International Journal of Research in Computer Science
eISSN 2249-8265 Volume 4 Issue 6 (2015) pp. 1-7
www.ijorcs.org, A Unit of White Globe Publications

HEAD AND HAND DETECTION USING KINECT CAMERA 360

[1]Mostafa Karbasi, [2]Ahmad Waqas, [3]Parham Nooralishahi, [4]Seyed Mohammad Reza Mazloomnezhad
[1,2]Faculty of Information Computer Technology, IIUM University, MALAYSIA
[1]E-mail: mostafa.karbasi@live.iium.edu.my
[2]Email: ahmad.waqas@live.iium.edu.my
[4]Email: specialist.m@live.com

[3]Faculty of Computer Science and Information Technology, UM University, MALAYSIA
[3]E-mail: parham.nooralishahi@gmail.com

Abstract: *Using head and hand blobs as an input to the computer are very crucial for human-computer interaction (HCI) applications. These blobs play an important role in bridging the information gap between a human and computer. One of the famous technologies that play a crucial role as an advanced input device for HCI is the Kinect camera developed by Microsoft. Kinect camera (codenamed Project Nathal) has a distinct advantage over other 3D cameras because it obtains more accurate depth information of a subject easily and very fast. By using Kinect, one can track up to six people concurrently and also obtain motion analysis with feature extraction. Being extremely useful in indoor HCI applications, it cannot be used in outdoor applications because its infrared depth sensor makes it extremely sensitive to sunlight.*

Keywords: *kinect, head detection, hand detection, indoor*

I. INTRODUCTION

Lee and Oh [1] mentions the first Kinect digital camera launched in June 1, 2009 under the particular brand of “Project Natal”. Traditionally, Microsoft utilizes cities as a code name for their project. “Project Natal” got its name from one of the Brazilian city of Natal by Microsoft director Alex Kipman who introduced and launched this project. According to Kean, Hall [2] and Hsu [3] The actual meaning of natal is “of or relating to birth”. It indicates the view of project which means “next generation of home entertainment”. A kinect component is shown in Figure 1, integrated with red-green-blue camera, an infrared sensor, a four-array microphone, and a three-axis sensor.

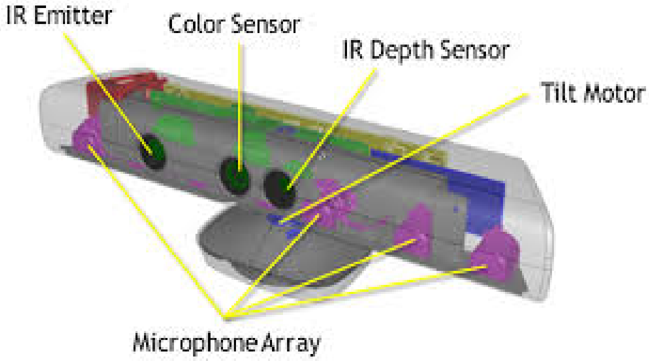

Figure 1: Kinect sensor component

The design of Kinect is based on Infrared technology which is operated by Microsoft, as well as a variety of digital camera technologies which are owned by Israel developer Prime Sense. The depth sensor in this digital camera is a infrared laser projector which is merged with a monochrome CMOS sensor. The depth sensor can easily record and capture video clip data in 3D within any kind of light condition. The full designed system and program (RGB camera, depth sensor and multi-array microphone) can recognize and interpret specific gestures, producing absolutely hands-free control of electrical gadgets which is feasible by utilizing an infrared projector, a digital camera and a specific micro-chip to monitor the motion of physical objects and individuals which have three dimensions. Also, Kinect software development kit was released in June 16, 2011. This SDK provides permission to create apps in C++/CLI, C# or Visual Basic.NET. Kinect version for Windows was launched on February 1, 2012.

Wang and Sung [4] The Kinect digital camera can handle full-body 3D movement record, face recognition and tone of voice recognition. Kinect camera has numerous benefits when compared to other video cameras. By working with Kinect, one can easily monitor up to six individuals simultaneously.

