Muay Thai Posture Classification using Skeletal Data from Kinect and k-Nearest Neighbors

Ketchart Kaewplee, Nirattaya Khamsemanan, Cholwich Nattee
Information and Communication Technology
for Embedded System Program
Sirindhorn International Institute of Technology
Thammasat University

***Abstract*—Muay Thai (also known as Thai boxing) is a martial art originated from Thailand. It is a discipline to make a bare-handed fight by efficiently combining eight limbs. Because of its uniqueness, Muay Thai has become popular internationally. Many people travel to Thailand in order to study Muay Thai. At the same time, many Muay Thai schools have been opened in many countries. To efficiently utilize limbs or parts of the body in the combat, Muay Thai trainees need to study and practice various postures. Each posture is a combination of multiple limbs to make an attack.**

All the Muay Thai practices are based on one-on-one training. The trainees learn postures and practice them for a period of time. In this paper, we present an idea to apply Microsoft Kinect for Muay Thai practicing support. To achieve the goal, we propose a technique that identifies a Muay Thai posture from each body movement of player. Using Kinect, we can collect a sequence of skeletal data from each body movement. We then extract a number of features, and apply 1-NN technique based on Dynamic Time Warping to identify Muay Thai postures. In our experiment, we use straight punch, swing punch and upper cut postures to test our system.

***Keywords* —Muay Thai Posture Classification , Muay Thai, Kinect**

I. Introduction

Nowadays, Muay Thai becomes a worldwide knowledge. There are a lot of documentaries, science shows, 3D animations or Youtube videos about Muay Thai. I have seen a lot of them. Therefore, it is curious that "Is it possible to make more correctness of understanding Mauy Thai by using nowadays technology?. Fortunately, Kinect from Microsoft is already come out. Kinect works well for dancing game. In my opinion, Kinect is really good for posture recognition.

Muay Thai is good knowledge for sport or real combat for a long time. Nobody try to develop this knowledge to be easier or more convenient to learn. Without having Mauy Thai course, there are just a few books that tell only the low details for all Muay Thai

First of all, we must be able to identify posture.

II. Related Works and Theories

Kinect has been used to help the trainee to practice the postures in many ways. In case of golf (Kinect based Golf Swing Score and Grade System Using Gaussian Mixture Model and Support Vector Machines [1]), Kinect is used to capture skeleton data from user. Then, extract velocity of the hand from the skeleton data. But, in our system, we extract the angle of shoulder, elbow and wrist. We believe that joint angles is better data to use to evaluate the postures correctness than velocity.

Kinect has been already used to practice matrial arts. In Game Based Approach to Learn Martial Arts for Beginners [2], Kinect is used to be a tool for learning material for Karate. Punching bubbles and obstacles in Karate style makes people fun and teach Karate at the same time.

The paper gives negative feedback to Kinect.

- User Height is not quite correct in far distance.
- User Hand Size is not quite correct.
- User Arm Span errors might happen.

The paper also gives positive feedback to Kinect.

- Kinect is a low-cost sensor for 3D sensor in market.
- Kinect does not require material that attached to the user.
- Kinect gives a lot and useful 3D data modeling.

Muay Thai has already had education knowledge in animation like Edutainment - Thai Art of Self-Defense and Boxing by Motion Capture Technique [3]. People can learn it everywhere by visiting to their website. 3D animations that show detail about Mauy Thai postures make people quite interested in Mauy Thai. People just only learn by watching 3D animation and cannot practice their own posture because they cannot compare their own posture to the animation.

Muay Thai posture practicing system must be in 3D learning system. 2D system is not enough. For example, Martial Arts in Artificial Reality [4], Player moves in area of 5x1 square meters area in real 3D world. But, it represents Kung Fu fighting 2D game. Fighting with 90 degree rotation may confuse the player and it cannot improve the real fighting skill.

