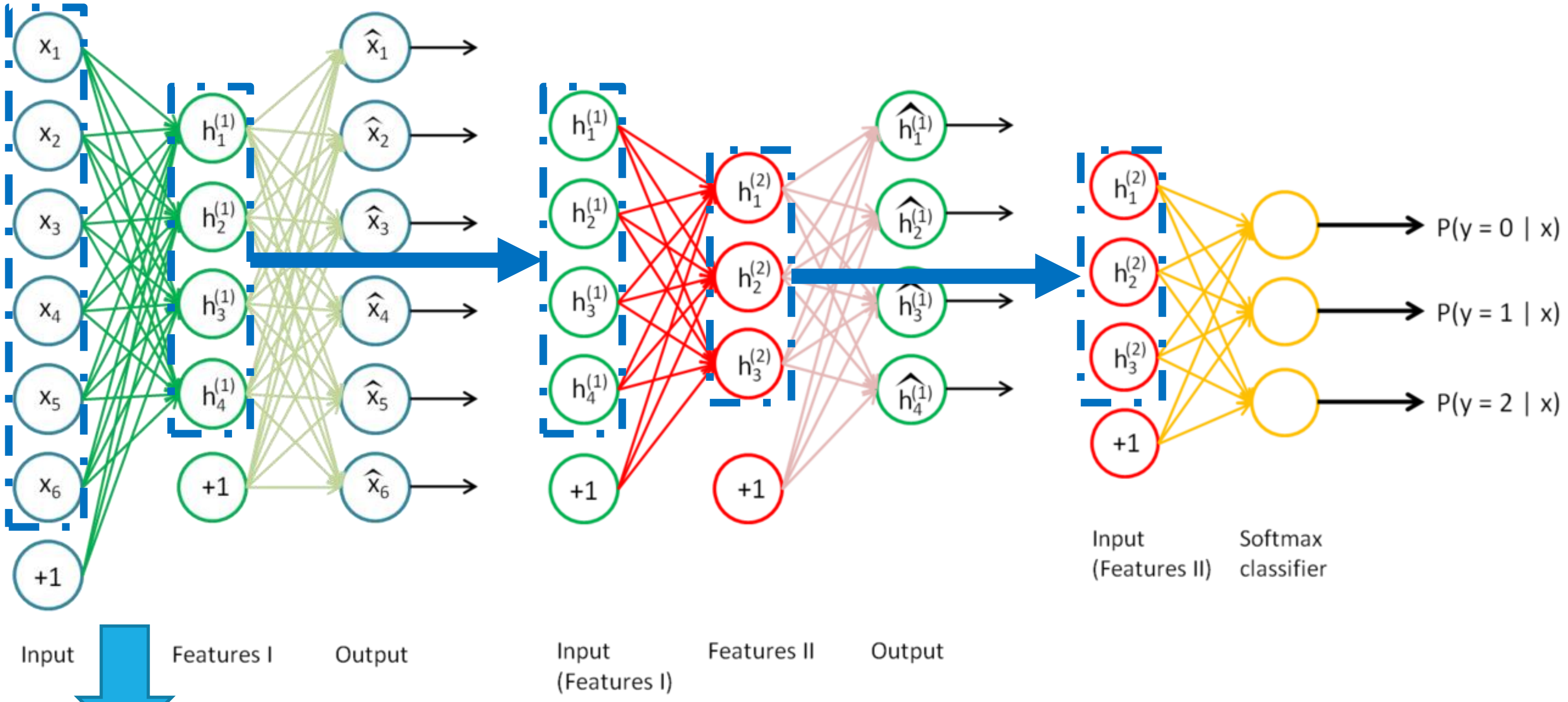


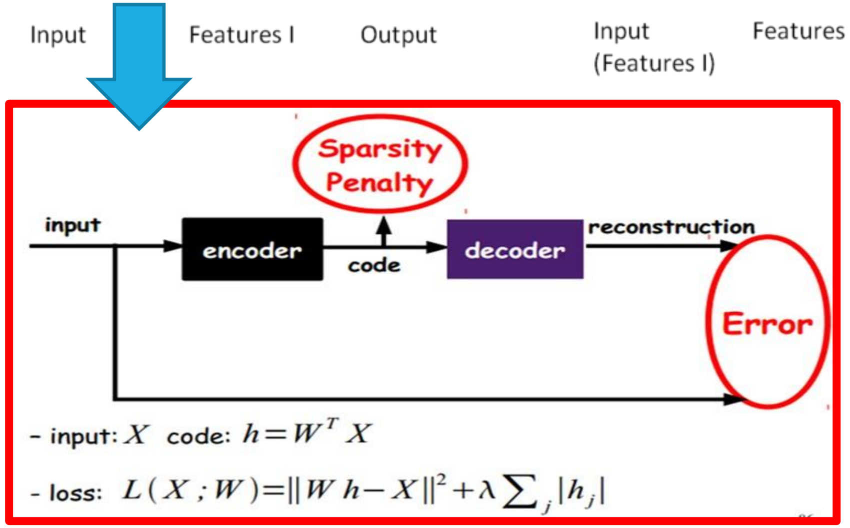
The bias has the effect of increasing or lowering the net input of the activation function, depending on whether it is positive or negative, respectively.