The movement analysis with function eradication of 20 joint per player can be received. Kinect camera field-of-view range is limited between 1.2-3.5m (3.9-11.5 ft.)[5, 6]. The horizontal area of the Kinect sensor is at a minimal observing distance of ~87 cm (34 in), and the vertical area is ~63 cm (25 in), producing in a resolution of just over 1.3 mm (0.051 in) per pixel. There are four microphones in Kinect with each channel of 16-bit audio at a rate of 16 kHz [7] [8] . The depth sensor from Kinect gets an 11-bit number raw data (d_{ij}) that requires additional procedure to get the accurate depth ($z_{i,j}$). The true depth can be identified by equation 1.

$$z(d) = a_1 \times \exp(-\left(\frac{d-b_1}{c_1})^2\right) + a_2 \times \exp(-\left(\frac{d-d_2}{c_1})^2\right) \ldots\ldots\ldots\ldots (1)$$

$a_1 = 3.169 \times 10^4$
$b_1 = 1338.0$
$c_1 = 140.4$
$a_2 = 6.334 \times 10^{18}$
$b_2 = 2.035 \times 10^4$
$c_1 = 3154.0$

The specifications of Kinect sensor is show below in Table 1.

Table 1: Specification of Kinect Sensor (Lee)

Type	***Description***
Resolution of Color image	640x480 pix
Resolution of Depth image	320x240 pix
Resolution (z axis)	10mm @ 2000mm
Frame rate	30 fps
Horizontal angle	43 degree
Vertical angle	57 degree

The skeleton model used consists of 3D-coordinates of 20 joints shown in Table 2.

Table2: Joints of Skeleton Model (Lee)

Head	Shoulder Center	Spine
Hip Right	Knee Right	Ankle Right
Hip Left	Knee Left	Ankle Left
Elbow Right	Shoulder Right	Wrist Right
Elbow Left	Shoulder Left	Wrist Left
Hip Center	Foot Right	Foot Left
Hand Right	Hand, Left	

II. RELATED RESEARCH

Kinect is Microsoft’s motion game playing system for the Xbox 360. The system delivers a natural individual user interface (NUI) that enables individuals to interact automatically and without the need of any intermediary gadget, such as a controller. The Kinect system recognizes individual participants via face recognition and voice recognition. A depth camera, which “sees” in 3-D, generates a skeleton graphic and picture of a player and a motion sensor registers their motions. Speech recognition software enables the system to comprehend and recognize verbal instructions and also gesture recognition provides the tracking of player motions. Even though, Kinect was developed for playing games actively, the technological innovation has been used to real-world apps as diverse as electronic signage, virtual shopping, education, health services delivery and other areas of health IT.

In March 2011, medical professionals at Sunnybrook Hospital in Toronto commenced using Kinect to manipulate medical graphics via gestures and signals throughout some operations. Performing this method enables doctors to have interaction with the images without leaving the sterile operating place that indicates they do not have to clean up frequently. Stay of the doctors in sterile area decreases the risk of contamination and can easily avoid delays of up to an hour over the course of an operation. Cumulatively, that time could allow more operations to be completed. The medical centers also have planned to utilize Kinect for other applications, such as physiotherapy. Kinect was released on November 4, 2010, and over 80 million devices were distributed by January 3, 2011 which grabbed the Guinness World Record for the fastest-selling electronic device to customers[9] .

Many systems have attempted to use Kinect sensor as an input gadget for action, movement and speech recognition. Miranda, Vieira [10] presented a method for real-time motion recognition from a noisy skeleton stream, such as the ones made from Kinect depth sensors. Every single pose is identified using a tailored angular reflection of the skeleton joints.

Other researcher Rimkus, Bukis [11], utilized a Kinect system as a 3D data scanning device. Consequently, the 3D coordinates are computed directly from depth images. They have also the neural network construction to enhance outcomes for recognition and monitoring. Some methods utilize the location of functions such as the eyes, mouth, and nose tip to identify the head position from their comparative arrangement [12]. Tuzel, Porikli [13] described new location descriptor for item recognition and structure distinction. They identified a quick technique for computation of covariance based on integral graphics. Erol, Bebis [14] described hand position evaluation systems and seeks to catch the real 3D motion of the hands. Their findings present a literature review on the last mentioned research direction, which is a very challenging issue in the HCI perspective. Athitsos and Sclaroff [15] suggested a method that can create a rated list of possible three-dimensional hand options that best match an input image. Hand pose estimation is designed as an image data source indexing issue, where the best matches for an input hand image are gathered from a significant database of synthetic hand graphics. Rosales, Athitsos [16] shown that 3D hand pose from monocular color series is recommended. The program operates as a non-linear monitored mastering framework, the specific mappings architecture (SMA), to map image features to likely 3D hand poses. A new model-based method used for 3D hand position, the hand texture, and the illuminant are dynamically estimated via minimization of an objective function. The objective function helps precise utilization of temporary texture continuity and shading information and facts while managing essential self-occlusions and time-varying illumination[17]. de La Gorce, Fleet [17] implemented the concept of an advanced reflection for the item whose position is to be approximated. They generated synthetic hand graphics and labeled their parts, such that each skeleton joint is at the center of one of the labeled areas. They also created large database developed from randomly and manually set skeleton parameters, and practiced various randomized decision trees (RDT) which are then applied to categorize each pixel of the retrieved depth image. Park, Hasan [18] used depth details to recognize scale and rotation invariant positions dynamically.

