 doi:10.20944/preprints201806.0367.v1

Article

A Real-time Calibration Method of Kinect Recognition Range Extension for Media Art

Sunghyun Kim [1], and Won-hyung Lee [2,*]

[1] Department of Multimedia Art & Technology, The Graduate School of Advanced Imaging Science. Chung-ang University, Seoul 06974, Republic of Korea; kimkshgugu@naver.com

[2] Department of Multimedia Art & Technology, The Graduate School of Advanced Imaging Science. Chung-ang University, Seoul 06974, Republic of Korea; whlee@cau.ac.kr

* Correspondence: whlee@cau.ac.kr

Abstract: Kinect is a device that has been widely used in many areas since it was released in 2010. Kinect SDK was announced in 2011 and used in many other areas than its original purpose, which was a controller for gaming. In particular, it has been used by a number of artists in digital media art since it is inexpensive and has a fast recognition rate. However, there is a problem. Kinect create 3D coordinates with a single 2D RGB image for x, y value - single depth image for z value. And this creates a significant limitation on the installation for interactivity of media art. Because the Cartesian XY coordinate and the spherical Z coordinate system are used in combination, depth error depending on the distance is generated, which makes real-time rotation recognition and coordinate correction difficult above coordinate system. This paper proposes a real-time calibration method of Kinect recognition range expansion for useful application in the digital media art area. The proposed method can recognize the viewer accurately by calibrating a coordinate in any direction in front of the viewer. 3,400 datasets witch acquire from experiment were measured as five stances: the 1m attention stance, 1m hands-up stance, 2m attention stance, 2m hands-up stance, and 2m hands-half-up stance, which were taken and recorded every 0.5 sec. The experimental results showed that the accuracy rate was improved about 11.5% compared with front measurement data according to Kinect reference installation method.

Keywords: Kinect; Depth calibration; RGB-D; Media Art; Skeletal Joint Data

---

1. Introduction

With the advancement of Internet computer technology, studies on methods that make the interaction between computers and humans simple, fast, and easy to understand have been conducted continuously. In particular, human tracking in the video game industry has been developed as a control method [1], and it has now been easily and rapidly applied through a number of study results. This has led to the development of a gesture interface followed by the realization of the natural user interface concept [2], which has been studied since 1971.

Human tracking has been studied continuously as a new user interface method. As a result, various products for commercialization and equipment for experiments have been launched in the market. More recently, with the advancement of infrared (IR) sensors and various image-processing techniques, much attention has been paid to apparatus without applying equipment to the human body.

The Microsoft Kinect has been recognized as the most well-implemented and applied device for realizing a gesture interface using human tracking as a new input device.

Kinect is the world's fastest-selling equipment, as certified by Guinness World Records, after it was launched in the market as an accessory for the Xbox 360 console in December 2010. The production of Kinect was suspended from October 2017, but the core technology RGB single-depth motion capture has been utilized in the Microsoft HoloLens.

Kinect is a motion and voice recognition device based on RGB single-depth images [3]. The original purpose of Kinect development was as a game controller. Kinect shows a good recognition rate even though it is an inexpensive sensor. Not long after Kinect was launched, the equipment was hacked, and the software development kit (SDK) was released. Since then, it has been widely used in many areas, including art [4,5,6], application programs [7,8,9,10,11,37], computer vision [12,13,14], and medical engineering [15,16,17], due to its inexpensive motion recognition function.

However, since the RGB single-depth image-based coordinate system in Kinect employs the Cartesian X-Y and spherical depth coordinate systems in a mixed manner, it has a shortcoming in that the equipment has to be located at the front always [18].

As a correction method to overcome the above installation limitation, several studies using a fixed-coordinate correction method [19], multi-cameras [20,21,22], and Kinect fusion [23] have been proposed, but these methods employ a polar coordinate system, which is different from the XYZ coordinate system used in the Microsoft Kinect Software Development Kits (MS Kinect SDKs). Microsoft has released Kinect V2 for Kinect for Windows and Xbox One, which improved the recognition problem [24]. However, this also had to fix the installation location of the equipment and subject [25,26].

As described above, Kinect is a low-cost motion recognition sensor and originally game control device. Although it has been widely used in various areas rapidly, it is based on RGB single-depth images, which created a problem due to installation requirements. In particular, this problem has become serious in the game and art areas because Kinect must be straight from Kinect – Screen - User, and the coordinate value will change greatly when moving outside the reference installation method [18]. And this is a big problem in games and media arts where fast production and accessibility matters [1].

Figure 1. Indoor installation media art using one Kinect

In particular, there is an installation problem between media art and device in the case of indoor media art. For accurate recognition by Kinect, Kinect has to be installed in the front and start recognition of human bodies and it is recommended to measure at a distance of 2.5 ~ 3M [18]. However, it is realistically difficult to install Kinect in front of works in media art exhibition centers due to the continuous entry and exit of audience members.

In addition, a real-time correction method that is fast and simple without coordinate system transformation is needed for the freedom of composition and installation formation approach preferred by artists.

To achieve this, this paper proposes a sideline installation solution for Kinect that can perform real-time calibration based on the user's body depth values using one Kinect and does not change the coordinate system for media game art. Additionally, errors among the original coordinate values of Kinect sensor, twisted coordinates, and calibrated coordinates are compared.

2. Related research

According to the in-home depth camera calibration [18], Kinect recognizes user using the unique coordinate system, which is made by combining a 2d Cartesian coordinate system using RGB camera and a 3d spherical coordinate system using an IR camera. A depth-column-based 3D coordinate system is excellent for single plane application when tracking objects in front of it. This is the reason that Kinect enables good quality motion tracking at a small size and low price. But according to related research [18,19,25,36] it has disadvantages of suffering from many errors when tracking from the side. To overcome this, a method such as multi-tracking or coordinate system transformation has been used [26,27,28,29]. A polar coordinate transformation method requires not only high-level understanding from programmers but also many computation processes [30,31,32,33,34] and an accurate installation angle [20,35].

As shown in (1), the rotation transformation requires a rotation angle θ. According to P. Kumar's paper [36], the coordinates of the Kinect do not provide the rotation angle θ, and the rotation angle θ is derived from two different 3D transformations.

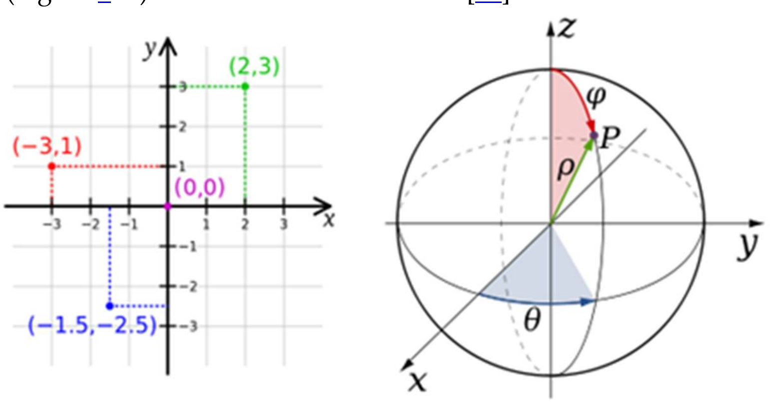
$$\begin{pmatrix} x\prime \\ y\prime \end{pmatrix} = \begin{pmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \quad (1)$$

2.1. Tracking and gesture recognition using RGB cameras

If a gesture is recognized using a method based on image recognition, motions can be utilized without the need for wearing a sensor or hardware at a specific area, expanding the utilization range. Although there has been a case of inputting the hand motion of a user via a mouse using general RGB cameras, it is not easy to separate a user's movements from background images in complex environments[43], which is a drawback of the degradation in recognition rate. To overcome this, the background and user are separated through the IR light-emitting element and IR cameras to increase the recognition rate, and gestures made by fingers are recognized in real space and utilized as inputs. More recently, the image-based motion recognition rate has improved significantly since Microsoft has configured the above functions in a single module and Kinect Microsoft released in the market as a prototype has been utilized.

2.2. Types and features of the coordinate system used in image recognition based 3D tracking

The coordinate system is a way to mark the direction correctly. There are rectangular coordinate system and polar coordinate system. The difference between the two is based on the distance in each direction from the origin, the distance and the angle from the origin. Kinect is a mixture of two coordinate systems. It uses rectangular coordinate system(Figure 2. a) to obtain the X and Y coordinates using the RGB camera image and the 3D polar coordinate system - spherical coordinate system(Figure 2. b) to obtain the Z coordinate [18].