A. Kinect

Kinect is from the Microsoft Xbox360 game player accessories can be recognized by the player (Facial Recognition) and allows players to control games through body movements of

the player directly (3D Motion Recognition). It is not required joystick anymore. It is able to recognize the sound of players (Voice Recognition) and it has function to play the sound. In This Paper, we present the posture recognition by using Kinect.

B. Skeleton Data

Kinect and Microsoft Visual Studio recieve a skeleton data that has 20 body parts (joints). We extract skeleton data to significant features (vectors) to recognition the posture. Each joint has its position that is represented by X,Y and Z. We extract skeleton data to significant features (vectors) to recognition the posture.

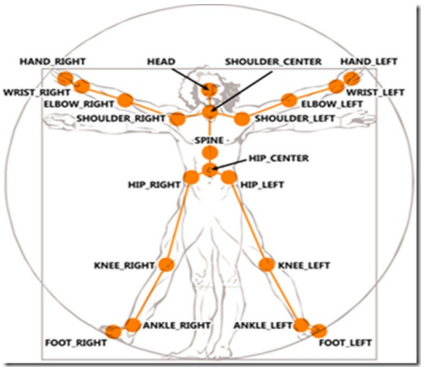

Fig. 1. Skeleton Data

C. Posture

A posture, S_i is represented by a sequence of frames $\{f_{ij}\}_{j=1}^{N_i}$ where f_{ij} is a vector containing 20 coordinates of body joints.

$$S_i = \{f_{i1}, f_{i2}, f_{i3}, \ldots, f_{iN_i}\} \tag{1}$$

$$f_{ij} = \begin{bmatrix} P_{hip\ center} \\ P_{spine} \\ P_{shoulder\ center} \\ \vdots \\ P_{foot\ right} \end{bmatrix} \tag{2}$$

Where P is position of body joints

D. Joint

The joints are defined by the Kinect skeletal tracking system. The joints have hierarchy that the Hip Center joint as the root and extends to the feet, head, and hands. In figure 2 diagram, the upper joint is parent joint and the lower joint is child joint. Every joint has its position in three dimensions (P_{joint}), where $P_{joint} = (X, Y, Z)$

E. Bone of Joint

Bones are specified by the parent and child joints that enclose the bone. For example, the Hip Left bone is enclosed by the Hip Center joint (parent) and the Hip Left joint (child).

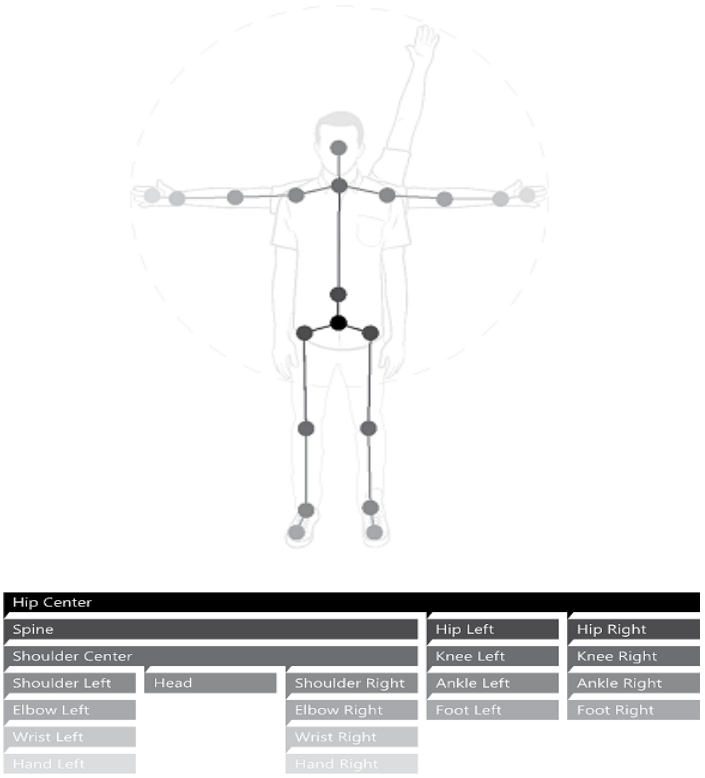

Fig. 2. Joint (msdn.microsoft.com)

