Open Access Journal of Information Systems (OAJIS)

Available at http://is.its.ac.id/pubs/oajis/index.php

The Third Information Systems International Conference

Development Of Basic Taekwondo Training System Application Based On Real Time Motion Capture Using Microsoft Kinect

Nisfu Asrul Sani*, M Afif Hendrawan, Febriliyan Samopa

Information Systems Department, Institut Teknologi Sepuluh Nopember, Indonesia

Abstract

The high criminal rates in Indonesia have caused in increased interest in martial arts among the local populace, but the means to access training are largely unavailable. New technology may make it possible for civilians to study with a virtual coach without leaving their home. This allows them to learn proficiency with the detail oriented software that tracks proper form and range of motion. Using the motion capture features from a Microsoft Kinect, the user can develop basic taekwondo training and practice by using the virtual coach. Hopefully this progress in motion capture will afford a multitude of people the means to practice taekwondo for self defense or sport needs.

Keywords : motion capture; taekwondo; Microsoft Kinect

1. Introduction

Martial art training has become a priority to consider nowadays, and those living in urban environments have expressed a need for self-defense training. One of the most popular martial art disciplines is Taekwondo. Taekwondo is a mixed South Korean martial art which combines kicks and punches. The issue for most poeple are lack of time, the accessibility to a training center. A standard training video is not sophisticated enough to ensure proper effect to overcome those problems, a better system is needed to give a practice and supervise the user's progress.

The solution is motion capture technology which usually used for three-dimensional video game and movie animation, but this technology is quite expensive for personal users. Another cheaper solution is using the camera to capture the motion such as Microsoft Kinect. Kinect is a motion sensor camera which can pick up both a user's frame and range of motion. The camera has three sensors, one for video, and two infrared sensors for emitting and receiving [2].

Using this technology from Microsoft, Kinect could be developed for taekwondo self training with which progress can be monitored.

2. Literature Review

2.1 Microsoft Kinect

Microsoft Kinect is a line of motion sensing input device which developed by Microsoft. Two infrared sensor technology enables Kinect to capture the depth of an object in front of it [2]. There are 2 types of Kinect, Kinect for Xbox 360 and Kinect for Windows. Kinect from XBOX 360 is a sensor which is devoted for XBOX360, on the other hand Kinect for windows is specialized for windows application based usage. Technically, the basic difference are near mode and seated mode sensor that only belongs to Kinect for Windows. Near mode, provides Kinect to capture an object from 0.4 meters to 3 meters and 400mm depth. Standard feature for both type of Kinect is seated mode which provides motion sensing capture to an object with 1.2-4 meters and object's depth 800mm .Kinect 360 can be applied to windows application based without using near mode. Both Kinect is the first technology (Kinect v1).

2.2 Microsoft Kinect Capacity

Kinect may simultaneously track up to six people, including two active players for motion analysis. Kinect v1 capable to capture 20 of human's joints using its sensor. The Kinect v1 camera can project several format of video resolution,

1. 640x480 pixel with 30 fps maximum using red, green, blue (RGB) format.
2. 1280x480 pixel with 12 fps using RGB format.
3. 640x480 pixel with 15fps using YUV format.

The depth sensing sensor can project the depth of an object with following format with 30 fps maximum [2] in three video resolution,

1. 640x480 pixel.
2. 320x240 pixel.
3. 80x60 pixel.

2.3. Kinect for Windows Software Development Kit (SDK)

Kinect for Windows (SDK) is an official library provided Microsoft Windows to developed Kinect application based on Windows platform. Kinect for Windows SDK works as an interface between developed application and Kinect. Using application programming interface (API) in SDK helps to access features on Kinect. API will forward the command based to Kinects driver.

2.4. XNA Game Studio

XNA Game Studio is a framework which designed to make a game based Microsoft Windows, XBOX 360 and Windows Phone. XNA connects to Microsoft Visual Studio in framework and tools. Framework from XNA consists of specialized libraries in game developing. In package installation of XNA Game Studio consist of tools to manage audio and graphic which is needed in a game.
XNA Game Studio framework based on Microsoft .NET platform, so XNA Game Studio can use its own frame work or .NET framework. Standard programming language which is used by XNA Game Studio is C#, theoretically Visual Basic (VB) and C++ can be used also.