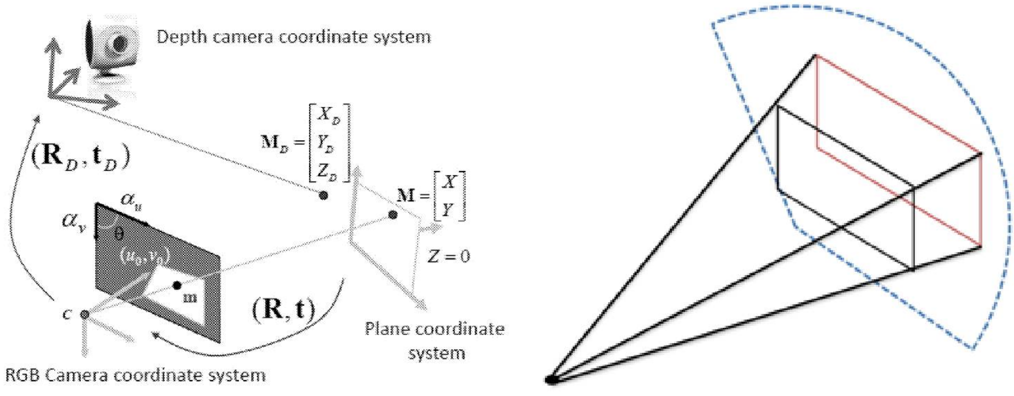


(a) (b)

Figure 2. Rectangular coordinate system (a) Spherical coordinate system (b)

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 24 June 2018 doi:10.20944/preprints201806.0367.v1

2.3. Depth-column-based 3D coordinate system



(a) (b)

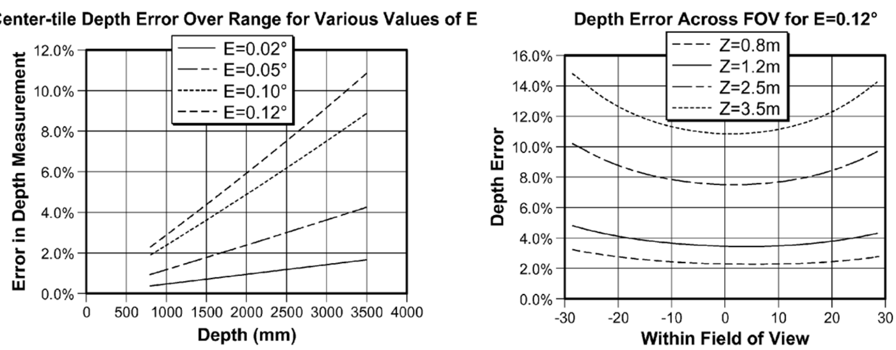
Figure 3. An illustration of a plane of the calibration object relative to a coordinate system of the depth camera and relative to a coordinate system of the RGB camera [18 (pp.14,23)] (a) Structure diagram of depth-column-based 3D coordinate system (b)

The depth-column-based 3D coordinate system (Figure 3.) is a unique coordinate system used in Kinect. It is made by combining the values in two different coordinate systems [21].

The Z value is provided from the Euclidean distance from the origin r value in the 3D spherical coordinates. Combine this with the 2D rectangular coordinate system and use it as 3D coordinates.

This unique coordinate system of Kinect has a depth error based on the distance and FOV value. (Figure 4.)

With this depth-column-based 3D coordinate system have many limitations on the installation and utilization of single Kinect.



(a) (b)

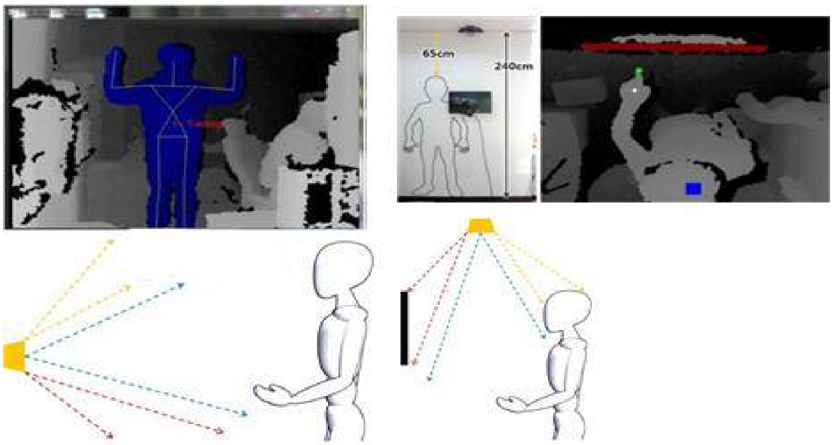
Figure 4. Depth Error Rate Depending on Device - User's Angle(a), Depth Error Rate Depending on Device - User's Distance(b) [18(pp.17)]

Despite these disadvantages, depth-column-based 3D coordinate system is still used because the size of the equipment can be reduced, without wearing equipment and remote sensor with fast and accurate measurement in limited installation unlike existing tracking equipment such as RGB cameras, motion trackers and controllers. According to related research, Kinect is a high-quality motion tracking device at low cost.

doi:10.20944/preprints201806.0367.v1

2.4. Tracking and application method of Kinect

Kinect from Microsoft is a depth-column-based 3D tracking device that has been widely used due to its low price and high performance compared to existing tracking equipment following its release in Nov. 2010.



(a) **(b)**

Figure 5. Reference Front Tracking (a) and Axis Conversion Top Tracking (b)

According to Kinect patents IN-Home Depth Camera Calibration [18], basic tracking method is to place the equipment at least 2 m outside, in front of the user, as shown in Figure 5a. Obtaining the best recognition rate is recommended when using Kinect. And this tracking method does not have a shade area for most human motion created. However, Kinect has a significant limitation in terms of its installation environment or usage due to the characteristics of its installation, which require the screen and user angle to be matched [7,18 (pp.25)].

There is a method by installing the equipment at a right angle (90°) in the angular direction as shown in Figure 5b. [7,18]. This can solve some installation limitations, however, there is no solution to the depth error shown in figure 4.

2.5. Gesture interface recognition and subject rotation recognition

Previously, various methods have been developed to facilitate gesture recognition from various angles using Kinect, such as multilayer perception (MLP), artificial neural networks (ANN), and support vector machines (SVM). Table 1 summarizes the approach method, datasets, and accuracy of the previous study results that have attempted calibration using other methods.

Many studies have been conducted on rotation recognition, and most of them have been applied to recognition of gesture recognition or sign language. The Depth-column-based 3D coordinate system was used 3 and 6, and the 6th item was measured by converting the angle between the user and Kinect into 0, 45, and 90 degrees.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 24 June 2018 doi:10.20944/preprints201806.0367.v1

Table 1. Summary of the related work in comparison to the proposed methodology

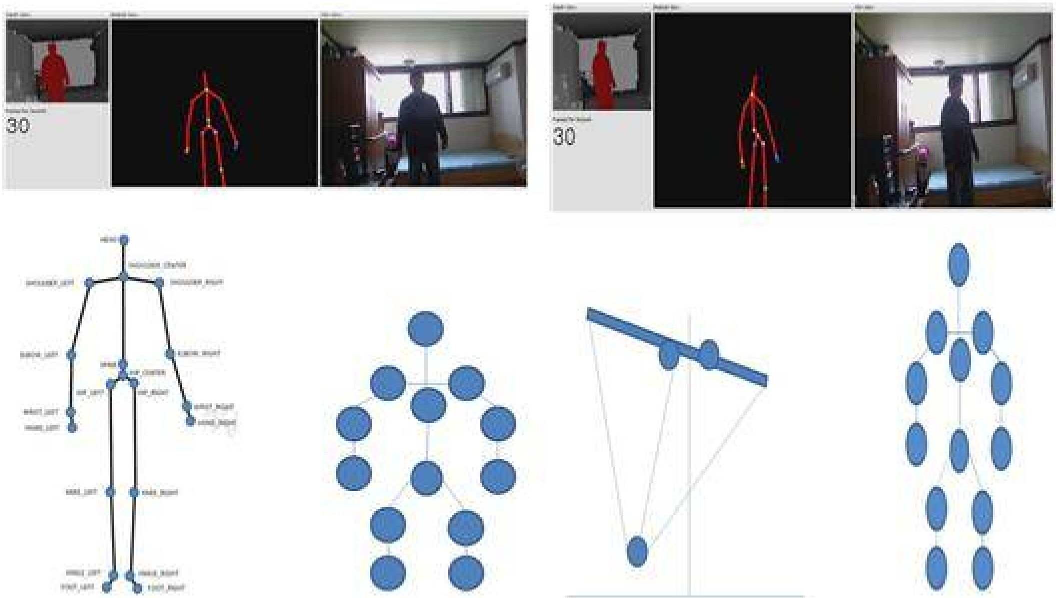
Author & Year	Approach	Dataset	Accuracy (%)
Patsadu et al. [38], 2012	Kinect joint positions, SVM, Decision Tree, NN, Naive Bayes	3 gestures	93.72%
Monir et al. [39], 2012	3D joint positions, angular features, priority matching	4 postures	89.1%
Almeida et al. [40], 2014	Hands shape, movement, position, SVM	34 BSL signs	80%
Lim et al. [41], 2016	Serial particle filter, covariance matrix	50 ASL signs	87.33%
Uebersax et al. [42], 2011	Hand orientation, MM, DD	7 ASL alphabets	88%
Pradeep et al. [36], 2017	3D hand joints, Affine Transformation, angular, curvature, velocity, dynamic features, HMM, SVM	30 ISL signs	83.77%
Proposed Methodology	3D hand joints, Side correction, Human proportional, velocity, real time	2 distances 3 postures	84.82%

3. Proposed tracking calibration method

The proposed method in this paper aims to calibrate a coordinate by making it close to that of the front measurement after a reference point is set up using the characteristics of the human body without changing the coordinate system.