Each bone can be represented by 3 bone vectors($\overrightarrow{X}, \overrightarrow{Y}, \overrightarrow{Z}$)

Example,

$$\overrightarrow{B}_{Hip\ left} = \begin{bmatrix} \overrightarrow{X}_{Hip\ left} \\ \overrightarrow{Y}_{Hip\ left} \\ \overrightarrow{Z}_{Hip\ left} \end{bmatrix} \tag{3}$$

$$\overrightarrow{Y}_{Hip\ left} = P_{Hip\ left} - P_{Hip\ center} \tag{4}$$

$$\overrightarrow{Z}_{Hip\ left} = \overrightarrow{Y}_{Hip\ left} \times \overrightarrow{X}_{Hip\ center} \tag{5}$$

$$\overrightarrow{X}_{Hip\ left} = \overrightarrow{Y}_{Hip\ left} \times \overrightarrow{Z}_{Hip\ left} \tag{6}$$

Where P is position of body joints

F. Absolute Bone Rotation

Each joint has Absolute Bone Rotation

$$R^{(a)}_{(joint)} = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \tag{7}$$

The Kinect camera is origin angle of X, Y, Z as show in figure. Absolute Bone Rotation is data that represents how each bone rotate relative to Kinect camera represent by 3×3 matrix.

For example, the kinect camera always has direction as origin axis that is represented by unit vector that is normalized by the sum of X vector ($1\overrightarrow{i}$), Y vector ($1\overrightarrow{j}$) and Z vector ($1\overrightarrow{k}$). The direction of kinect camera is represented by 3x1 matrix (X,Y,Z) from the direction of the unit vector. If the

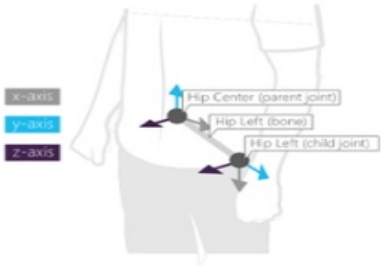

Fig. 3. Bone of Joint (msdn.microsoft.com)

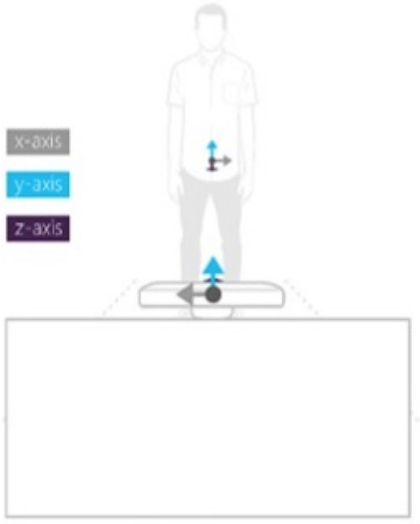

Fig. 4. Absolute Bone Rotation (msdn.microsoft.com)

user right wrist has a direction (the direction of Y vector of wrist bone) that is another 3x1 matrix (X,Y,Z), let it be V matrix. Let the direction of kinect camera be U matrix. The rotation relativity between user right wrist and kinect camera is represented by MV = U, where M is Absolute Bone Rotation ($R^{(a)}_{(joint)}$) that is in the form of 3x3 matrix.

G. Hierarchical Bone Rotation

Each joint has Hierarchical Bone Rotation

$$R^{(h)}_{(joint)} = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \tag{8}$$

Hierarchical Bone Rotation is data that represents how child X, Y, Z bone rotate relative to parent X, Y, Z bone that represent by 3×3 matrix.