Deep Neural Networks



Stacked Auto-Encoder

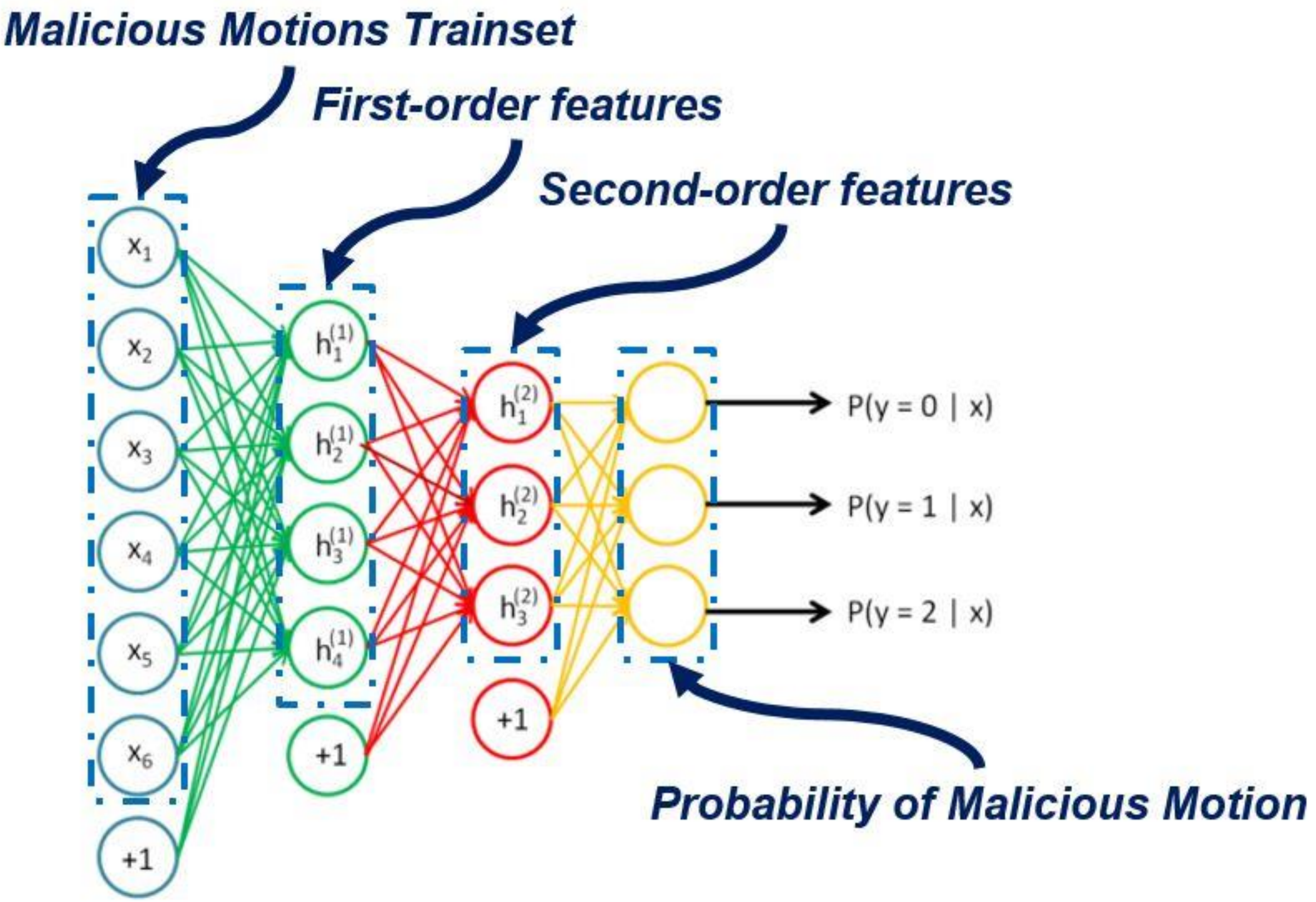




A stacked autoencoder enjoys all the benefits of any deep network of greater expressive power. Further, it often captures a useful "hierarchical grouping" or "part-whole decomposition" of the input.

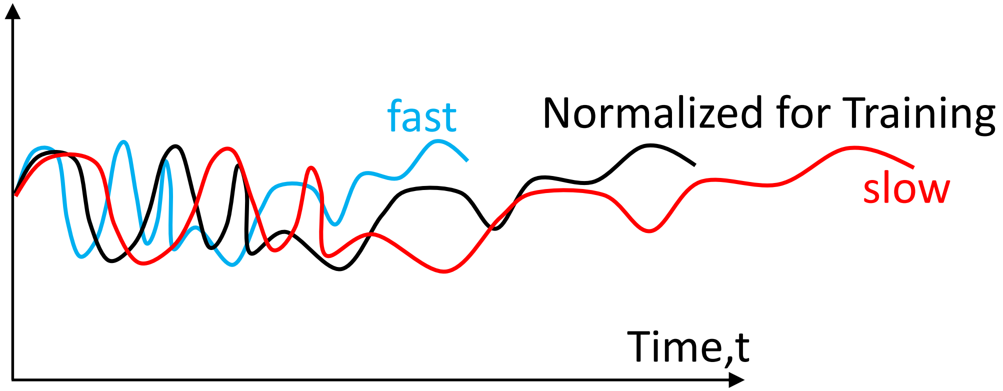
(Stanford University UFLDL Tutorial: http://ufldl.stanford.edu/wiki/index.php/Stacked_Autoencoders)