The aims of the proposed method are as follows:

1. First, conversion of the existing XYZ coordinate system used in Kinect SDK is not performed. Because an official software development kit is provided, and official SDK is one of the easiest ways for developers to access Kinect application development including media art.
2. Second, the proposed method aims to the real - time correction. Because media art is constantly replacing and accessing viewers.
3. Third, the proposed method aims to perform calibration at a distance of approximately 2m in consideration of indoor installation.
4. Fourth, viewers are constantly replaced in media art. It is not suitable for using post-processing such as machine learning. Therefore, the proposed method aims to correct the initial value without post-processing.



5.

(a) (b)

Figure 6. Skeleton tracking with front installation (a) Skeleton tracking with rotation (b)

The human body has organic movements overall rather than moving one part or organ separately. Thus, the reference point of calibration was set as inside the human body.

Figure 6a. shows an example of a skeleton that is recognized by Kinect when tracking is performed using the recommended front installation method. The right side of Figure 6a. simplifies this, and the human skeleton model that is displayed has the same depth values overall. Thus, the coordinate value can be expressed as follows:

$$\text{Right hand}\left(R_{x_1}, R_{y_1}, R_{z_1}\right)$$
$$\text{left hand}\left(L_{x_1}, L_{y_1}, L_{z_1}\right)$$
$$\text{Shoulder Depth}\left(\mathrm{LS}_{Z_1}, \mathrm{CS}_{Z_1}, \mathrm{RS}_{Z_1}\right)$$
$$R_{Z_1} = L_{Z_1} = RS_{Z_1} = LS_{Z_1} = CS_{Z_1} \quad (2)$$

A twist phenomenon of the coordinate values in (X, Y, Z) occurred more as measuring was performed at the side more due to the characteristics of Kinect, as shown in Figure 6b. because it uses the "Depth-column-based 3D coordinate system" which overlaps only the x, y 2D rectangular coordinates and depth values of the 3D spherical coordinate, not the rotation equations or the same reference point. Therefore, when the user rotates, Kinect does not recognize that the user has rotated, and recognizes that the human body has been reduced to the right and left [18] (pp. 29). This is called the coordinate twist phenomenon. The coordinate twist phenomenon occurs the most in the X axis, while it occurs relatively less in the Y axis. The coordinate value is expressed as follows:

$$\text{Right hand}\left(R_{x_2}, R_{y_2}, R_{z_2}\right)$$
$$\text{left hand}\left(R_{x_2}, R_{y_2}, R_{z_2}\right)$$
$$\text{Shoulder Depth}\left(\mathrm{RS}_{Z_2}, \mathrm{RS}_{Z_2}, \mathrm{CS}_{Z_2}\right) \quad (3)$$

$$R_{Z_2} \neq L_{Z_2}$$
$$\mathrm{LS}_{Z_2} \neq \mathrm{RS}_{Z_2} \neq \mathrm{CS}_{Z_2}$$
$$L_{Z_2} = \mathrm{LS}_{Z_2}$$
$$R_{Z_2} = \mathrm{RS}_{Z_2} \quad (4)$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 24 June 2018 doi:10.20944/preprints201806.0367.v1

3.2. Calibration method using depth-column application

If recognition is performed through real-time calibration on the side, as shown in Figure 6.a, calibration is needed through equations to recognize Figure 6b. as Figure 6a. to extend the installation range of Kinect and facilitate program use as well as promote freer planning and production. The installation range of the equipment can be expanded if calibration is performed as if the X and Z coordinates that are twisted due to the measurement from the side were measured from the front.

We approach based on facts "Some parts of the human body, such as the shoulders, are necessarily on the same plane". The following equation can be extracted from the data obtained in (2),(3), and(4).

$$L_{x_1} : \mathrm{LS}_{\mathrm{Z}_1} = L_{x_2} : \mathrm{LS}_{\mathrm{Z}_2}$$
$$\therefore L_{x_1} = \frac{L_{x_2} * \mathrm{LS}_{\mathrm{Z}_1}}{\mathrm{LS}_{\mathrm{Z}_2}}$$
$$R_{x_1} : \mathrm{RS}_{\mathrm{Z}_1} = R_{x_2} : \mathrm{RS}_{\mathrm{Z}_2}$$
$$\therefore R_{x_1} = \frac{R_{x_2} * \mathrm{RS}_{\mathrm{Z}_1}}{\mathrm{RS}_{\mathrm{Z}_2}}$$
$$\mathrm{LS}_{\mathrm{Z}_1} - \mathrm{RS}_{\mathrm{Z}_1} = \mathit{Difference\ of\ reference\ depth\ value}\ Z_\delta$$
$$L_{z_1} = L_{z_2} - Z_\delta \tag{5}$$

As described above, the opposite side was calibrated based on a depth value from one side (the left or right side) using a proportional equation to enable recognition as if they were located in the same depth value. Each coordinate modified into the same depth plane can be expressed as follows:

$$\text{Left hand position } \left(L_{x_1} = \frac{L_{x_2} * \mathrm{LS}_{\mathrm{Z}_1}}{\mathrm{LS}_{\mathrm{Z}_2}}, L_{y_1}, L_{z_2} - Z_\delta\right)$$
$$\text{Right hand position } (R_{x_1}, R_{y_1}, R_{z_1})$$
$$\text{But, } Z_\delta = \mathrm{LS}_{\mathrm{Z}_2} - \mathrm{RS}_{\mathrm{Z}_2}, \text{ Suppose } \mathrm{RS}_{\mathrm{Z}_1} = \mathrm{RS}_{\mathrm{z}} \tag{6}$$

$$\text{Left hand position } (L_{x_1}, L_{y_1}, L_{z_1})$$
$$\text{Right hand position} \left(R_{x_1} = \frac{R_{x_2} * \mathrm{RS}_{\mathrm{Z}_1}}{\mathrm{RS}_{\mathrm{Z}_2}}, R_{y_1}, R_{z_2} - Z_\delta\right)$$
$$\text{But, } Z_\delta = \mathrm{RS}_{\mathrm{Z}_2} - \mathrm{LS}_{\mathrm{Z}_2}, \text{ Suppose } \mathrm{LS}_{\mathrm{Z}_1} = \mathrm{LS}_{\mathrm{z}} \tag{7}$$
$$\text{If } \mathrm{LS}_{\mathrm{z}} < \mathrm{RS}_{\mathrm{z}} \text{ using (6)}$$
$$\text{If } \mathrm{LS}_{\mathrm{z}} > \mathrm{RS}_{\mathrm{z}} \text{ using (7)}$$

(6) and (7) are depth values at the left and right sides, respectively, in which the angles are modified by calibrating the opposite side. (6) and (7) were used when Kinect was located on the right and left sides of the subject, respectively. When Kinect was located on the right side of the subject, the error rate became larger than the target error in the original coordinate if the opposite equation shown in (6) was used as its state became an inverse correction.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 24 June 2018 doi:10.20944/preprints201806.0367.v1

4. Analysis and evaluation of experimental results

In the experiment, three types of coordinate error rates were calculated: the normal coordinates, which became the reference through front measurements; the original coordinates, which varied according to the recognition location; and the modified calibrated coordinates through the algorithm.

Kinect has been released up to v2 now, and v2 is up to 10 times different in distance error from 4m compared to v1. However, Kinect v2 is not often used in media art because it does not support a variety of applications compared to v1, and the resolution between IR camera and RGB camera is not directly proportional to v2 [24]. Therefore, in this experiment, Kinect v1 was used for the experiment due to the advantages of resolution and media art usage.

Kinect's RGB cameras support 1280x1024 or 640 x 480 resolutions and are used to measure X and Y axes. IR cameras support 640 X 480 resolutions and are used to measure the Z axis [18].

In this experiment, the resolution of the RGB camera and the IR camera was set to 640 x 480.

As shown in Figure 7, a measurement program was created to store the data every 500ms and output it to a file.

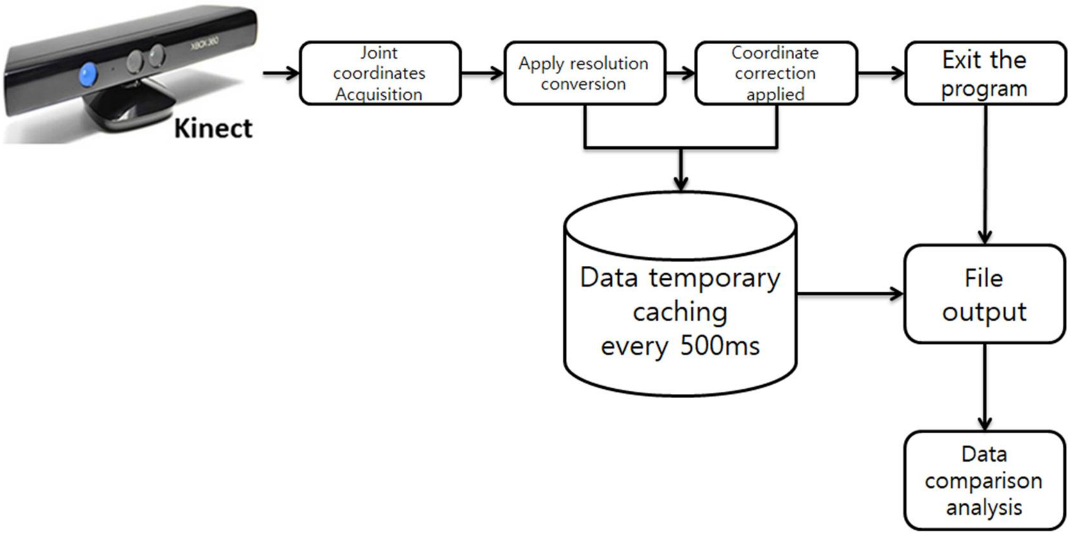


Figure 7. Block diagram of the proposed solution-applied measurement and comparison program

 doi:10.20944/preprints201806.0367.v1

Figure 8. Program interface screen of several datasets

Figure 8. shows the fabricated coordinate measurement program for measuring the original and calibrated coordinates at the same time. The five sets of coordinates represent the coordinates of the left hand, the coordinate of the left shoulder, the coordinates of the right hand, and the coordinates of the right shoulder from above, and the calibrated coordinates of the left hand.