III. DEPTH CUES

By having the depth of pixel, the face can be detected and its size can be obtained as well. We can save a lot of computation time by restricting the scales at which we search for face and hand. To develop the effectiveness of recognition detection, one can apply distance thresholding method. Locations of the image related to points further more than a provided distance are likely to be unclear, or to contain too few pixels for efficient recognition. Due to the fact we cannot detect interested areas, even though if they are existing, there is no position in seeking, and we can prevent operating blobs sensor over these image locations. Remarkably, denser stereo images, such as those created by structured light systems may not be better for boosting up face and hand recognition than sparser ones[19]. Traditional stereo cameras deliver sparse data. They rely on passively discovering texture in the graphics and, therefore, are unable to provide depth details in areas without having texture. Traditional stereo camera systems can easily recognize these areas (by not detecting texture), although more advanced, active systems can not (since they project their own texture on the world). We can say that, depth images created from passive techniques are likely to enable us to more strongly prune the search area of the face-detection algorithm.

IV. OVERALL WORKFLOW

The overall work flow process is composed of four steps as shown in Figure 2.

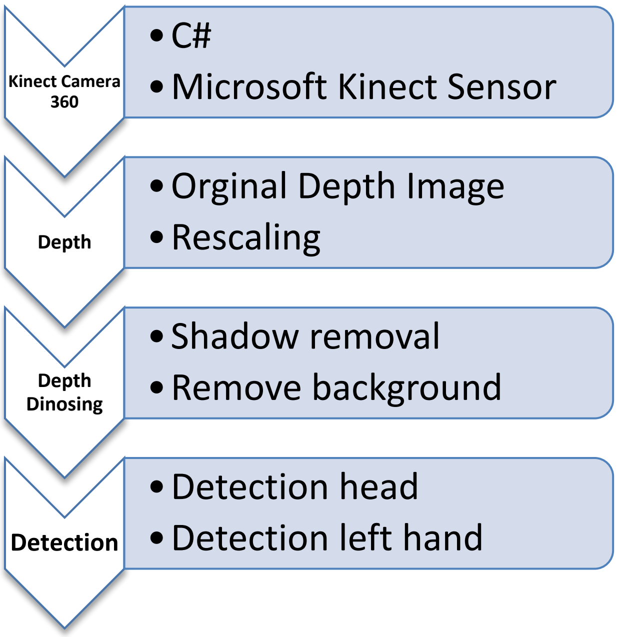

Figure 2: Overall Kinect workflow for head and left hand detection

A. The first step is to setup programing environment

- Visual C#
- Kinect camera
- Kinect Sensor.

B. Depth image

1. Orginal Depth image

In the beginning, depth images with a resolution of 640*480 at 30fps can be obtained using Kinect Sensor library. Each frame is converted into a matrix of uint8-gray scale.

2. Rescaling

To improve the depth image, it must be rescaled by adjustment of the scale factor. This can be achieved by making the sensitive range with a minimum depth to about 1 meter from Kinect. When the user operates Kinect camera correctly, this maximum depth should reach some point at the user's body.

C. Depth Denoising

Working with depth data means that broad shapes are consistent, smaller shapes can be inaccurate because of the noise present in Kinect data. This noise isn't purely random, but rather due to the combination of limited resolution, offset between the IR camera/sensor, and of course the nature of the surface of whatever you're looking at.

1. Shadow removal

We use an iterative process that propagate values from known depth values to unknown depth values in the same segmented cluster.

2. Remove background

Depth images, as they are provided by Kinect, suffer from quite unusual noise. Removing the background is the best way was to calculate distance. In this method, by determining the minimum and maximum distance, the background can be removed.

D. Detection

The face and hand does not contain distinct characteristics, and a depth image would give only the shape information. The use of shape information requires a segmentation of an object, which is a very complex process.