Stacked Auto-Encoder

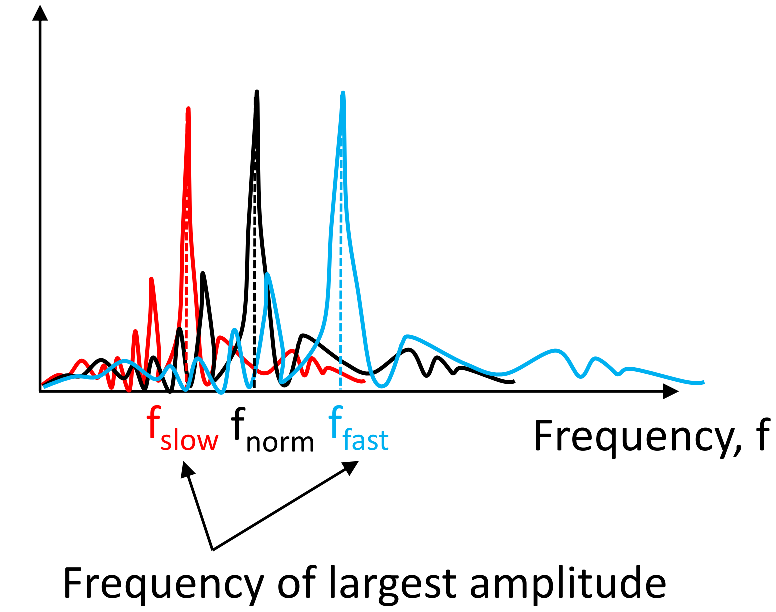


How to identify a same hand behavior with different speed

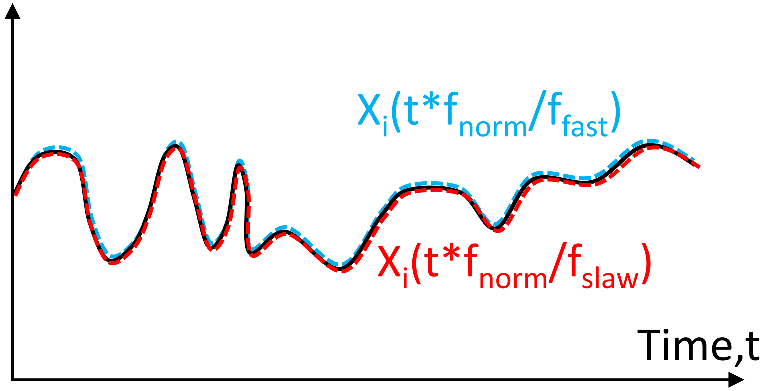
Time Series Data of Finger Position $X_i(t)$



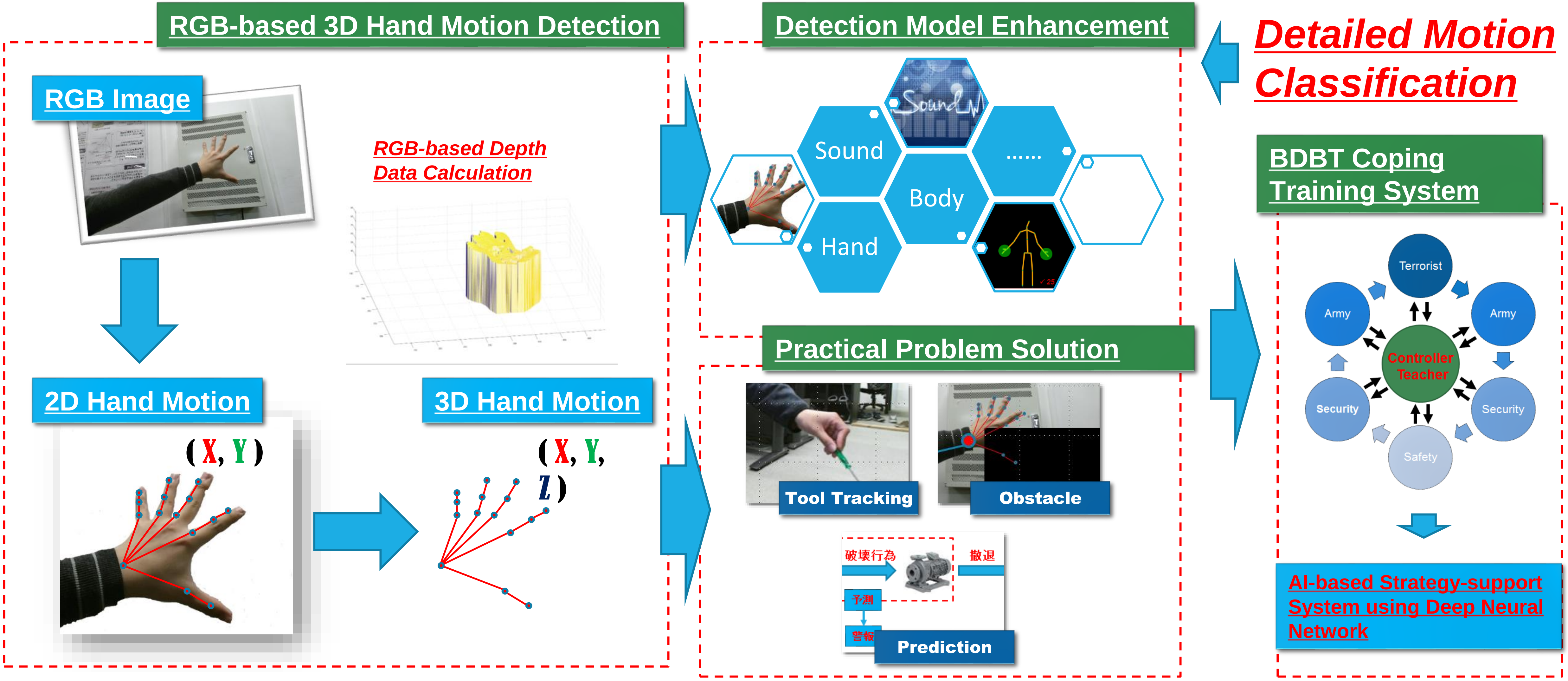
FFT of $X_i(t)$



Normalized Time Series Data of Finger Position $X_i(t)$



Future Work



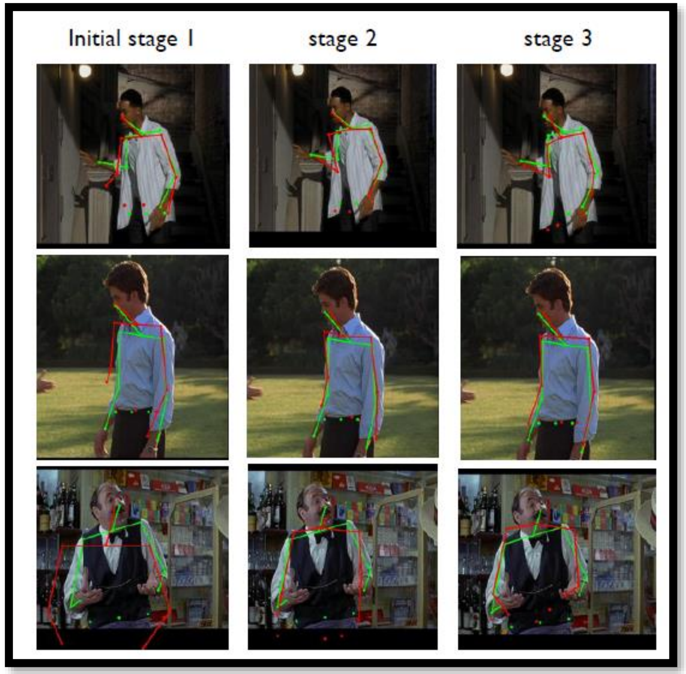
RGB-based 3D Hand Motion Detection
RGB Image
RGB-based Depth
Data Calculation
2D Hand Motion
(X, Y)
3D Hand Motion
(X, Y, Z)
Detection Model Enhancement
Sound
Body
Hand
......
Practical Problem Solution
Tool Tracking
Obstacle
破壞行為
撤退
予測
警報
Prediction
Detailed Motion Classification
BDBT Coping Training System
Terrorist
Army
Army
Controller
Teacher
Security
Security
Safety
AI-based Strategy-support System using Deep Neural Network