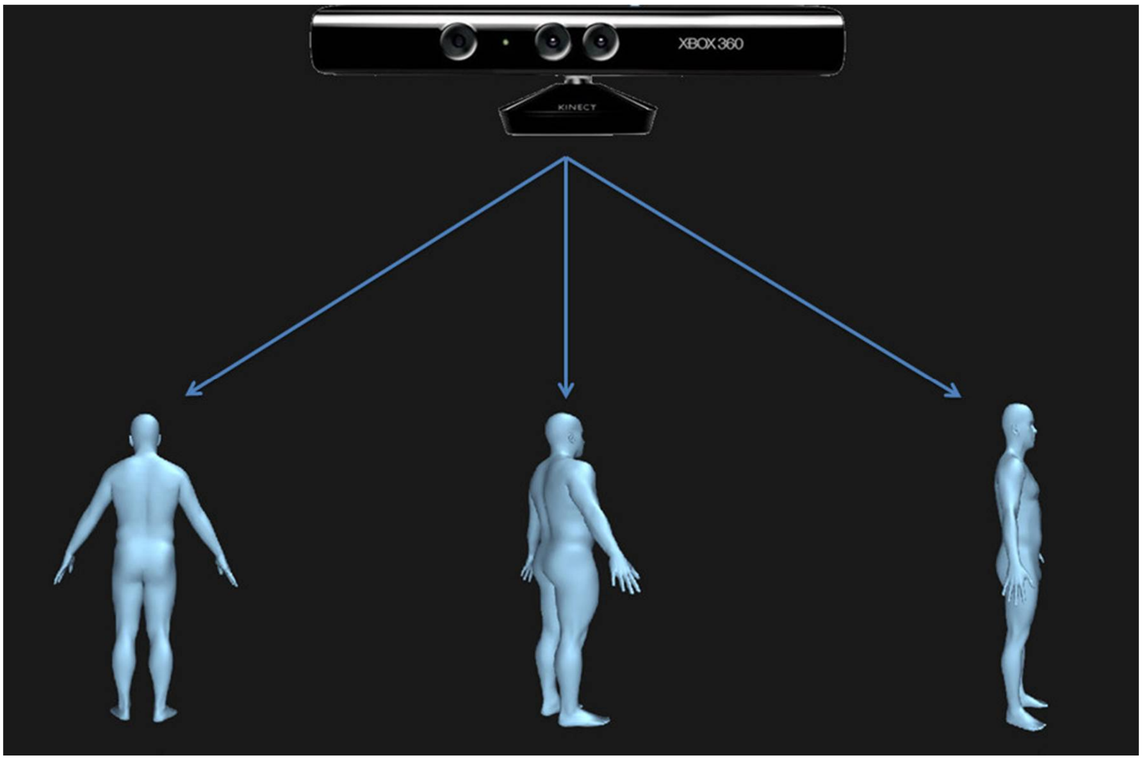
4.1 Experimental results

In the experiments, we have selected three basic postures: attention postures, hands-up postures, and hands-half-up postures for act on something [36,38,39,40,41,42]. This was measured for each pose up to 2 meters, which is most affected by distance, and 3 meters where the error sharply decreased [18,24], were measured only attention postures.

The rotations were conducted at different speeds and in different directions for each stance to evaluate the real-time measurement performance of the algorithm.

For the application of the algorithm, calibration was performed only when Kinect was located on the left side of the subject (when the subject was rotated in the right direction) to evaluate the inverse calibration state.

 doi:10.20944/preprints201806.0367.v1



(a) (b) (c)

Figure 9. Subjects at different viewing angles: (a) front view with zero degrees, (b) side view with approximately 45°, (c) side view with approximately 90°

In this experiment, Kinect was fixed to the location and the subject was also fixed to look at the front (A) and rotated until the angle between Kinect and subject became 90° (Figure 9a.-> Figure 9b.-> Figure 9c.), and then they gazed at the front again (Figure 9c.-> Figure 9b.-> Figure 9a.) to measure the accuracy rate of real-time calibration in a range of motion.

The error rate compared with the target value was measured between non-calibrated and calibrated values only in the section that entered into the calibration state.

The following is a comparison of measurement data and error rate by posture and distance.

Figures 10, 12, 14, 16, 18, and 20 are time-position graphs that measure a rotating subject once every 0.5 second.

Green line is set to the target value when the subject is in front.

Red line is raw data that is not corrected to the value when the subject rotates.

Blue line is the correction data that is corrected from 0 'to 90' as shown in Figure 9. and inverse corrected from 0 'to -90'.

Performance has been compared with raw data and the results are presented in "figure. 11, 13, 15, 17, 19and 21.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 24 June 2018 doi:10.20944/preprints201806.0367.v1

1M Attention stance turn left–turn right

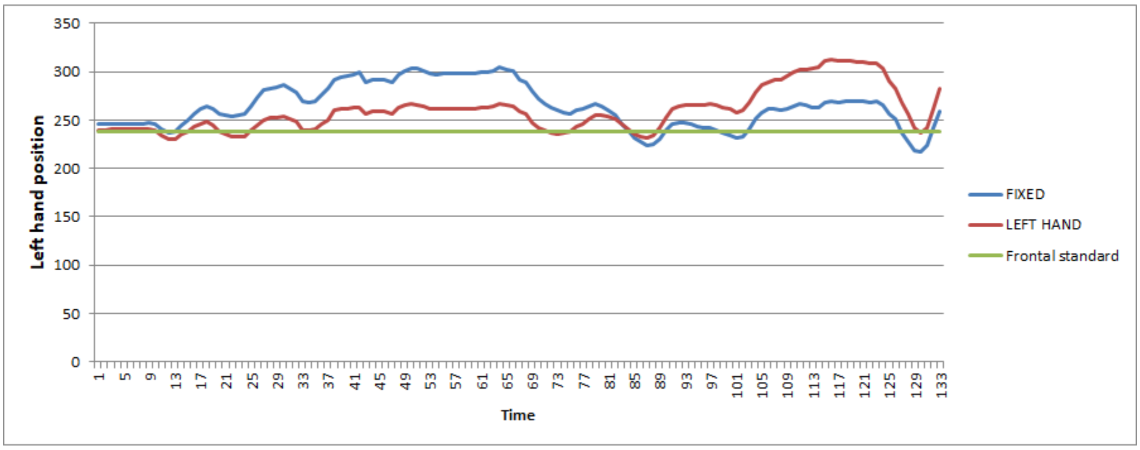


Figure 10. Coordinate change over time, 1m attention stance, turn left–turn right

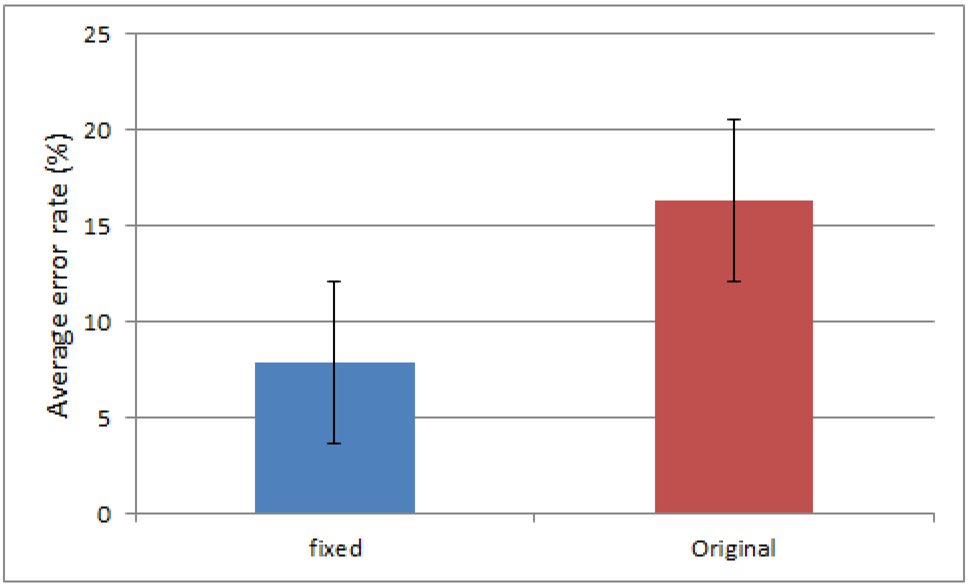


Figure 11. Average error rate, 1m attention stance, turn left–turn right

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 24 June 2018 doi:10.20944/preprints201806.0367.v1