For example, If the user right wrist has a direction that is 3x1 matrix (X,Y,Z), let it be V matrix. And if the user right elbow has another direction that is 3x1 matrix (X,Y,Z), let it be U matrix. The rotation relativity between user right wrist and user right elbow is represented by MV = U, where M is Hierarchical Bone Rotation ($R^{(h)}_{(joint)}$) of user right wrist that is in the form of 3x3 matrix.

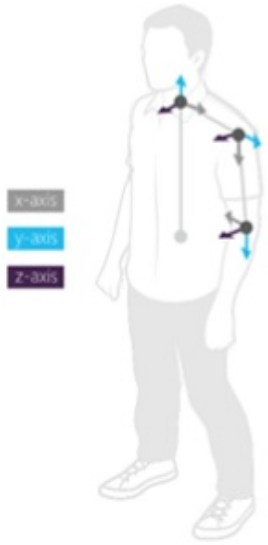

Fig. 5. Hierarchical Bone Rotation (msdn.microsoft.com)

H. Dynamic Time Warping

Dynamic Time Warping is an algorithm that we use to recognition the posture. we can tell the correctness (similarity between 2 data) by this algorithm.
Let

$$X = (x_1, x_2, \ \dots \ , x_M) \tag{9}$$
$$Y = (y_1, y_2, \ \dots \ , y_N) \tag{10}$$

We define a cost MxN Matrix, C where

$$C_{i,j} = d(x_i, y_j) \tag{11}$$

d is distance between x and y

We define the cost of the path (difference between 2 data),

$$\gamma_{i,j} = C_{i,j} + min\{\gamma_{i-1,j}, \gamma_{i,j-1}, \gamma_{i-1,j-1}\} \tag{12}$$
$$\gamma_{1,1} = 0 \tag{13}$$
$$\gamma_{i,0} = \infty \ (i = 1, 2, ..., M) \tag{14}$$
$$\gamma_{0,j} = \infty \ (j = 1, 2, ..., N) \tag{15}$$

Dynamic Time Warping of x and y

$$DTW(x, y) = \gamma_{M,N} \tag{16}$$

III. Proposed Method

First, we capture the whole video of a posture by Kinect camera. Then, we extract the array data of angle of shoulder, elbow and wrist. In our purpose, we would like to use standard array data from professional Muay Thai boxer. The trainee will compare his posture with the standard data and know the difference between his and standard data. Our system will

guide the correct way to improve the posture correctness. But, In our experment, we use 12 people data to compare each other to test our system.

1) A: Sequence of Frame
The first thing is capturing the whole video of a posture by Kinect camera. The participants do the posture and kinect capture it. This data is skeleton data frame sequence. We do not use this data and we need to extract the array data of angle of shoulder, elbow and wrist.

2) B: Pre Processing
During capturing posture data, There is unnecessary data at the first and the end of video that it is not the part of posture data. We have to trim it out.

3) C: Feature Extraction
We extract the array data of angle of shoulder, elbow and wrist from skeleton data frame sequence.

4) D: Sequence of Posture
The array data of angle of shoulder, elbow and wrist.

5) E: Posture Recognition
In our purpose, we would like to use standard array data from professional Muay Thai boxer. Our system will show the trainee difference between compare trainee posture and the standard data and guide the correct way to improve the posture correctness.

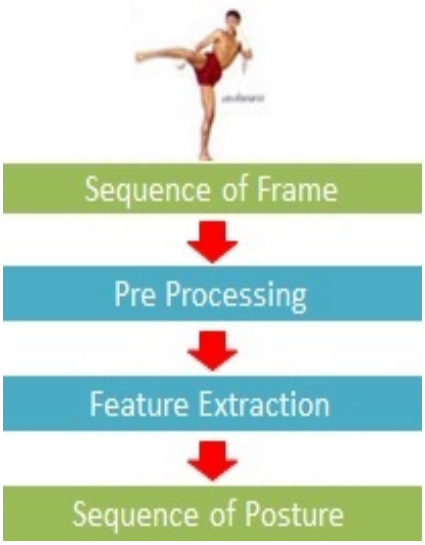

Fig. 6. Methodology

A. Feature Extraction

Given a posture S_i.
We extract 3 sequence of Hierarchical Bone Rotation.

$$\{R^{(h)}_{Shoulder\ right,k}\}^{N_i}_{k=1} \tag{17}$$
$$\{R^{(h)}_{Elbow\ right,k}\}^{N_i}_{k=1} \tag{18}$$
$$\{R^{(h)}_{Wrist\ right,k}\}^{N_i}_{k=1} \tag{19}$$

We use them for identification the type of posture.