2D Hand Motion Detection

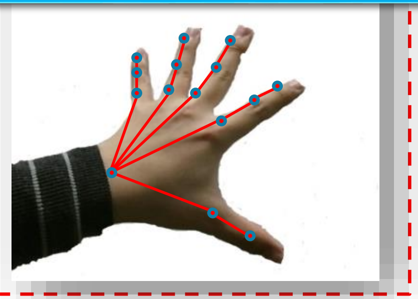
Human Pose Estimation via Deep Neural Networks

- Using Convolutional Neural Network (CNN) to learn features of images and estimate position of each body joint;
- Cascade of Pose Regressors.

Result (2D positions)



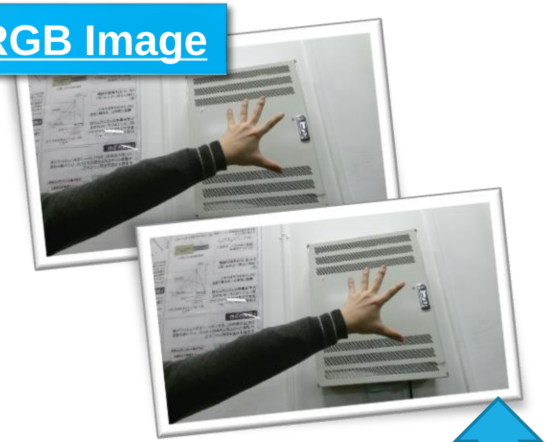
2D Hand Motion Estimation



(Toshev, Alexander, and Christian Szegedy. "Deeppose: Human pose estimation via deep neural networks." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2014.)

3D Hand Motion Detection

RGB Image



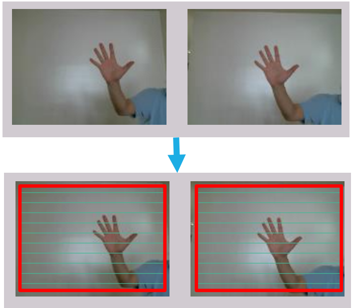
Stereo Camera

①

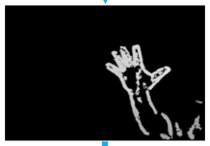
Capturing RGB images using stereo camera.

②

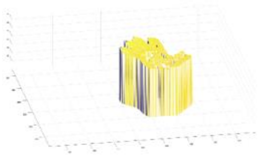
Creating depth image from RGB image using stereo vision algorithm.



Rectified stereo pair

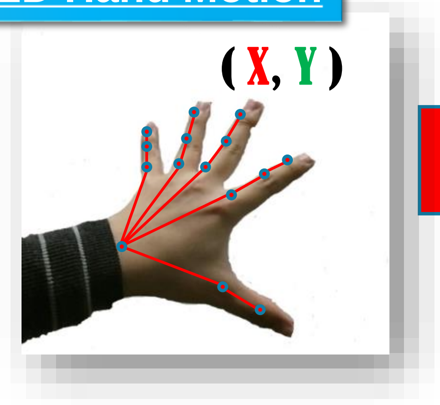


Disparity map

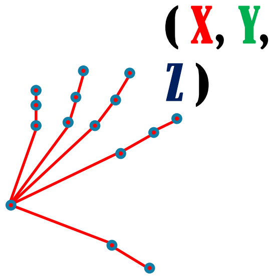


Depth map

2D Hand Motion



3D Hand Motion



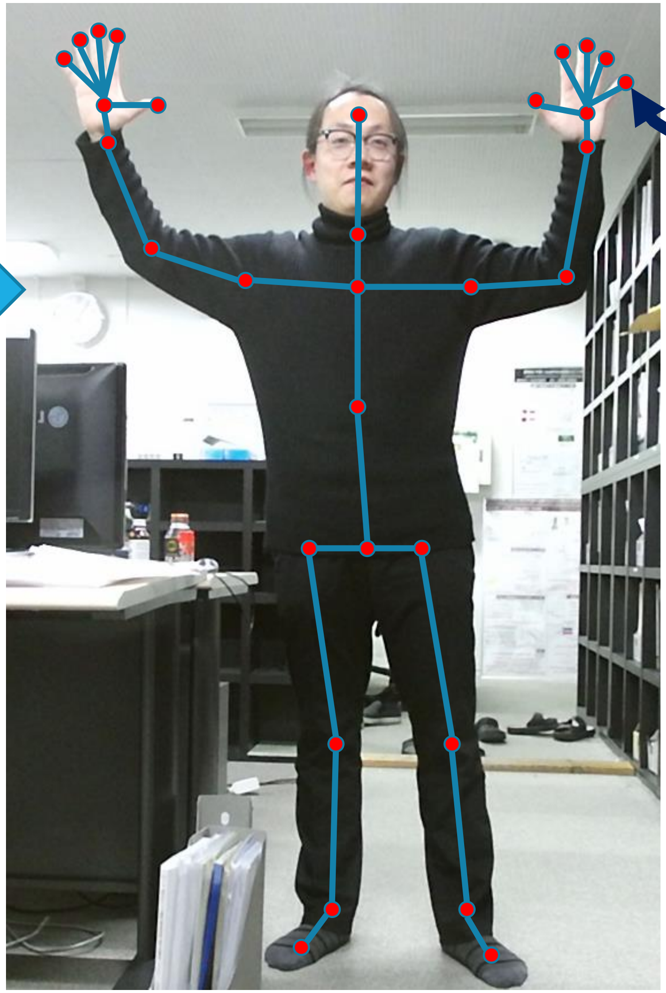
③

Implementing depth image for fingertip detection based on current algorithm and physical model of hand.