1M Hands-up stance

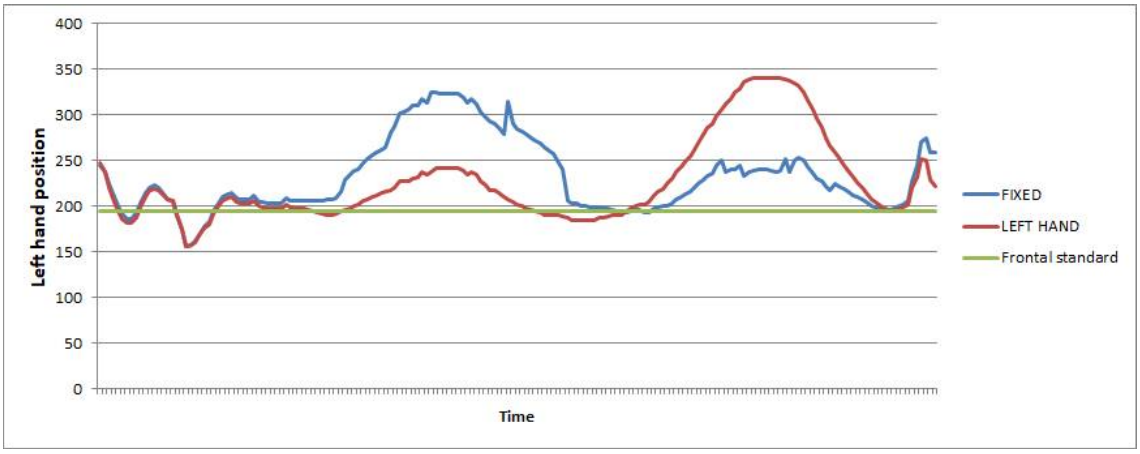


Figure 12. Coordinate change over time, 1m hands-up stance, turn right–turn left

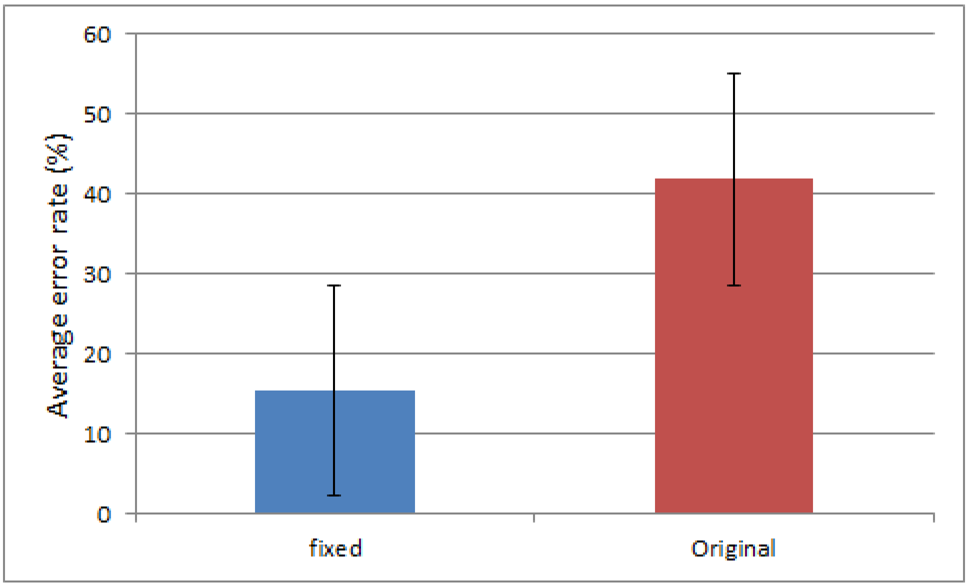


Figure 13. Average error rate, 1m hands-up stance, turn right–turn left

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 24 June 2018 doi:10.20944/preprints201806.0367.v1

2M Attention stance turn right (fast)–turn right (slow)

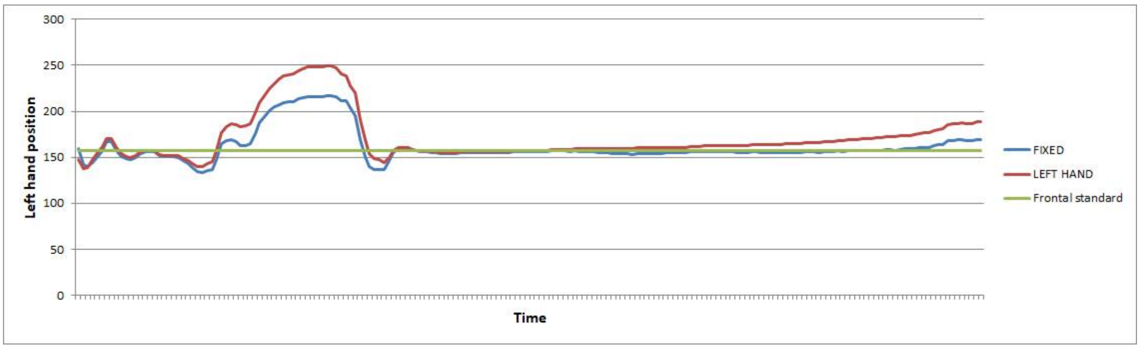


Figure 14. Coordinate change over time, 2m attention stance, turn right (fast)–turn right (slow)

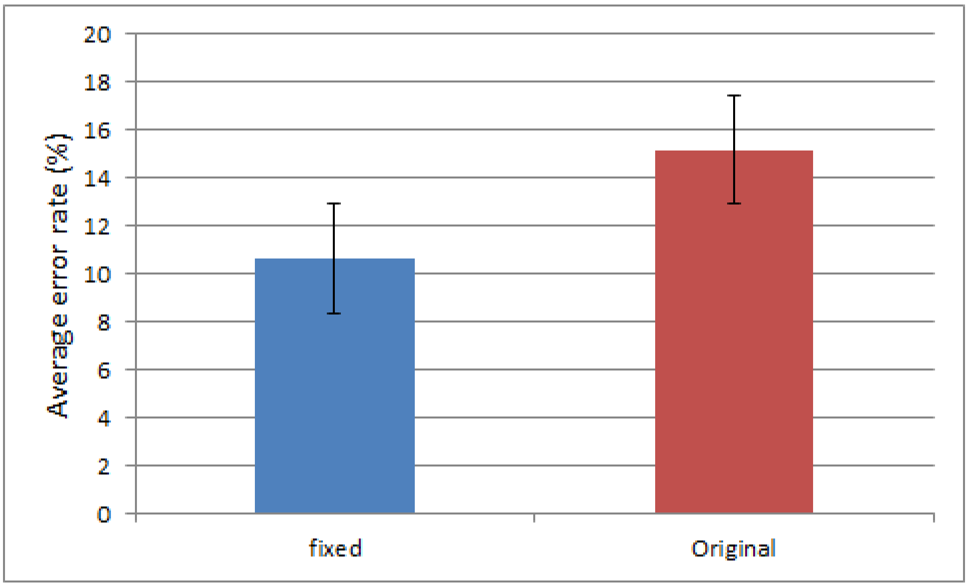


Figure 15. Average error rate, 2m attention stance, turn right (fast)–turn right (slow)

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 24 June 2018 doi:10.20944/preprints201806.0367.v1

2m Hands-up stance, turn left (fast)–turn left (slow)–turn right

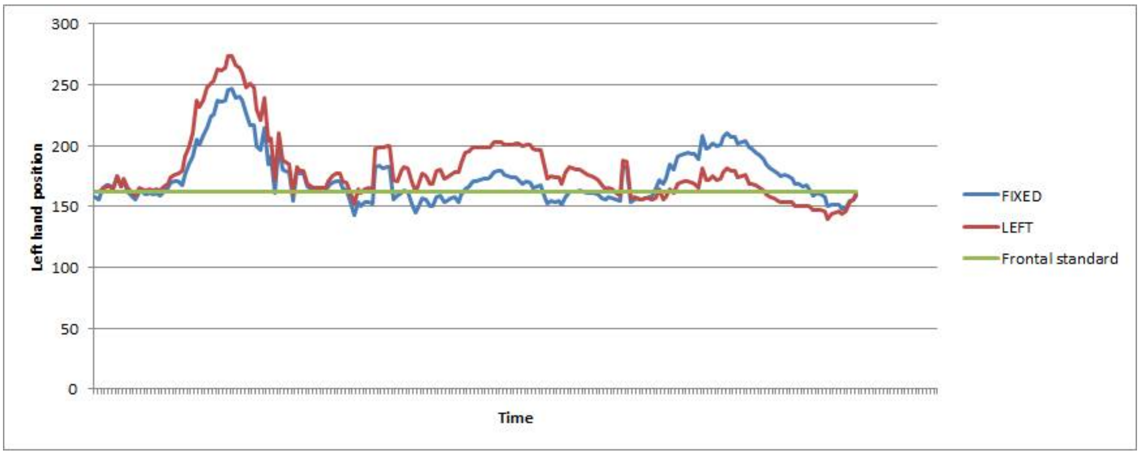


Figure 16. Coordinate change over time, 2m hands-up stance, turn left (fast)–turn left (slow)–turn right