V. THE EFFECT OF DISTANCE ON KINECT DEPTH

To find out the relationship between Kinect depth and object distance, the following experiment carried out. The object's distance from the sensor (1ft to 10ft) was measured. The findings of this experiment show that the relationship between Kinect depth and object's distance is not linear and it followed a logarithmic scaling. The following figure 3 and table 3 shows the details.

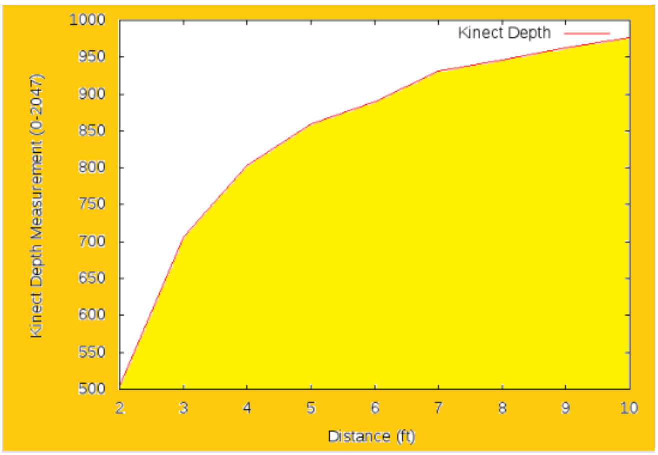

Figure3: Kinect Depth Measurement and Actual Distance

Table 3: Actual Distance and Kinect Data

Distance (Feet)	Capacity (Pixel)
2	504
3	706
4	803
5	859
6	890
7	931
8	946
9	963
10	976

The results reveal that when the object goes further, the image which is taken by Kinect is not clear.

VI. HEAD POSE DETECTION

The function of the form *Z*(*X*; *Y*; *t*) can be applied to view depth map from Kinect. By applying derivative of *Z*, we can use equation (2).

$$\frac{dZ}{dt}=\frac{\partial Z}{\partial t}\frac{dX}{dt}+\frac{\partial Z}{\partial Y}\frac{dY}{dt}+\frac{\partial Z}{\partial t}\dots\dots\dots\dots(2)$$

The depth rate constraint as shown in equation (3).

$$\dot{Z}=p\dot{X}+q\dot{Y}+Z_t\dots\dots\dots\dots(3)$$

Three partial derivatives of Z by equation (4).

$$p=\frac{\partial Z}{\partial X'}\ q=\frac{\partial Z}{\partial Y'}\ and\ Z_t=\frac{\partial Z}{\partial t}\dots\dots\dots\dots(4)$$

Calculate the components of the velocity of a point in the depth image by equation (5).

$$\dot{X} = \frac{dX}{dt}, \dot{Y} = \frac{dY}{dt} \; and \; \dot{z} = \frac{dZ}{dt} \dots\dots\dots\dots (5)$$

To increase the amount of constraint by using a direct method for obtaining motion from ordinary image sequence [19, 20]. This can be implemented because sensors are rigid and hence the motion of the head depends on the sensors. Then, for recovering the motion, six degree of freedom can be used. Vector *R*=(*X*, *Y*, *Z*)^T shows a point on a head. When the head move with instantaneous translational velocity t and instantaneous rotational velocity t with respect to the sensor, then we can notice that the point R appears to move with a velocity in equation (6).

$$\frac{dR}{dt} = -t - w \times R \dots\dots\dots\dots (6)$$

With respect to the sensor[21]. By following formula can obtain component of velocity vectors by equation (7).

$$t = \begin{pmatrix} u \\ v \\ w \end{pmatrix} \; and \; w = \begin{pmatrix} A \\ B \\ C \end{pmatrix} \dots\dots\dots\dots (7)$$

Expanded equation can be substituted into depth rate constraint equation (8) itself yields

$$pU + qV - W + rA + sB + tC = Zt \dots\dots\dots\dots (8)$$

Where

$$r = -Y - qZ, s = X + pZ, and \; t$$
$$= qX - pY \dots\dots\dots\dots (9)$$

Six degrees of freedom can be formulated by solving final matrix equation and computing matrix *X*. Figure 4 shows the boundary of the rectangle containing the head.