2.5. Software

Some softwares are needed in development of Basic Taekwondo Training System Application. The softwares which are used in this development are : 1) Virtual Studio 2010 as main IDE. 2) Visual Studio 2013 as a tester. 3) 3ds Max modelling 3 dimensional model.

3. Research Method

3.1. Literatur Review

At this phase, a study progress to survey the object and the technology for the research. The boundaries of the study are: 1) General Taekwondo movement. 2) Kinect's ability and limitation. 3) Taekwondo's movement which perfectly captured by Kinect 4) XNA Game Studio as a framework to develop an application.

3.2. Analysis

At this phase, a problem analysis based on the research such as the of Taekwondo's movement which perfectly captured by Kinect and the limitation of Kinect itself.

3.3. Analysis and Application Designing

A beginning analysis and the system requirement to find the main problem in the research. Based on literature review phase, minimal business requirement from training application are, Shows the user body position [3], Capture user's movement [3], Recognize the user's movement [4], User has a virtual trainer as a coach [4]. User can give feedback exercise directly [4], and Monitor the exercise progress .[4].

The other application requirement are Capture user's movement, Read three dimentional structure object, User's join framework transformation, Capture and shows depth stream views, Three dimensional model animation as a trainer, Movement's name, Movement's recording, Playback record feature, Movement's match, and Exercise report

4. Implementation

4.1. Implementation Process Based on Movement.

The process of the movement is made from application analysis requirement to capture user's movement, showing depth stream views and 3 dimensional animation. In these phase the user's movement will be captured by Kinect, then transmitted into data framework which is consist of 20 joints information in each frame of 3 dimensional room of Kinect. Data framework will be transformed by XNA to change 3 dimension of Kinect into XNA framework. After transformed, framework information will be projected to 3 dimension framework which follows every user's movement directly. Depth stream also takes part to ease the position in sensor reaches and recognize the real movement from user.

4.2. Implementation Process Based on Movement Recording

Implementation process based on movement recording is made based on application requirement (movement's name and movement's recording). This method helps to tag movement's name in movement's recording file. Implementation of recording function using serialize and deserialize methods in binary formats. At first, the writer will use recording methods, note the movement coordinate joints in each frame separately using coma as a separator, but this methods is ineffective and influence the replay features. That problems occurs because the frame information and joints captured separately. Using the serialize and deserialize function it may ease the implementation of replay features because the frame and joints are recorded in an information. Implementation of serialize can be seen in Code Snippet 1.

The result of movement's recording will be used as user exercise reference. The first movement reference are *Yop Jireugi* or side punch which is taken from set movement of *Taebaek* and *Are Maki*, a basic movement of Taekwondo.

```
private void Serialize(Skeleton[] skel)
    {
        try
        {
            BinaryFormatter binF = new BinaryFormatter();
            binF.Serialize(this.fs, skel);
        }
        catch (Exception e)
        {
            e.ToString();
        }
```

Fig. 1. Code Snippet 1. Implementation of Serialize Methods

4.3. Gesture Matching.

In the implementation of this process there is a replay function, matching movement and the calculation results of the exercise. Replay function to read binary files a result of serialize method which then translates it into information frames and joints user. This implementaionmethod using deserialize. Deserialize method can be seen in Code Snippet 2.

```
private Skeleton[] Deserialize()
{
  Skeleton[] record;

  BinaryFormatter binF = new BinaryFormatter();
  record = (Skeleton[])binF.Deserialize(this.fs);
  if (this.fs.Length == this.fs.Position)
  {
    this.curState = GameState.hasilScreen;
  }
```