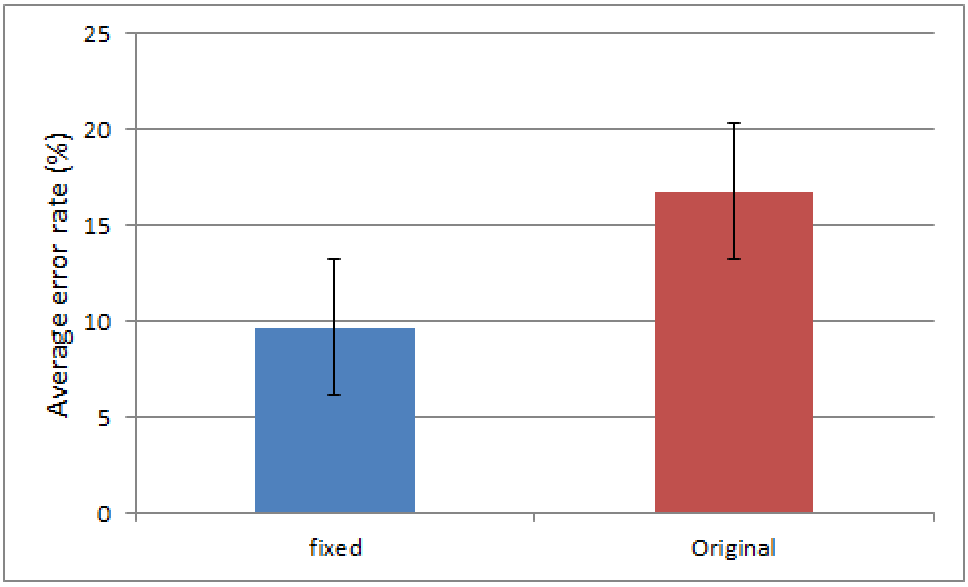


Figure 17. Average error rate, 2m hands-up stance, turn left (fast)–turn left (slow)–turn right

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 24 June 2018 doi:10.20944/preprints201806.0367.v1

2M Hands-half-up stance, turn right–turn left

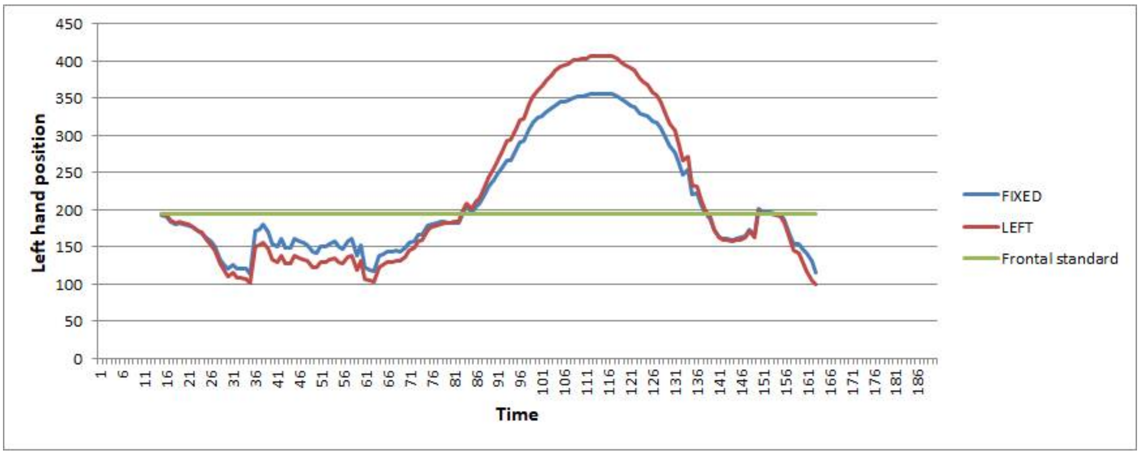


Figure 18. Coordinate change over time, 2m hands-half-up stance, turn right–turn left

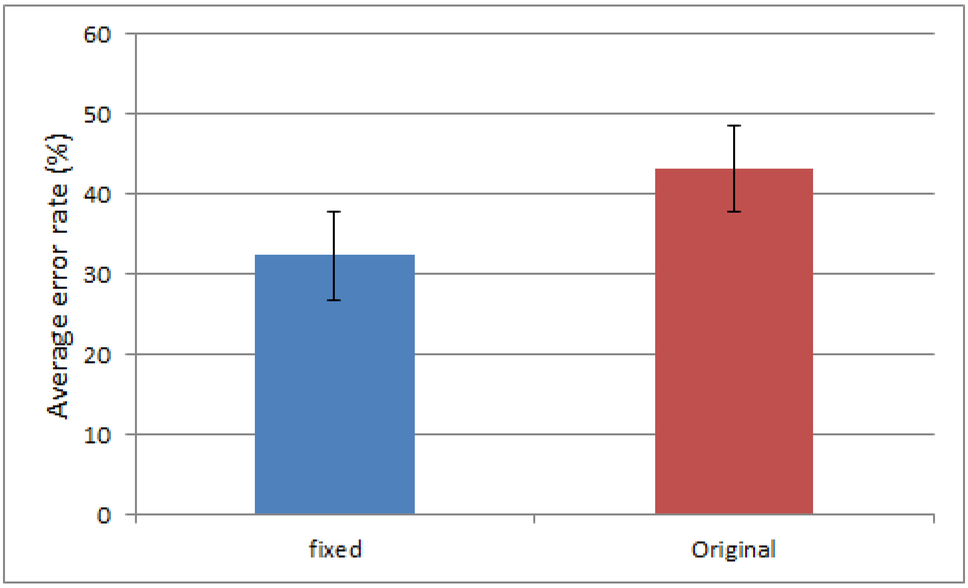


Figure 19. Average error rate, 2m hands-half-up stance, turn right–turn left

3m attention stance, turn right

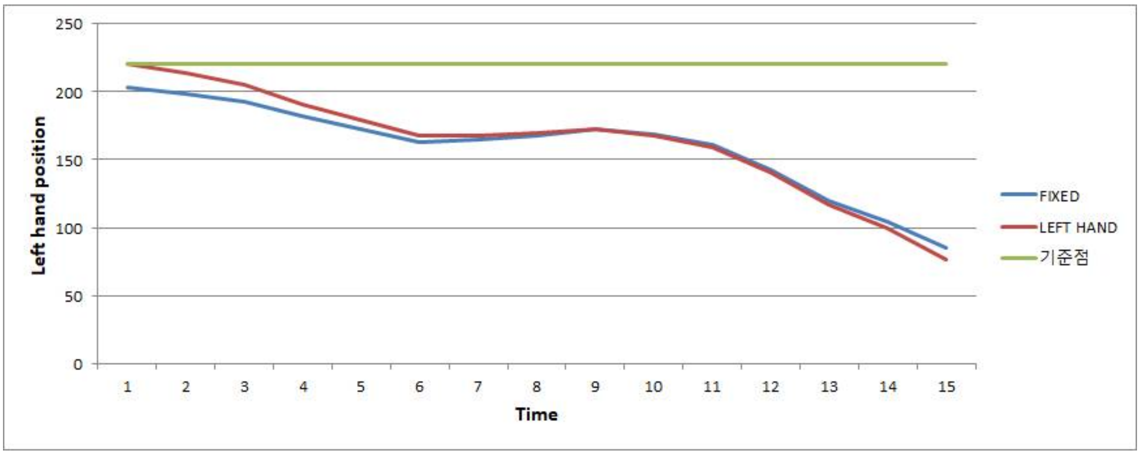


Figure 20. Coordinate change over time, 3M attention stance, turn right

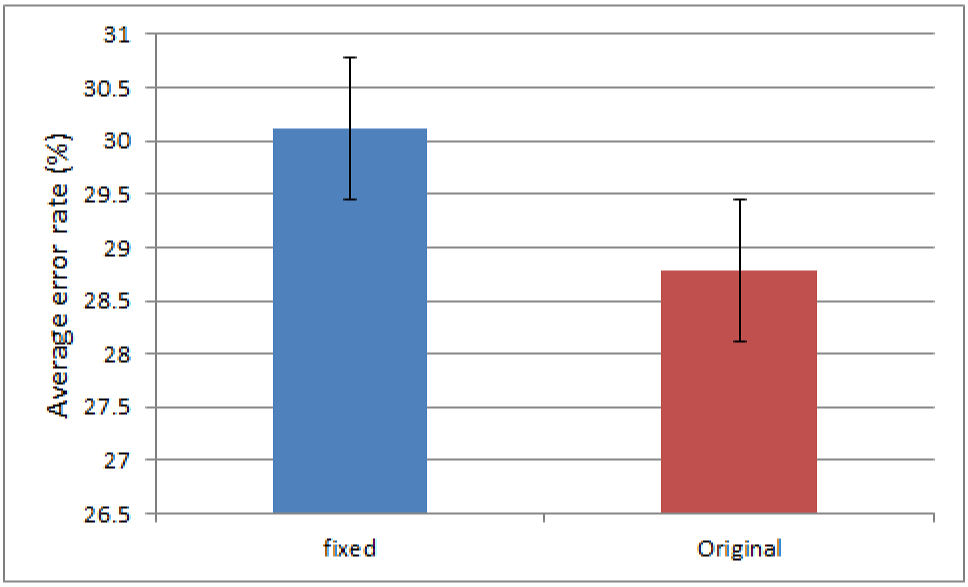


Figure 21. Average error rate, 3M attention stance, turn right

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 24 June 2018 doi:10.20944/preprints201806.0367.v1