Fig. 7. Posture Recognition

Compare posture S_i and S_j where

$$S_i = \{f_{i1}, f_{i2}, f_{i3}, \ \dots\ , f_{iN_i}\} \tag{20}$$
$$S_j = \{f_{j1}, f_{j2}, f_{j3}, \ \dots\ , f_{jN_j}\} \tag{21}$$

by Dynamic Time Warping,
We define a cost $N_i \times N_j$ Matrix, C where

$$C_{l,m} = d_{average}(S_{i,l}, S_{j,m}) \tag{22}$$

B. Posture Recognition

By the cost of the path, $\gamma_{l,m}$

$$\gamma_{l,m} = C_{l,m} + min\{\gamma_{l-1,m}, \gamma_{l,m-1}, \gamma_{l-1,m-1}\} \tag{23}$$

In our purpose, The cost represents the diffence of angle of shoulder, elbow and wrist between trainee and standard data.

γ_{N_i,N_j} is the summation value of the diffence between trainee and standard data.

In our system, we represent the similarity between trainee and standard data in precentage. So, we have to scale the data. we scale the $Ratio(\gamma_{N_i,N_j})$ to [0,1] first.

To make γ_{N_i,N_j} = [0,1]

$$Ratio(\gamma_{N_i,N_j}) = \frac{\gamma_{N_i,N_j}}{Number\ of\ Steps} \tag{24}$$

Then, we have to make it in percentage. So, we have to multiply by 100%. We want to show the trainee posture correctness. But, γ_{N_i,N_j} is the diffence of angle of shoulder, elbow and wrist between trainee and standard data. Therefore, we have to show value of 1 - $Ratio(\gamma_{N_i,N_j})$ instead.

To make γ_{N_i,N_j} in percentage of accuracy

$$\%\gamma_{N_i,N_j} = (1 - \frac{\gamma_{N_i,N_j}}{Number\ of\ Steps}) \times 100\% \tag{25}$$

$d_{average}(S_{i,l}, S_{j,m})$ is average value of our eucledian distance (formula 27) of shoulder, elbow and wrist between trainee and standard data.

$$d_{average}(S_{i,l}, S_{j,m}) =$$

$$\frac{d(R^{(h)}_{Shoulder\ right,i,l}, R^{(h)}_{Shoulder\ right,j,m})}{3} + \frac{d(R^{(h)}_{Elbow\ right,i,l}, R^{(h)}_{Elbow\ right,j,m})}{3} + \frac{d(R^{(h)}_{Wrist\ right,i,l}, R^{(h)}_{Wrist\ right,j,m})}{3} \quad (26)$$

In eucledian distance (formula 28), the eucledian distance always has value more than 0 and has maximum value (Maximum Distance) = $\sqrt{6}$ (formula 29). We want to scale the value.

So, We scale $d(R^{(h)}_i, R^{(h)}_j)$ = [0,1]

$$d(R^{(h)}_i, R^{(h)}_j) = \frac{d_{Eucledian}(R^{(h)}_i, R^{(h)}_j)}{Maximum\ Distance} \quad (27)$$

The Matrix value that we use to recognize the posture is Hierarchical Bone Rotation Matrix that $\overrightarrow{V} = M\overrightarrow{U}$, where M = Hierarchical Bone Rotation Matrix. $\overrightarrow{V} = \begin{bmatrix} \overrightarrow{X} \\ \overrightarrow{Y} \\ \overrightarrow{Z} \end{bmatrix}$ = parent bone vector direction. $\overrightarrow{U} = \begin{bmatrix} \overrightarrow{X} \\ \overrightarrow{Y} \\ \overrightarrow{Z} \end{bmatrix}$ = child bone vector direction.

