Jurnal
Teknologi

Full Paper

REAL-TIME HAND DETECTION BY DEPTH IMAGES: A SURVEY

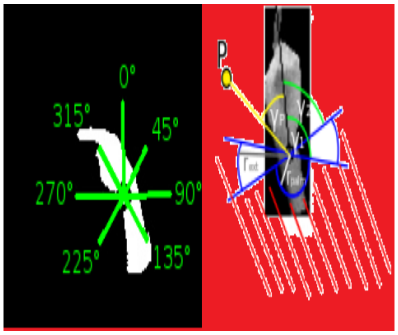
Article history
Received
27 August 2015
Received in revised form
23 September 2015
Accepted
15 January 2016

Mostafa Karbasi[a], Zeeshan Bhatti[a], Reza Aghababaeyan[b], Sara Bilal[a], Abdolvahab Ehsani Rad[c], Asadullah Shah[a], Ahmad Waqas[a]

[a]Khulliyyah of Information and Communication Technology, International Islamic University Malaysia
[b]Department of Computer, Rodehen Branch, Islamic Azad University, Rodehen, Iran
[c]Department of Computer Engineering, Faculty of Advance Informatics School, University Teknologi of Malaysia

*Corresponding author
mostafa.karbasi@live.iium.edu.my

Graphical abstract



Abstract

Human hand detection can enable human to communicate with a machine and interact without any external device. Human hands play an important role in different applications such as medical image processing, sign language translator, gesture recognition and augmented reality. A human hand has different length and breadth for male and female. So, it is a complex articulated object consisting many connected parts and joints. Traditional methods for hand detection and tracking used color and shape information from RGB camera. Using a depth camera for hand detection and tracking is a challenging and interesting domain in computer vision. Some research has shown that using depth data for hand detection can improve human computer interaction. Recently, researchers used depth data in different hand detection and tracking methods in real time application. This paper explains different types of methods which are used for human hand detection. Various techniques and methods are explored and analyzed in this survey to determine the shortfalls and future directions in the field of hand detection from depth data.

Keywords: HCI Application, Depth Camera, Hand Detection, Depth Data, Depth Images

1.0 INTRODUCTION

In current modern technological era, the use of computer systems and its various applications are deeply connected and embedded into our society. This technology oriented environment needs a new type of human computer interaction tools, which are natural and easy to use. The use of hand gestures to control the computers and their operations is now becoming a great need of time. The hand detection and tracking, are a very interesting research area to the scientists, due to potentially large number of application involving a lot complexity. This research problem and area of study deals with inferring the process and tracking of highly articulated and self-occluding non-rigid 3D object from images. This articulation can be caused due to:

- Noise in images.
- Complex object in scene.
- Complex object shapes.
- Loss of information.

There are different strategies used for hand detection and tracking since last two decades. One of the famous and common methods is skin color-base hand detection, which can detect hand region base on color information. Differentiating hands from overlapping hand and objects with similar skin color are limitation of skin color-base method [23] .Hand shape detection is an another strategy used by [14] which was able to distinguish hand by identifying the presence of human hands within an image and classifying the hand shape robustly. Further, Motion flow information is another modality that can fill pervious gaps under certain condition [25]. This system assumes that the hand is a faster moving object in the image frame. However, this method could not be suitable when the hand does not carry a fast motion. A part from pervious methods, there are few other systems, which work on appearance based detection. In [1-6-29] used Hairline features for hand detection but this approach is limited to a few postures of the hand.

To overcome these restrictions in the previous methods, some researchers have tried to use the depth information for hand detection and tracking. These researchers have attempted to enhance hand detection and tracking in their applications by applying existing methods with depth information. This paper tries to review various methods which are used for hand detection and tracking by using depth information.

2.0 HAND DETECTION APPROACHES USING DEPTH INFORMATION

There have been many techniques used throughout the literature, to detect the hand by depth data which were acquired from images after preprocessing. These various different approaches and techniques used for hand detection are further categorized and explained in the following sections.

2.1 Cluster Merging and Filtering

Kd-tree structure is used [19] to obtain color candidate cluster. The Figure 1 shows that red cluster is too small compared with other clusters. Blue and yellow cluster can be merged since Haus dorff distance between them is small enough. At the end of process three clusters are remained, from which, the green one is filtered and place in depth. Thus, the remaining clusters are labeled as a R and L hands.

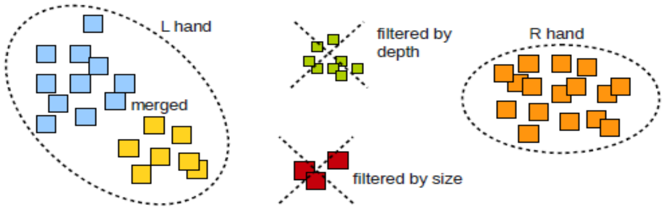


Figure 1 Cluster Merging and Filtering Hand Detection (Xavier Suau)

Merging: If the distance between two clusters is less than given distance threshold δmin, two clusters can be merged as a single cluster.