4.2 Results

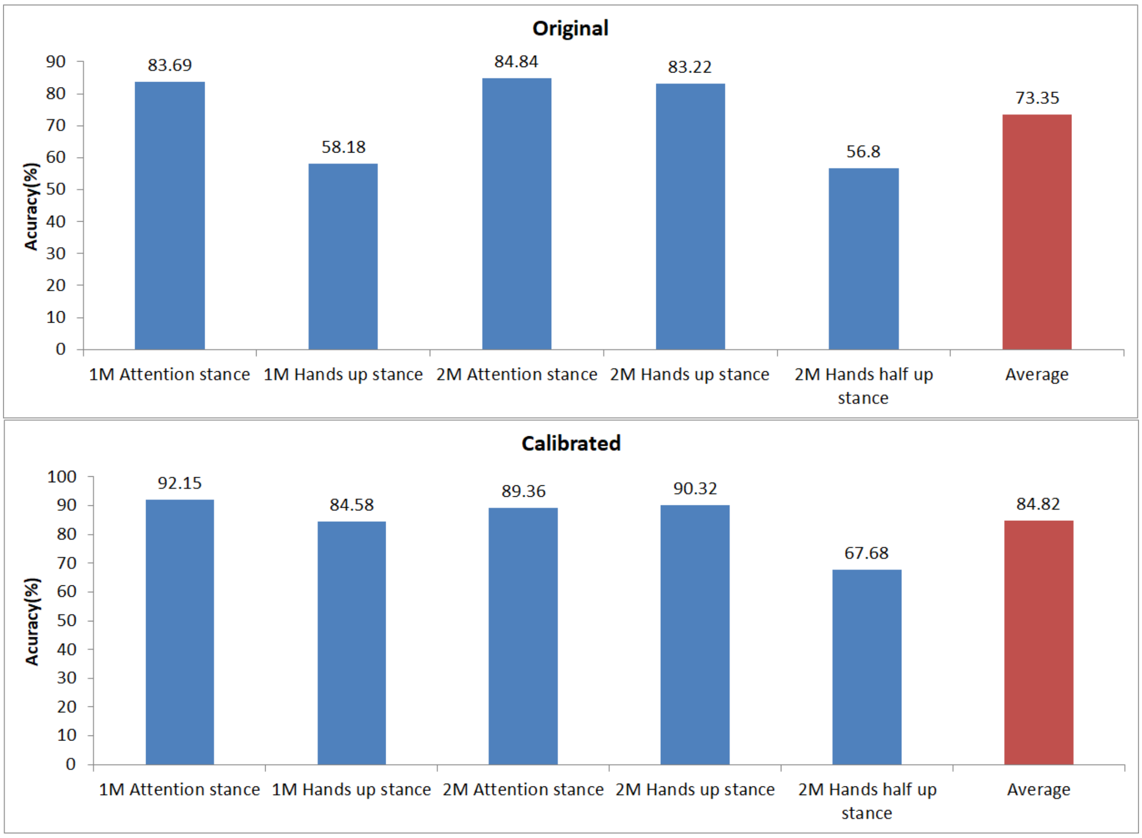


Figure 22. Original tracking error rate (above) and corrected coordinate error rate (bottom)

This section analyzes raw data and calibrated data.

The most calibrated value posture was measured as 84.58%, which was 26.4% higher than 58.18% before calibration with 1m hands up stance.

This is estimated to be due to the largest error that occurs in close proximity of Kinect as shown in figure 4.

In addition, the calibrated value was lowered as the distance was increased. As shown in Figure 22. about 10% in 1m attention stance, about 26% in 1m hands-up stance, about 4.5% in 2m attention stance, about 6.8% in 2m hands-up stance and less than 1% in 3m attention stance.

This distribution of the distance error is considered to be the main cause of the error rate of the distance according to the FOV value of Kinect and the spherical Z coordinate system as shown in figure 4.

In the case of 2m hands half up, it showed 10.8% calibrated value, but it is difficult to manipulate finely with the accuracy rate of 67.68% according to the proposed method.

However, according to related research rotation calibration using the SVM [36], the accuracy rate is 45 ~ 50% in the first attempt and it has a maximum accuracy rate of 77% after 50 ~ 60 times of machine learning. In the proposed method, the initial hit rate is high, but the final hit rate drops by about 10% because there is no post-processing. Media art, where in-out is fast and users are constantly changing, is difficult to use post-processing methods such as machine learning. Therefore, it is necessary to improve the proposed method in accordance with the characteristics of media art.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 24 June 2018 doi:10.20944/preprints201806.0367.v1

5. Conclusions

In this paper, we have proposed a real-time calibration method of Kinect recognition range expansion for media art. Our method was proposed with the goal of real - time correction without using coordinate transformation, without using machine learning. The proposed method has been tested on a large dataset of 3,400. Dataset were measured and error rates were compared by changing the distance, posture, direction, and speed to prove the utility of the proposed method. The recommended installation method for Kinect was originally 2m away from the subject, but in this experiment, 1m and 3m distances were also used for measurements in various environments. As shown in Figure. 10-22., the accuracy rate was improved by more than 12.5% on average compared with the error rate in the non-calibration status, and higher accuracy rates were revealed as the installation angle between Kinect and subject became narrower. A large error occurred even if calibration was performed in the area where the conversion of the Z-axis coordinate of the hand was conducted, such as the hand-half-up stance, which was the most widely used range of motion.

At 2m, related research rotation calibration using User Learning, Support Vector Machine and Hidden Markov Model is about 10% more effective than our proposed method with a maximum hit rate of 77% [36]. However, related research rotation calibration using User Learning, Support Vector Machine and Hidden Markov Model has about 45% initial accuracy rate and that is about 15% lower than the initial accuracy rate of our proposed method. Therefore, we can expect to increase the accuracy rate in parallel with future learning methods.

The accuracy was increased by 5-10% in most cases, and accuracy was improved further at 1m than at 2m and when the hand was raised than when the hand was lowered. The largest error calibration rate was exhibited when the hand was raised within a distance of 1m, in which the accuracy was improved by 26%.

In future, we will combine various gesture interfaces, such as sign language recognition, to analyze various motion recognition rates. In related research has obtained better results with machine learning such as Support Vector Machine or Hidden Markov Model [36,38,39,40,41,42]. In this paper, we tried to get a better initial value by suggesting that machine learning is not suitable in media art because media art is constantly changing audiences. As a result, the results were at least 5% up to about 27% except the 2M Half Hands Up posture, but about 11% lower at the 2M Half Hands Up posture.

As future tasks, we can get better results by combining machine learning methods like Support Vector Machine or Hidden Markov Model or User Learning as shown in related research. But machine learning is mainly used for long-term and limited subject datasets, it is necessary to study and modify short-term and various object datasets for media art. Additionally, more complicated gesture recognition or data gathering and analysis about sign language mainly used in media art should be performed. We will design a gesture interface analysis system in combination with skeletal motion data.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 24 June 2018 doi:10.20944/preprints201806.0367.v1