Fig. 2. Code Snippet 2. Implementaion of Deserialize Methods

Deserialize function will control 3 dimension model with frame and joints information. The next phase is implementation of motion matching movement which match the user movement and with the avatar. this matching process is based on the coordinate difference between the user's movement coordinate and the avatar's. Based on test trial of 300 frames in one movement's set with 2.7 meters users and height from the sensor 0.84 meters, the difference of coordinate is 0.25 meters. This result will be a tolerance value from user's movement caused by limitation of Kinect itself. The last step is calculation report of the exercise using this following formula. Total correct movement is the correct movement from the user and total exercise frame is the whole movement from a set of exercise movement.

$$Exercise\ Result\ (\%) = \frac{Total\ of\ Correct\ Frame}{Total\ of\ Excercise\ Frame} x\ 100 \quad (1)$$

4.4. Tolerance Test Calculation

Tolerance test calculations performed to obtain the value of tolerance in the calculation of assessment exercises. Giving tolerance due to the limitations of the sensor cannot read the joint completely and limitations of the sensors in the read speed of movement of the user. To get the values of tolerance, conducted experiments comparing the results of recording the movement of a reference to direct the movement of the user. This experiment was performed on 20 joints are read by sensors on one set of movements. From these experiments obtained an average difference (Δ) coordinates values of X and Y.

Values X and Y coordinate of a reference in the calculation results of the exercise. Experimental results can be seen in Table 1

Table 1. Tolerance Test Calculation Result

No	Joint	Δ X (m)	Δ Y (m)
1	*AnkleLeft*	0,132466784	0,009946713
2	*AnkleRight*	0,014825913	0,008429622
3	*ElbowLeft*	0,138961068	0,129848899
4	*ElbowRight*	0,084784746	0,025486557
5	*FootLeft*	0,133443789	0,014986829
6	*FootRight*	0,017974592	0,010257875
7	*HandLeft*	0,276476692	0,214657378
8	*HandRight*	0,082490647	0,034281892
9	*Head*	0,083178614	0,029824833
10	*HipCenter*	0,078666355	0,030589551
11	*HipLeft*	0,077921526	0,030758907
12	*HipRight*	0,078176462	0,029872683
13	*KneeLeft*	0,109570465	0,014100891
14	*KneeRight*	0,046378461	0,007901217
15	*ShoulderCenter*	0,082412972	0,029171146
16	*ShoulderLeft*	0,083046884	0,03528456
17	*ShoulderRight*	0,08385806	0,025119762
18	*Spine*	0,080028464	0,029222661
19	*WristLeft*	0,237801848	0,199782126
20	*WristRight*	0,078509614	0,019537492

4.5. Raw data matching

Before real test is conducted, we evaluate differences between recorded movement and real movement. Calculation has been performed to find the average value of ΔX, ΔY, and ΔZ. Then performed experiments to determine whether the value Δ resulting from previous accuracy test as a reference tolerance calculations. Accuracy is judged by the results of the calculation of the system compared to the expert assessment.

Table 2. Example of evaluation

Recorded				Actual			
	X	Y	Z		X	Y	Z
120@sLeft:	-0.1456	0.3846	2.814	254@254@sLeft:	-0.1514	0.392	2.815
eLeft:	-0.1633	0.1071	2.7754	eLeft:	-0.1658	0.1214	2.7821
wLeft:	-0.1776	-0.1155	2.7071	wLeft:	-0.1754	-0.0997	2.7121
hLeft:	-0.1786	-0.1926	2.6902	hLeft:	-0.1757	-0.1966	2.692

Comparison has been performed by evaluate three elemen X, Y, and Z. each element will be examined the delta (difference).

Table 3. comparison between recorded and actual

part	recx	actx	dtx	recy	acty	dty	recz	actz	dtz
120@sLeft:	-0.1456	-0.1514	0.0058	0.3846	0.392	0.0074	2.814	2.815	0.001
121@sLeft:	-0.1454	-0.1513	0.0059	0.3847	0.392	0.0073	2.8141	2.8151	0.001
122@sLeft:	-0.1457	-0.1515	0.0058	0.3848	0.3921	0.0073	2.8145	2.8151	0.0006
123@sLeft:	-0.146	-0.1511	0.0051	0.384	0.3924	0.0084	2.8152	2.8153	0.0001