Detection Model Enhancement

Physical Model

- ① **Hand**
- ② **Body**



3D Position
(X, Y, Z)

By enhanced the detection physical model with body, more complex malicious motion can be detected.

Detection Model Enhancement

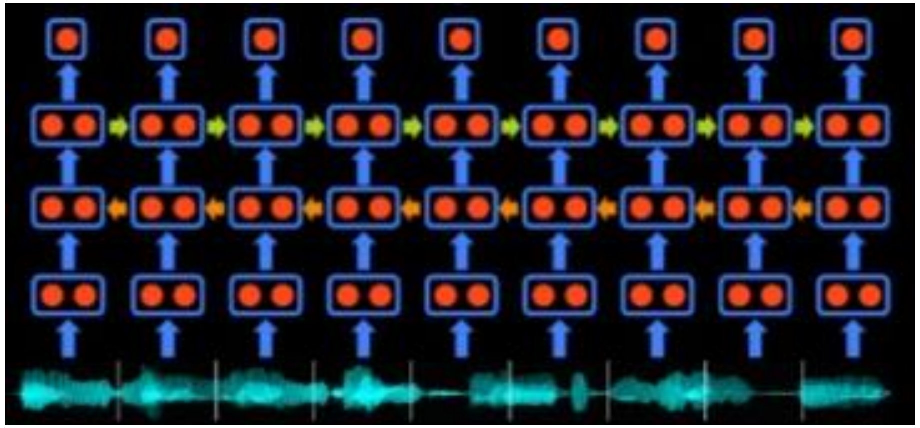
Physical Model

① **Hand**

② **Body**

Sound Information

Sound Recognition using Recurrent Neural Network (RNN)

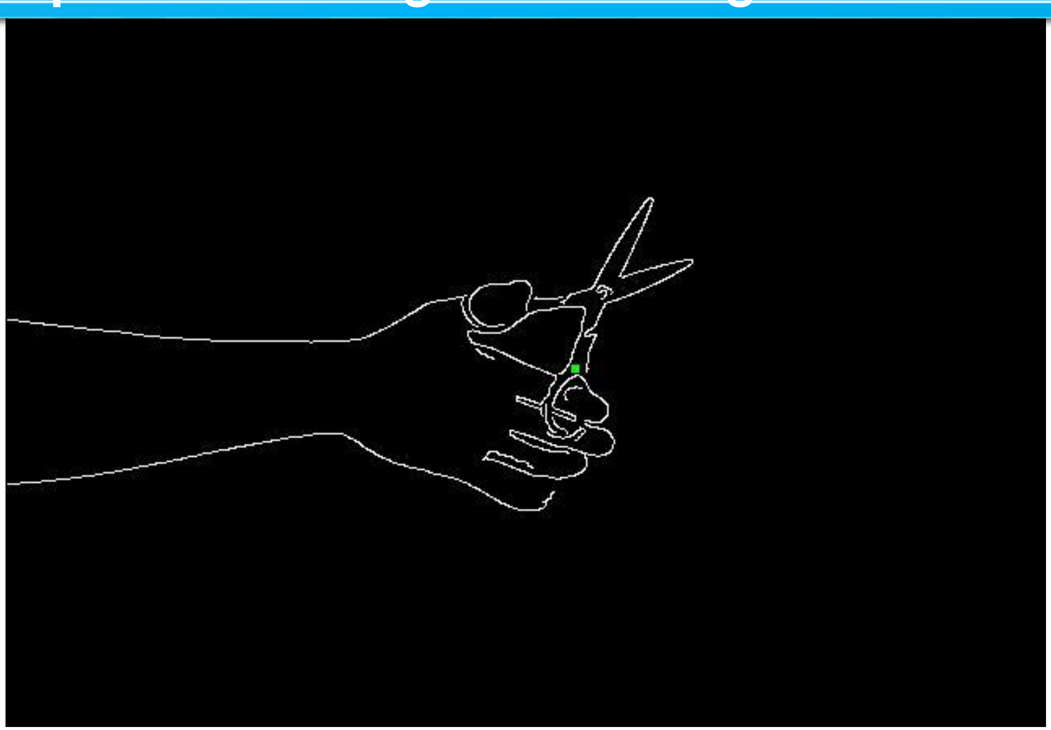


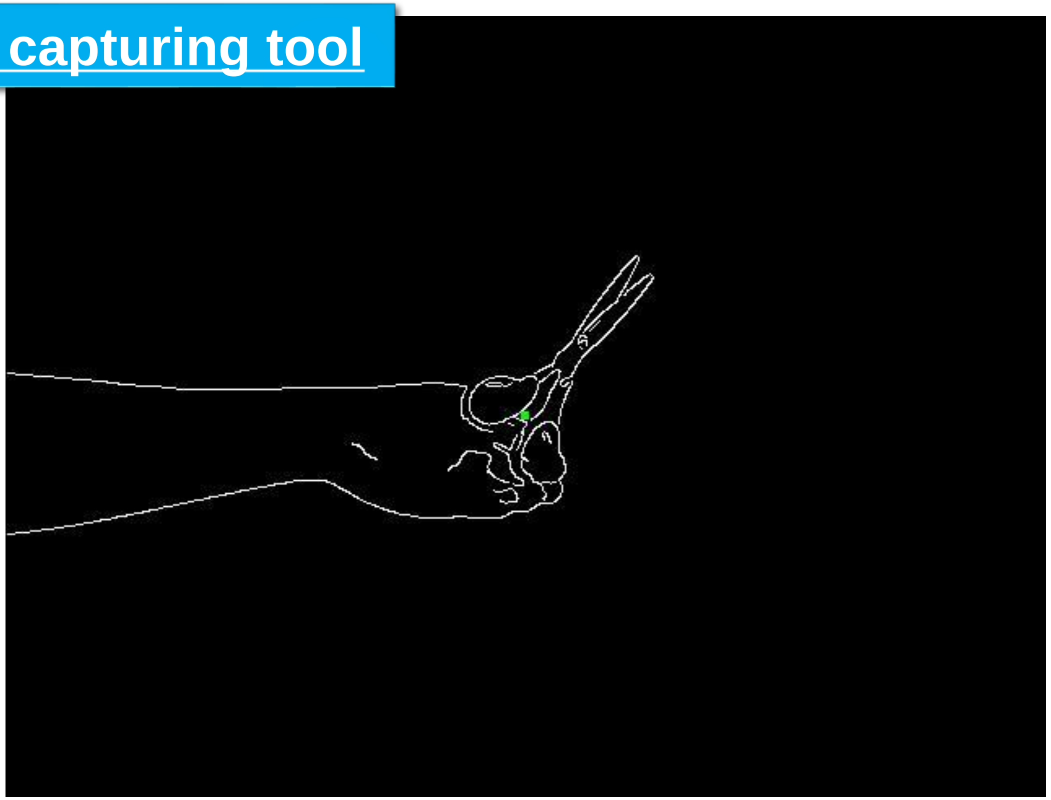
Malicious?