$$M_1 = \begin{bmatrix} m_{1,11} & m_{1,12} & m_{1,13} \\ m_{1,21} & m_{1,22} & m_{1,23} \\ m_{1,31} & m_{1,32} & m_{1,33} \end{bmatrix} M_2 = \begin{bmatrix} m_{2,11} & m_{2,12} & m_{2,13} \\ m_{2,21} & m_{2,22} & m_{2,23} \\ m_{2,31} & m_{2,32} & m_{2,33} \end{bmatrix}$$

$d_{Eucledian}(M_1, M_2)$ is the eucledian distance of all components in Hierarchical Bone Rotation Matrix between trainee and standard data.

$$d_{Eucledian}(M_1, M_2) = \sqrt{\Delta m^2_{11} + \Delta m^2_{12} + \Delta m^2_{13} + \Delta m^2_{21} + \Delta m^2_{22} + \Delta m^2_{23} + \Delta m^2_{31} + \Delta m^2_{32} + \Delta m^2_{33}} \quad (28)$$

$$\Delta m_{ij} = m_{1,ij} - m_{2,ij}$$

The eucledian distance of two Hierarchical Bone Rotation Matrix always has value more than 0 and has less value than $\sqrt{6}$. In the example, we show the maximum value of eucledian distance that it is possible to be.

Eucledian distance value would be largest if the components in the first and second matrix are not in the same row and column. It could be 1 in $m_{1,12}$ and $m_{2,13}$ (as example). So, in the first row, in the root of eucledian distance(formula 29), it would be 0+1+1. But, if it is 1 in $m_{1,12}$ and $m_{2,12}$. In the first row, in the root of eucledian distance(formula 29), it would be 0+0+0.

In conclusion, all of the cases in eucledian distance of two matrix. The eucledian distance has the value between 0 and $\sqrt{6}$. So, $Maximum\ Distance$ must be $\sqrt{6}$.

$Maximum\ Distance\ Example$:

$$M_1 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} M_2 = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$d_{Eucledian}(M_1, M_2) = \sqrt{0+1+1+1+0+1+1+1+0} = \sqrt{6} \quad (29)$$

IV. Experiment

We capture Straight Punch, Swing Punch and Upper Cut posture from 12 people, 5 times per posture.

Fig. 8. Experiment

The postures that we use to test are basic posture of bare-handed combat. The steps in each posture are very obvious and they can be found in any martial art book.

A. Straight Punch

First, lift the arm parallel to the ground and push elbow backward to set the arm in the v shape. Secondly, launch the fist forward straightly.

B. Swing Punch

First, straighten the arm beside the body. Secondly, swing the arm to the front.

C. Upper Cut

First, drop the elbow down beside and parallel the body. Lift the fist straight to the front and parallel to the ground. Secondly, launch the fist upward.

V. Posture Recognition

Our system shows the result of user posture by the graph of similarity of user and professional data (the data of angle of shoulder, elbow and wrist which was extracted from skeleton data) in vertical axis with sequence (time) in horizontal axis. Therefore, user imagine and realized how the user compares the posture between professional and user.

Our experiment only show the example of the simple postures (Straight Punch, Swing Punch and Upper Cut). They are not totally Mauy Thai posture, so it is not necessary to use the angles of entire skeleton data.