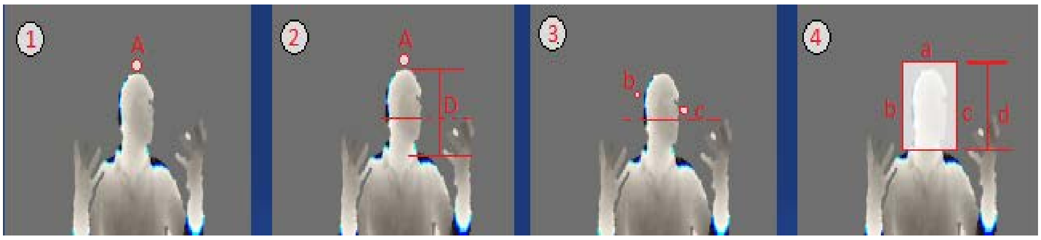

Figure4: Head Localization

VII. HAND POSE DETECTION

In order to distinct the hands from the background to track and record them, depth thresholds are usually fixed manually in progress to identify the exact depth range where motions need to appear to be recognized. The detection range, is between 0.5m and 0.8m; up to two hands can be detected within this level and particular range. Pixels with absolute depths out of the range are ignored in the rest of the motion and gesture detection procedure. Therefore, a hand pixel might also be recognized to as a point. These hand pixels are estimated to a 2D space for subsequent analysis and evaluation. Distance between two pixels p$_1$ (x$_l$, y$_1$) and p$_2$(x$_2$, y$_2$) is identified by equation (10).

$$D(p_1, p_2) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \dots\dots\dots\dots (10)$$

The K-means clustering formula and algorithm can be used to partition all these pixels to two clusters. K-means clustering is a technique to partition *n* observations into *k* clusters ; each observation is identify with the closest mean, (x, y), which is measured as the mean of points in . K-means clustering decreases the within-cluster sum of squares by equation (11).

$$arc \; min \sum_{i=1}^{k} \sum_{pj(x,y)\in ci} ||p_j(x, y)$$
$$- \mu_i(x, y)||^2 \dots\dots\dots\dots (11)$$

K-means clustering can be used anytime which there is transform in the input data source at the beginning of each version. Each empty cluster is initialize with a random point identify within the zRange as the mean. After K-means converges, the points that belong to each hand must be group. If the length between the two centroids of hands is less than a pre-defined value, the two clusters are combined into one. Figure 5 shows the result of hand region detection.

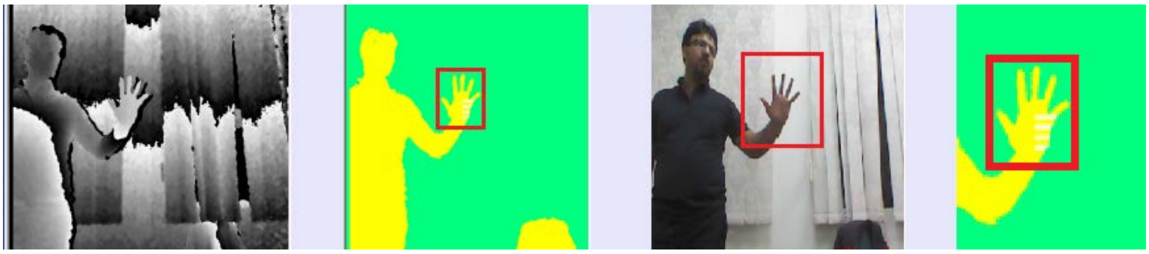

Figure 5: Detection of hand region

However, we still have to eliminate the pixels of un-hand such as body or head in hand region. Here, we hypothesized that the depth value in cropped region is distributed less than two class which is the hand and un-hand. For instance, the distribution of depth value in above figure is shown in Fig 6.

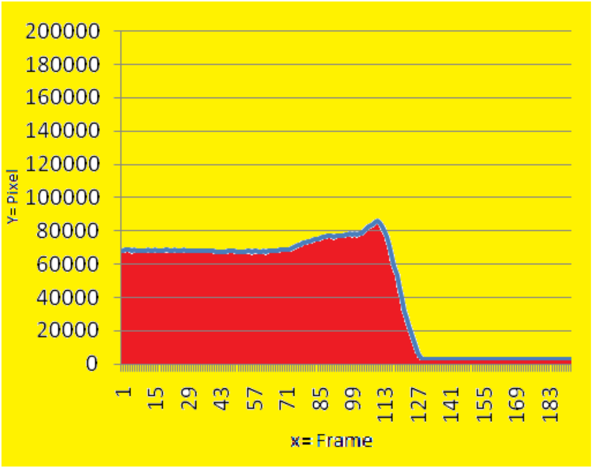

Figure 6: depth pixel in hand region by frame

VIII. DISCUSSION

The experimental result shows that if distance of the object is less than 2ft, we cannot obtain depth data through kinect. Moreover, when the object is in distance more than 10ft, the data will be lost. We have confirmed that the six degree freedom head can be used to detect head with more accuracy. Further, the K-means clustering formula and algorithm are less expensive to be used for hand detection.