Ordinary?

Practical Problem Solution

Proposal for recognition of finger motion when capturing tool

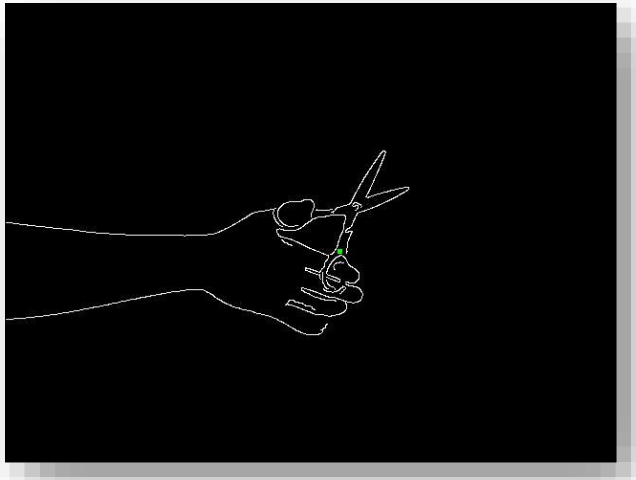




The edge of fingers and tool can be detected by using Canny edge detector

Practical Problem Solution

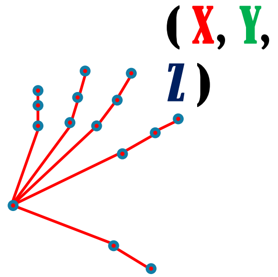
Proposal for recognition of finger motion when capturing tool



The edge of fingers and tool can be detected by using Canny edge detector

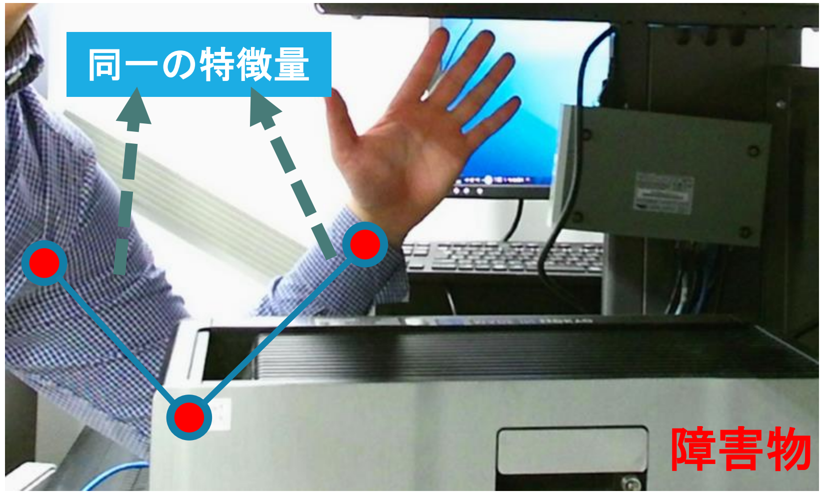
Tool detection and tracking (boundary of tool can be detected)