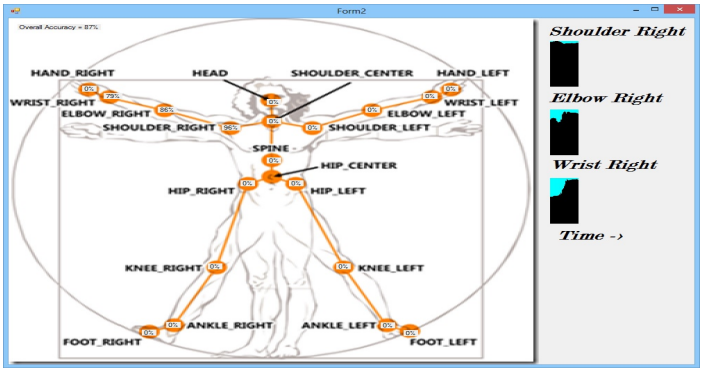

Fig. 9. Posture Recognition Result

VI. Result

To recognize the result we use K-nearest neighbour,

1-nn, first, we separate all posture data of all people in 5 sets (5 folds). In each fold, it contains Straight Punch, Swing Punch and Upper Cut posture data. Every data compares to all data in other folds to find the most similar data by using dynamic time warping. If the most similar data is the same posture, that data is marked as 1. If the most similar data is the different posture, that data is marked as 0. After marking all data, we calculate the percentage of mark(marked as 1) in each fold. And then, find the average percentage and standard error of all folds

3-nn and 5-nn, the whole process is just different from 1-nn in finding most similar data process. In 3-nn, we pick 3 most similar data for each data. Then, we decide the mark by the voting of that 3 data. For example, if the data is Straight Punch and its most similar data are Straight Punch, Straight Punch and Swing Punch. Its mark will become 1 by score of voting 2 of 3. In 5-nn, we pick 5 most similar data for each data and use the same method.

TABLE I. Recognition Rate

Algorithm	Fold1	Fold2	Fold3	Fold4	Fold5	Average $\pm$ Standard Error
1-nn	100%	100%	100%	100%	100%	100% $\pm$ 0%
3-nn	100%	100%	100%	100%	93.35%	98.67% $\pm$ 2.97%
5-nn	100%	100%	100%	100%	100%	100% $\pm$ 0%

The result shows that the system is quite accurate. Most folds are 100%.

However, we look clearly in all data and notice that ...

Straight Punch, most people forget to lift the arm first and punch immediately to the front. Some people lift the arm not parallel to the ground.

Swing Punch is the easiest posture to do. Everyone understands and do it well.

Upper Cut makes most people confuse because they do not know how to start and end the posture. Some people do Upper Cut in the wrong way (they upper cut to hit opponent beside their body not in the front), so the system detects it as Swing Punch.

The result shows that our system can recognize straight punch, swing punch and upper cut precisely. In our program, if 2 users do the same posture, it detects as 80% or higher. If they do the different postures, it detects about 60% or lower.

Kinect show little drawback that if body parts align in the line that Kinect camera cannot see through (Line of Sight problem), it cannot detect the body parts that stay behind. Consequently, the limitation in our system is the posture has to show the whole body parts that are used in posture in all of sequence of frame.

VII. Future Work

In The future of work, we will improve our system to be able to recognize posture in continuing time.

In the real situation, the boxer does not face the kinect camera directly and always use his posture quickly. We have to adapt our system to be able to recognize the posture automatically.

References

[1] Lichao Zhang, A Kinect based Golf Swing Score and Grade System Using GMM and SVM, 2012 5th International Congress on Image and Signal Processing (CISP 2012)

[2] Connsynn Chye, Tatsuo Nakajima, Game based approach to learn Martial Arts for beginners, 2012 IEEE International Conference on Embedded and Real-Time Computing Systems and Applications

[3] Suwichai Phunsa, Nawuttagorn Potisarn, Suwich Tirakoat, Edutainment - Thai Art of Self-Defense and Boxing by Motion Capture Technique, International Conference on Computer Modeling and Simulation

[4] Perttu Hmlinen,Tommi Ilmonen,Johanna Hysniemi, Mikko Lindholm, Ari Nyknen, Martial Arts in Artificial Reality, CHI 2005 PAPERS: Enhancing Virtual Spaces and Large Displays, April 27 Portland, Oregon, USA