References

1. TH Apperley. The body of the gamer: Game art and gestural access. *Digital Creativity. 24* ,2013, pp.145-156.
2. Wikipedia-Natural User Interfaces Available online: https://en.wikipedia.org/wiki/Natural_user_interface (accessed on 26 April 2018).
3. J. Shotton. Real-Time Human Pose Recognition in Parts from a Single Depth Image, *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2011, 1297-1304.
4. Hyun-chul Jung.; Nam-jin Kim.; Lee-kwon Choi. The Individual Discrimination Location Tracking *Technology for Multimodal Interaction at the Exhibition, Journal of Intelligence and Information Systems* **2012**, 18,2, pp.19-28.
5. Rodrigues, D.G.; Grenader, E.; Da Silva Nos, F.; De Sena Dall'Agnol, M. Hansen, T.E.; Weibel, N.; 6 MotionDraw: a Tool for Enhancing Art and Performance Using Kinect. *Conference on Human Factors in Computing Systems*. 2013, 1197-1202
6. Chen, K.-M.; Wong, S.-K. Interactive sand art drawing using kinect. *ACM International Conference Proceeding Series*. 2014, 78-87.
7. Woohyeon Kim. Touch and Gesture based interactions with Auto-stereoscopic Display based on Depth Cue, Master's thesis, Konkuk University, Seoul, 2011.
8. Lim, Jung-Geun.; Han, Kyongho. Remote Image Control by Hand Motion Detection. *Journal of IKEEE* **2012,** 16,4, pp.369-374.
9. Sang-Beom Lee.; Yo-Sung Ho. Real-time Eye Contact System Using a Kinect Depth Camera for Realistic Telepresence. *The Journal of Korean Institute of Communications and Information Sciences* **2012**, 37, 4, pp.277-282.
10. El-Iaithy, R.A.; Jidong, H.; Yeh, M. Study on the Use of Microsoft Kinect for Robotics Applications. In Proceedings of the Position Location and Navigation Symposium (PLANS), Myrtle Beach, SC, USA, 23–26 April 2012; 1280–1288.
11. Macknojia, R.; Chavez-Aragon, A.; Payeur, P.; Laganiere, R. Calibration of a Network of Kinect Sensors for Robotic Inspection over a Large Workspace. In Proceedings of the 2013 IEEE Workshop on Robot Vision (WORV), Clearwater, FL, USA, 15–17 January 2013; pp. 184–190.
12. Richard A. Newcombe , Shahram Izadi , Otmar Hilliges , David Molyneaux , David Kim , Andrew J. Davison , Pushmeet Kohli , Jamie Shotton , Steve Hodges , Andrew Fitzgibbon, KinectFusion: Real-time dense surface mapping and tracking, *Proceedings of the 2011 10th IEEE International Symposium on Mixed and Augmented Reality* **2011**, p.127-136.
13. J. Smisek, M. Jancosek, and T. Pajdla, "3D with Kinect," in ICCV Workshop on Consumer Depth Cameras for Computer Vision, 2011.
14. Min-Ho Song, Kwan-Hee Yoo. GPU Based Filling Methods for Expansion of Depth Image Captured by Kinect. *Journal of The Korean Society for Computer Game* **2016**, 29, pp.47-54.
15. Bernacchia, N.; Scalise, L.; Casacanditella, L.; Ercoli, I.; Marchionni, P.; Tomasini, E.P. Non Contact Measurement of Heart and Respiration Rates Based on Kinect™. In Proceedings of the 2014 IEEE International Symposium on Medical Measurements and Applications (MeMeA), Lisbon, Poutugal, 11–12 June 2014; pp. 1–5.
16. Kastaniotis, D.; Economou, G.; Fotopoulos, S.; Kartsakalis, G.; Papathanasopoulos, P. Using Kinect for Assessing the State of Multiple Sclerosis Patients. In Proceedings of the 2014 EAI 4th International Conference on Wireless Mobile Communication and Healthcare (Mobihealth), Athens, Greece, 3–5 November 2014; pp. 164–167.
17. Han, J.; Shoo, L.; Xu, D.; Shotton, J. Enhanced Computer Vision with Microsoft Kinect Sensor: *A Review. IEEE Trans* **2013**, 43, 1318–1334.
18. Masalkar et al. "In-home depth camera calibration", United States Patent No. US 8,866,889 B2, Microsoft Corporation, Redmond, WA (US), 2014.
19. Zhu, M.; Huang, Z.; Ma, C.; Li, Y. An Objective Balance Error Scoring System for Sideline Concussion Evaluation Using Duplex Kinect Sensors. *Sensors* **2017**, 17, 2398.

 doi:10.20944/preprints201806.0367.v1

20. Pezzuolo, A.; Guarino, M.; Sartori, L.; Marinello, F. A Feasibility Study on the Use of a Structured Light Depth-Camera for Three-Dimensional Body Measurements of Dairy Cows in Free-Stall Barns. *Sensors* **2018**, 18, 673.
21. Chen, C.; Yang, B.; Song, S.; Tian, M.; Li, J.; Dai, W.; Fang, L. Calibrate Multiple Consumer RGB-D Cameras for Low-Cost and Efficient 3D Indoor Mapping. *Remote Sens.* **2018**, 10, 328.
22. Liao, Y.; Sun, Y.; Li, G.; Kong, J.; Jiang, G.; Jiang, D.; Cai, H.; Ju, Z.; Yu, H.; Liu, H. Simultaneous Calibration: A Joint Optimization Approach for Multiple Kinect and External Cameras. *Sensors* **2017**, 17, 1491.
23. Pagliari, D; Menna, F; Roncella, R; Remondino, F; Pinto, L. Kinect Fusion improvement using depth camera calibration, The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences; Gottingen Vol. XL, Iss. 5, 2014; 479-485.
24. Pagliari, D.; Pinto, L.Calibration of Kinect for Xbox One and Comparison between the Two Generations of Microsoft Sensors. *Sensors* **2015**, 15, 27569-27589.
25. Auvinet, E.; Multon, F.; Meunier, J. New Lower-Limb Gait Asymmetry Indices Based on a Depth Camera. *Sensors* **2015**, 15, 4605-4623.
26. Liao, Y.; Sun, Y.; Li, G.; Kong, J.; Jiang, G.; Jiang, D.; Cai, H.; Ju, Z.; Yu, H.; Liu, H. Simultaneous Calibration: A Joint Optimization Approach for Multiple Kinect and External Cameras. *Sensors* **2017**, 17, 1491.
27. Pezzuolo, A.; Guarino, M.; Sartori, L.; Marinello, F. A Feasibility Study on the Use of a Structured Light Depth-Camera for Three-Dimensional Body Measurements of Dairy Cows in Free-Stall Barns. *Sensors* **2018**, 18, 673.
28. Zhu, M.; Huang, Z.; Ma, C.; Li, Y.An Objective Balance Error Scoring System for Sideline Concussion Evaluation Using Duplex Kinect Sensors. *Sensors* **2017**, 17, 2398.
29. Chen, C.; Yang, B.; Song, S.; Tian, M.; Li, J.; Dai, W.; Fang, L. Calibrate Multiple Consumer RGB-D Cameras for Low-Cost and Efficient 3D Indoor Mapping. *Remote Sens.* **2018**, 10, 328.
30. Kim, J.-H.; Park, M.Visualization of Concrete Slump Flow Using Kinect Sensor. *Sensors* **2018**, 18, 771.
31. Liao, Y.; Sun, Y.; Li, G.; Kong, J.; Jiang, G.; Jiang, D.; Cai, H.; Ju, Z.; Yu, H.; Liu, H. Simultaneous Calibration: A Joint Optimization Approach for Multiple Kinect and External Cameras. *Sensors* **2017**, 17, 1491.
32. Tang, S.; Zhu, Q.; Chen, W.; Darwish, W.; Wu, B.; Hu, H.; Chen, M. Enhanced RGB-D Mapping Method for Detailed 3D Indoor and Outdoor Modeling. *Sensors* **2016**, 16, 1589.
33. Antensteiner, D.; Štolc, S.; Pock, T. A Review of Depth and Normal Fusion Algorithms. *Sensors* **2018**, 18, 431.
34. Di, K.; Zhao, Q.; Wan, W.; Wang, Y.; Gao, Y. RGB-D SLAM Based on Extended Bundle Adjustment with 2D and 3D Information. *Sensors* **2016**, 16, 1285.
35. Kim, J.; Hasegawa, T.; Sakamoto, Y. Hazardous Object Detection by Using Kinect Sensor in a Handle-Type Electric Wheelchair. *Sensors* **2017**, 17, 2936.
36. P. Kumar, R. Saini, P. P. Roy, D. P. Dogra, "A position and rotation invariant framework for sign language recognition (SLR) using KINECT" in *Multimedia Tools and Applications Springer* **2017**, pp. 1-24.
37. S Izadi, D Kim, O Hilliges, D Molyneaux, "KinectFusion: real-time 3D reconstruction and interaction using a moving depth camera", *Proceedings of the 24th annual ACM symposium on User interface software and technology* **2011**, pp. 559-568.
38. Patsadu O, Nukoolkit C, Watanapa B Human gesture recognition using kinect camera. International Joint Conference on Computer Science and Software Engineering, Bangkok, Thailand, 2012; pp. 28–32.
39. Monir S, Rubya S, Ferdous HS Rotation and scale invariant posture recognition using microsoft kinect skeletal tracking feature. International Conference on Intelligent Systems Design and Applications, Kochi, India, 2012; pp 404–409.
40. Almeida SGM.; Guimar˜aes FG.; Ram´ırez JA. Feature extraction in Brazilian sign language recognition based on phonological structure and using rgb-d sensors. *Expert Systems with Applications* **2014,** 41, pp. 7259–7271.
41. Lim KM, Tan AWC, Tan SC. A feature covariance matrix with serial particle filter for isolated sign language recognition. *Expert Systems with Applications* **2016**, 54, pp. 208–218.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 24 June 2018 doi:10.20944/preprints201806.0367.v1

42. Uebersax D, Gall J, Van den Bergh M, Gool LV. Real-time sign language letter and word recognition from depth data. International Conference on Computer Vision Workshops, Barcelona, Spain, 2011; pp 383–390
43. R. Lulu; Vanshri Kurhade; Neha Pande; Rohini Gaidhane; Nikita Khangar; Priya Chawhan; Ashwini Khaparde. International Journal of Advanced Research in Computer Engineering & Technology **2015**, 4,2, pp372-375.