part	recx	actx	dtx	recy	acty	dty	recz	actz	dtz
124@sLeft:	-0.1456	-0.1511	0.0055	0.3832	0.3924	0.0092	2.8153	2.8153	0
125@sLeft:	-0.1456	-0.1511	0.0055	0.3831	0.3926	0.0095	2.8152	2.8155	0.0003
126@sLeft:	-0.1457	-0.1511	0.0054	0.3831	0.3925	0.0094	2.8157	2.8155	0.0002
127@sLeft:	-0.1457	-0.1515	0.0058	0.3834	0.3926	0.0092	2.8156	2.816	0.0004
128@sLeft:	-0.1455	-0.1514	0.0059	0.3832	0.3922	0.009	2.8155	2.816	0.0005
129@sLeft:	-0.1454	-0.1515	0.0061	0.3832	0.3922	0.009	2.8155	2.8163	0.0008
130@sLeft:	-0.1455	-0.1516	0.0061	0.383	0.3921	0.0091	2.8156	2.8166	0.001
131@sLeft:	-0.1455	-0.1516	0.0061	0.3825	0.3921	0.0096	2.8151	2.8166	0.0015
132@sLeft:	-0.1456	-0.1514	0.0058	0.3825	0.3919	0.0094	2.8155	2.8165	0.001
133@sLeft:	-0.1457	-0.1518	0.0061	0.3825	0.3918	0.0093	2.8155	2.8174	0.0019
134@sLeft:	-0.1457	-0.1517	0.0060	0.3824	0.3917	0.0093	2.8156	2.8171	0.0015
135@sLeft:	-0.1456	-0.1517	0.0061	0.3827	0.3911	0.0084	2.8156	2.817	0.0014
136@sLeft:	-0.1456	-0.1519	0.0063	0.3828	0.3907	0.0079	2.8156	2.8171	0.0015

5. Testing

Functional and non-fuctional features will be tested on application testing. From functional test, the research will achieve a result from exercise which is done by the expert and the user. On the other sid, non-functional test will be run to achieve application performance and minimum requirement to run the application.

5.1. Functional Test

Expert judgement at functional test based on three main criteria, false, enough, and true. Otherwise, application judgment at functional test bases on exercise result. Movement manipulation test should be done to discover the tolerance given by application judgment.

5.1.1. Yop Jireugi Movement

Functional test at *yop jireugi* movement executed in 120 times to get judgement from expert. The result of *yop jireugi* movement can be seen in Table 4.

Table 4. Test Result for *Yop Jireugi* Movement

No	Expert Judgment	Application Judgement
1	False	68,84 %
2	Average	79.72 %
3	True	83.95 %

The next phase is manipulation test based on coordinate z (depth) at *yop jieugi* movement. Manipulation test focus is punch movement. Testing manipulation can be seen in Table 5. To estimate wrong movement, a test has to be done in an extreme condition. Extreme condition is manipulation movements which happen purposely to achieve expert judgement. Testing manipulation based on extreme condition at *yop jierugi* can be seen in Table 6.

Table 5. Manipulation Based on Z Coordinate of *Yop Jireugi* Movement

No	Z Manipulation	Application Judgement
1	+10˚	85.61 %
2	+45˚	85.51 %
3	-10˚	85.51 %

Table 6 Extreme Condition Test at *Yop Jireugi* Movement

No	Extreme Condition	Result
1	Counter clockwise punch movement	30.31%
2	Front punch (*momtong jireugi*)	64.41%

5.1.2 Are Maki Movement

Functional test on *are maki* movement executed in 80 time to get a judgement from the expert. The result of *are maki* movement test can be seen in Table 7. *Are maki* movement doesn't need manipulation test on z coordinate because in *are maki* movement, it is impossible to do it. Next phase is extreme condition test based on *are maki* movement. The result of *are maki* is 15.75%

Table 7 Test Result for *Are Maki* Movement

No	Expert Judgement	Application Judgement
1	False	55.21 %
2	Average	80.20 %
3	True	83.28 %

5.2. Non-Functional Test

The Environment which runs the test can be seen at Table 8. The usage of four environment test is to test performance test to CPU consumption, memory consumption and frame render each second (fps). Test result on each environment can be seen at Table 9. Whole circumstance of the test is 1 minute using Windows 7.