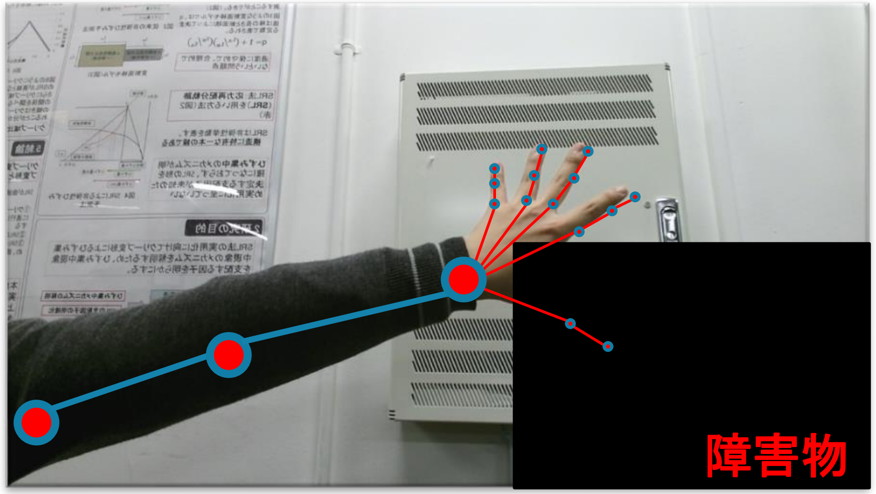
3D Hand Motion



Practical Problem Solution

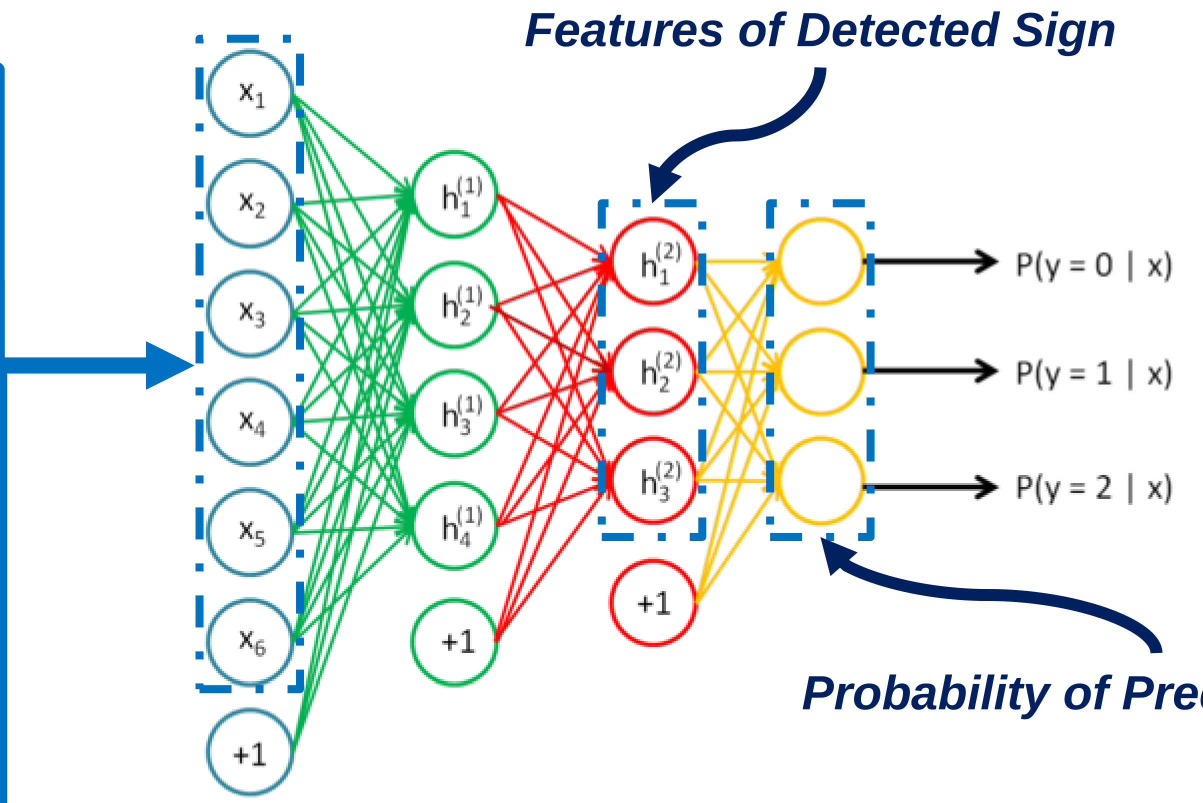
Parts of hand or body hidden in obstacle





Practical Problem Solution

Prediction for Earlier Response

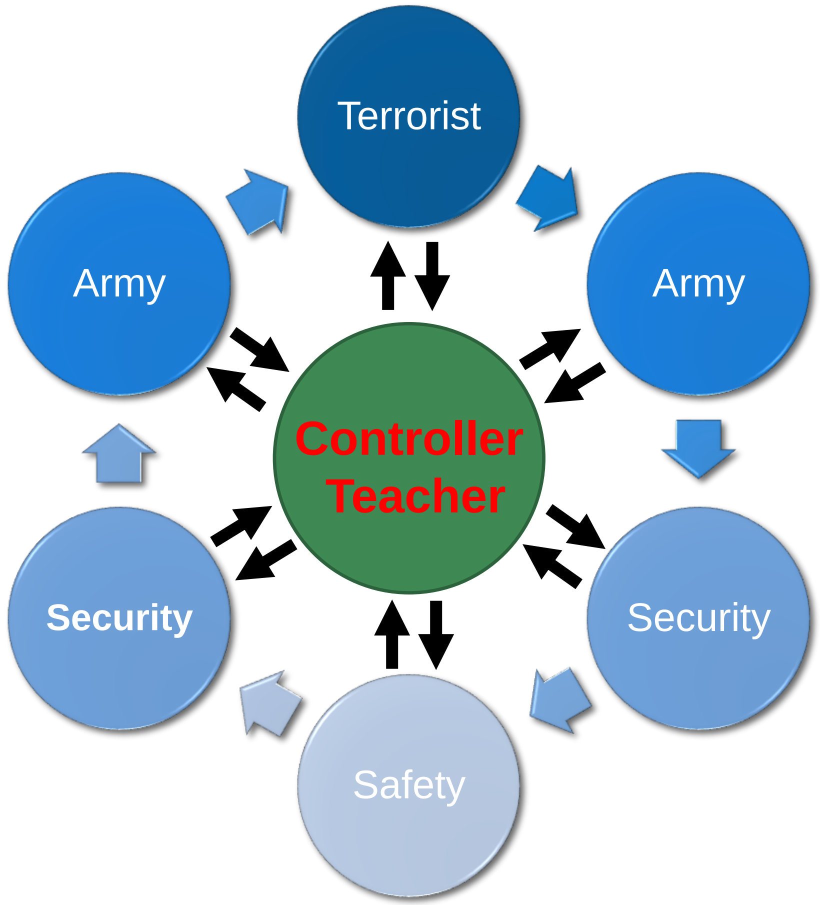


By using deep neural network, features of detected sign can be learned and future malicious motion can be predicted.