IX. CONCLUSION

Kinect camera has an advantage over other 3D cameras because exact depth information can be obtained easily and very fast within a range of 0.4m to 4m which makes it useful for many HCI applications. Head and hand blobs detection can be extracted for single and multiple people in a scene. But its applications are limited to indoor usage only because of its light sensitivity.

X. REFERENCES

[1] Lee, S.-H. and S.-H. Oh, A Kinect Sensor based Windows Control Interface. International Journal of Control & Automation, 2014. 7(3).

[2] Kean, S., J. Hall, and P. Perry, Meet the Kinect: An Introduction to Programming Natural User Interfaces. 2011: Apress.

[3] Hsu, H.-m.J., The potential of kinect in education. International Journal of Information and Education Technology, 2011. 1(5): p. 365-370.

[4] Wang, J.-G. and E. Sung, EM enhancement of 3D head pose estimated by point at infinity. Image and Vision Computing, 2007. 25(12): p. 1864-1874.

[5] Leyvand, T., et al., Kinect identity: Technology and experience. Computer, 2011. 44(4): p. 94-96.

[6] Schramm, M., Kinect: The company behind the tech explains how it works. AOL Tech, http://www. joystiq. com/2010/06/19/kinect-how-it-works-from-the-company-behind-the-tech, 2010.

[7] Jaemin, L., et al. A robust gesture recognition based on depth data. in Frontiers of Computer Vision,(FCV), 2013 19th Korea-Japan Joint Workshop on. 2013. IEEE.

[8] Supplies, P., 3-D-Sensing Technology to “Project Natal” for Xbox 360.

[9] Shao, L., et al., Computer vision for RGB-D sensors: Kinect and its applications. IEEE transactions on cybernetics, 2013. 43(5): p. 1314-1317.

[10] Miranda, L., et al. Real-time gesture recognition from depth data through key poses learning and decision forests. in Graphics, Patterns and Images (SIBGRAPI), 2012 25th SIBGRAPI Conference on. 2012. IEEE.

[11] Rimkus, K., et al. 3D human hand motion recognition system. in Human System Interaction (HSI), 2013 The 6th International Conference on. 2013. IEEE.

[12] Guo, K., P. Ishwar, and J. Konrad. Action recognition in video by covariance matching of silhouette tunnels. in Computer Graphics and Image Processing (SIBGRAPI), 2009 XXII Brazilian Symposium on. 2009. IEEE.

[13] Tuzel, O., F. Porikli, and P. Meer, Region covariance: A fast descriptor for detection and classification, in Computer Vision–ECCV 2006. 2006, Springer. p. 589-600.

[14] Erol, A., et al., Vision-based hand pose estimation: A review. Computer Vision and Image Understanding, 2007. 108(1): p. 52-73.

[15] Athitsos, V. and S. Sclaroff. Estimating 3D hand pose from a cluttered image. in Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on. 2003. IEEE.

[16] Rosales, R., et al. 3D hand pose reconstruction using specialized mappings. in Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on. 2001. IEEE.

[17] de La Gorce, M., D.J. Fleet, and N. Paragios, Model-based 3d hand pose estimation from monocular video. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2011. 33(9): p. 1793-1805.

[18] Park, M., et al. Hand detection and tracking using depth and color information. in Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV'12). 2012.

[19] Scharstein, D. and R. Szeliski. High-accuracy stereo depth maps using structured light. in Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on. 2003. IEEE.

[20] Horn, B.K. and E. Weldon Jr, Direct methods for recovering motion. International Journal of Computer Vision, 1988. 2(1): p. 51-76.

[21] Henry, P., et al. RGB-D mapping: Using depth cameras for dense 3D modeling of indoor environments. in In the 12th International Symposium on Experimental Robotics (ISER). 2010. Citeseer.

How to cite

Mostafa Karbasi, Ahmad Waqas, Parham Nooralishahi, Seyed Mohammad Reza Mazloomnezhad, "Head and Hand Detection using Kinect Camera 360". *International Journal of Research in Computer Science*, 4 (6): pp. 1-7, June 2015.