Table 8 Environment Test

Environment Test	ET1	ET2	ET3	ET4
CPU	Intel Core i7 Q740 @1.73 Ghz	Intel Core i5-3330 @ 3.00 Ghz	Intel Core 2 Duo E7500 @ 2.93 Ghz	Intel Core 2 Duo E7500 @ 2.93 Ghz
RAM	8 GB	4 GB	4 GB	2 GB
GPU	NVIDIA GeForce GT425M – 2GB	NVDIA GeForce GTX 650 Ti – 1.5 GB	Intel On Board	Intel On Board

Table 9 Non-Functional Test Result

ET	Condition	CPU	FPS	Condition	CPU	FPS	Condition	CPU	FPS
1	Normal	20% - 30%	60 fps	Recording	20% - 50%	60 fps	Exercise	20% - 30%	60 fps
2	Normal	20% - 30%	60 fps	Recording	20% - 50%	39 fps	Exercise	20% - 30%	60 fps
3	Normal	60% - 90%	57 fps	Recording	60% - 80%	57 fps	Exercise	70% - 90%	54 fps
4	Normal	90% - 100%	54 fps	Recording	90% - 100%	54 fps	Exercise	90% - 100%	52 fps

6. Conclusion

From the conducted research, there are some things that can be inferred

1. From the results of trials that have been conducted basic taekwondo training application system can be utilized to perform the exercises taekwondo with restrictions, Movement should be facing sensor.Requires tolerance in the calculation of the coordinates of the user joints. Z coordinate or depth movement in three dimensional space is not taken into account by the application. Speed users must comply with the reference model (avatar). Calculation of the fingers was not done bacause the limitation of sensor and SDK. According to expert assessmentm the user's movement can be said to be true if the end result of exercise a minimum of 83.95 % for *yop jireugi* movement and 83.28% for *are maki* movement.
2. In addition to functional limitations in point 1, of the test results obtained non-functional limitations. The optimal distance between user and sensor is 2.7 meters. Application cannot perform automatic duplication if there are names the same movement. The minimum specification of computer equipment to run the application. CPU : Intel Core i5. GPU: Intel on board. RAM: 2 GB (Kinect Minimum).

7. Future Work

In Basic Taekwondo Training application development system, there are still many shortcomings in the process of development. For that there are several things that can be used as a reference for further application development.

1. Application basic taekwondo training system is a study to test whether Kinect sensor is capable to calculate each frame in the case of self-defense movements, especially Taekwondo. So that the future can be tested for other martial movements or other movements such as gymnastics or dance.
2. The use of the virtual trainer to use avatars become a major problem due to the movement of matching avatar cannot move exactly with the data frame. With this problem needs to be done further research how to use avatars with better in the reading of the data frame. 3. The emergence of new technologies that kinect sensor Kinect for Windows 2.0 or Kinect for Xbox One and Kinect for Windows SDK 2.0 can be used for further research because of his ability in detecting human joints 25 points.
3. The new technology of Kinect, named Kinect for Windows 2.0 and Kinect for XBOX One, can be used on the next reasearch because the ability to sense 25 human joints.

References

[1] Muhammad, D. (2013, Desember 27). *Kapolri Klaim Kriminalitas 2013 Menurun*. Retrieved Maret 26, 2014, from Republika Online: http://www.republika.co.id/berita/nasional/hukum/13/12/27/mygtns-kapolri-klaim-kriminalitas-2013-menurun

[2] Catuhe, D. (2012). *Programming With The Kinect for Windows Software Development Kit.* Washington: Microsoft Press.

[3] Vij, M., & Dawn, S. (2014). Corrective Self Defense Training Unit using sensing of Kinect maps. *International Journal of Computer Applications*, 8-11.

[4] Nike. (2012). *Nike+ Kinect Training*. Retrieved Desember 12, 2014, from Nike: http://www.nike.com/us/en_us/c/training/nike-plus-kinect-training