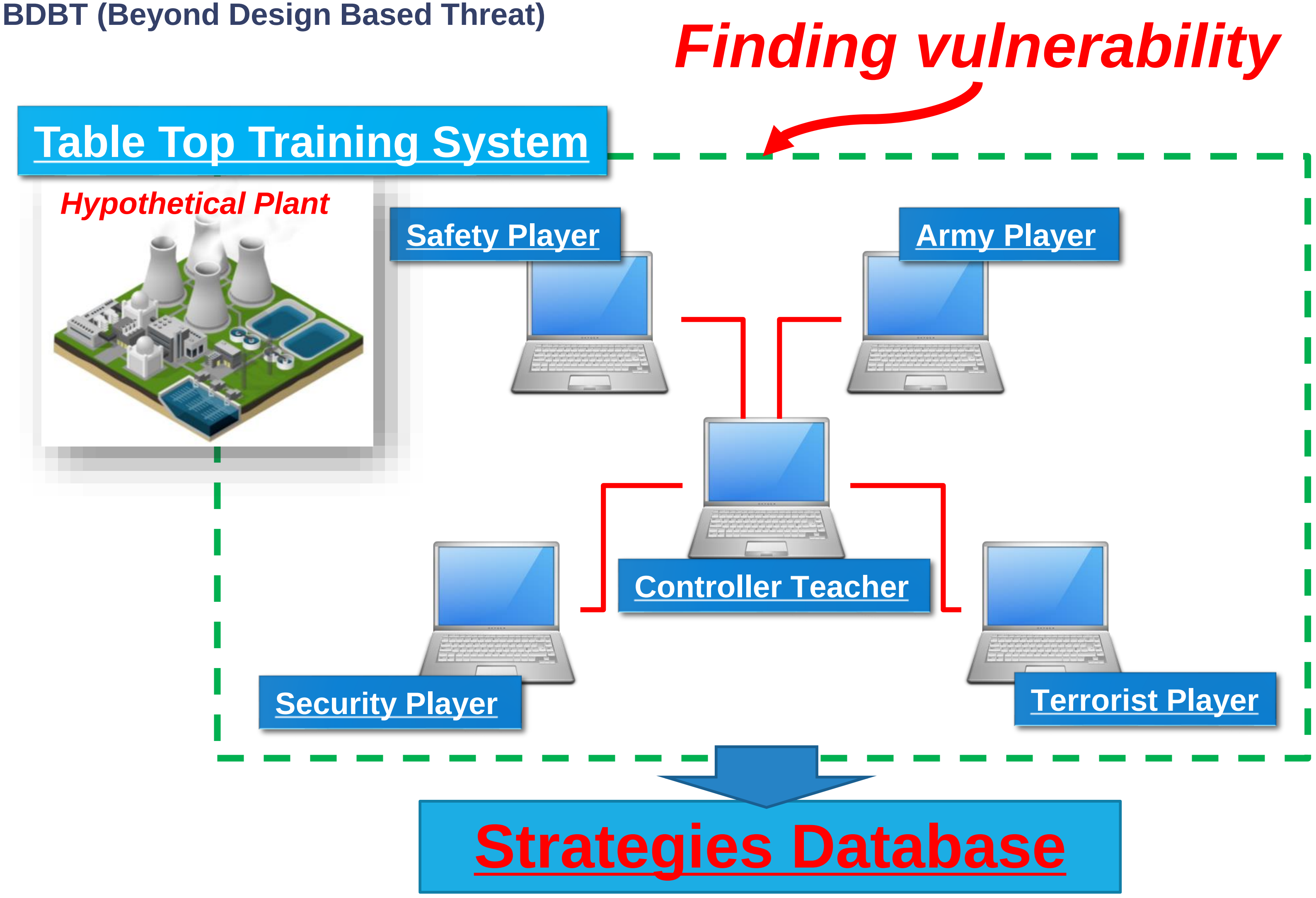
Malicious Motion Classification Database

- *Hand motion captured from 5 person;*
- *Different relative distance and angle to camera.*

Development of the BDBT Coping Training System



BDBT (Beyond Design Based Threat)

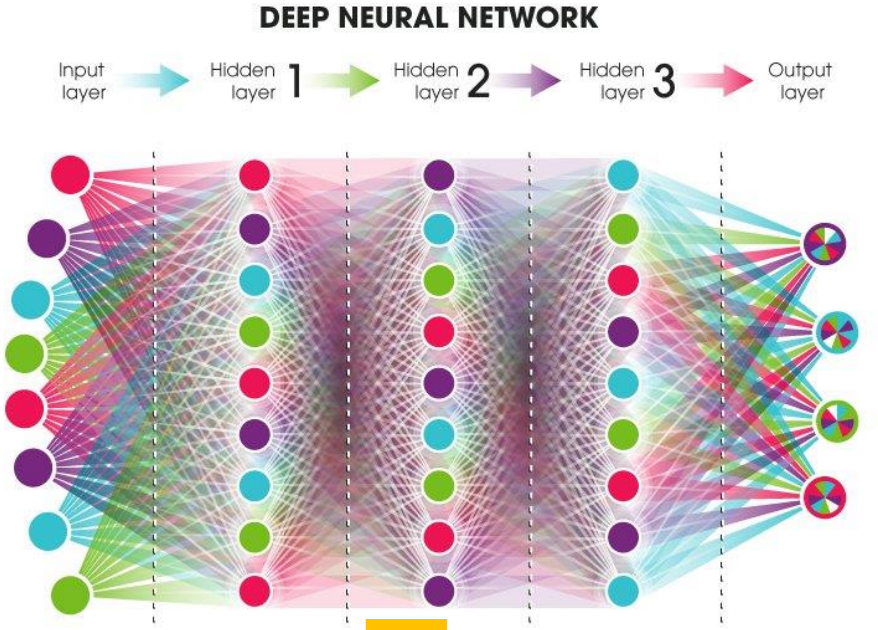


Development of the BDBT Coping Training System

Deep Neural Network Trained by using Nuclear Security Strategies Database

Strategies Database

- Initial Event 1
- Initial Event 2
- Initial Event 3
- Initial Event 4
- Initial Event 5
- Initial Event 6
- Initial Event 7
- Initial Event 8
- Initial Event 9
- Initial Event 10



Success Scenario

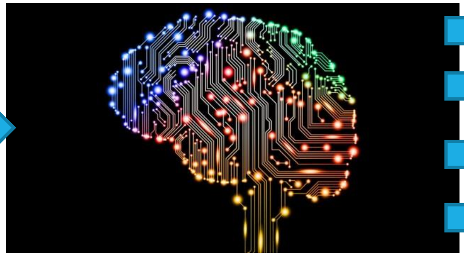
- Strategy Scenario 1-1
- Strategy Scenario 1-2
- Strategy Scenario 2-1
- Strategy Scenario 2-3
- Strategy Scenario 3-1

Failure Scenario

- Strategy Scenario 1-3
- Strategy Scenario 2-2
- Strategy Scenario 3-3
- Strategy Scenario 4-1
- Strategy Scenario 4-2

AI-based Strategy-support System using Deep Neural Network

New Event



- Alternative Strategy Scenario 1
- Alternative Strategy Scenario 2
- Alternative Strategy Scenario 3
- Alternative Strategy Scenario 4

Thank you for your kind attention!

東京大学 THE UNIVERSITY OF TOKYO

nem 原子力国際専攻 Nuclear Engineering and Management