Size filtering: The obtained clusters are filtered by size and the largest one are kept. Using threshold Smin provides that which clusters will be accepted as hand candidates.

Depth filtering: All clusters obtained by previous step are stored by depth. Clusters which are closed to camera are selected as a hand candidate.

The number of detecting hands depends on the number of clusters that pass the merging and filtering steps, resulting in two, one or no hands being detected in.

2.2 Adaptive Hand Detection

Park, [15] proposed adaptive hand detection approach by using 3-dimentional information from Kinect and tracks the hand using GHT based method. First, the candidate's hand was detected through the regions from the histogram of the depth image, and candidate each region was ranked by using color information to reduce the number of detected candidate regions. Then, the boundary of the hand was obtained to get the exact positional accuracy. Actually, the depth image includes many unwanted portions of the hand regions because of noise and low resolution troubles. So, color information was used to compensate the disadvantage of the characteristics in depth image and improve the rate of accuracy for extracting the contour of hands. The overview of proposed method shown in Figure 2.

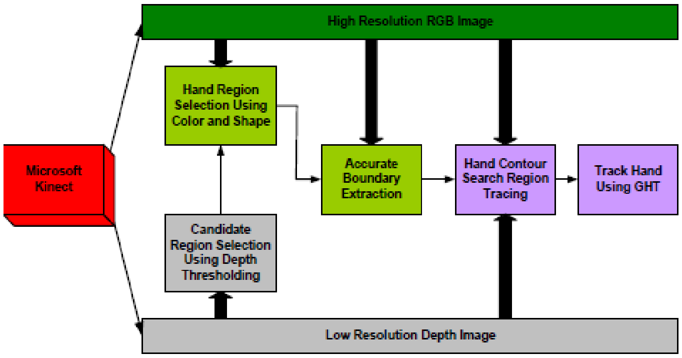

High Resolution RGB Image
Hand Region Selection Using Color and Shape
Microsoft Kinect
Accurate Boundary Extraction
Hand Contour Search Region Tracing
Track Hand Using GHT
Candidate Region Selection Using Depth Thresholding
Low Resolution Depth Image

Figure 2 Overview of Adaptive Hand Detection (Park *et al*., 2012)

2.3 3D Hand Model with Label Vertices

This method involving label vertices, incorporates training stage and estimation stage, was proposed by [27]. During the training stage both RGB and depth data used as a input variable and produced two classifier such as random forest and Bayesian classifier. Per-pixel hand part classification was obtained by forest classifier. Also, Bayesian classifier is employed as an object classifier to locate hand. Equation below shows that the shape feature is used for object segmentation.

$$\mathrm{acos}\big((x_{i-1}^{c}-x_{i}^{c}).(x_{i+1}^{c}-x_{i}^{c})\big) <\propto$$

$x^{c} = \ depth\, pixel\ ontar\ get\ counter$
$\propto$= Empirical threshold for each pixel x in the depth image

$$f_i(I,x) = s_I\left(x+\frac{u_i}{d_I(x)}\right)-s_I(x)$$

Where $\mathrm{d_I(x)}$ is the depth value of pixel x,$\mathrm{u_i}$ present an offset for i-th disparity feature

$$s_I(x) = \begin{cases} d_I(x) x \in hand \\ \quad cx \notin \ hand \end{cases}$$

2.4 Hand Region Growing Techniques

Chen, [5] used hand region growing techniques which includes 2 steps. In the first stage, hand position detection was obtained by using hand moving with a velocity. In second stage tried to segment entire hand region by using region growing technique on 3D point. a) Hand Position Prediction: position of the hand can be predicted based on hand movements and previous hand position $\mathrm{H_{t-1}}$
H= is the hand moving velocity estimate

$$H_t^{pred} = H_{t-1} + v,$$

b) Hand Region Segmentation: In the point cloud $\mathcal{P}$, the entire hand region can be found as a connected component. They used predicted hand position to find a seed point.

$$H_t^{seed} = \arg\underset{p\,\in\,\mathcal{P}}{min} d(\mathcal{P}, H_t^{pred})$$

Where d (.,.) = Euclidean distance between two points.
H_t^{seed}= is the nearest point cloud $\mathcal{P}$ from predicting hand position H_t^{pred}.

The connectivity between two points p, q in the point cloud $\mathcal{P}$ can be defined based on Euclidean distance as follows:

$$\text{Connected (p, q)} = \begin{cases} 1 & if d(p,q) < \delta \\ 0 & \quad otherwise \end{cases}$$

Where p = a pre-defined threshold specifying how far from each other two connected points can be[5].

2.5. Adaptive Skin Color and Depth Information

Hybrid RGB and TOF were proposed by [17 - 24]).They used skin color segmentation based on two methods. a) Gaussian mixture model (GMM), was trained offline and it was able to detect skin color under lighting condition. b) Histogram-based method which was trained online. Hybrid method can be obtained by multiplying the GMM-based skin color probability with histogram based skin color probability.

$$P_{GMM}(Skin|c).\frac{P_{hist}(c|Skin)}{P_{hist}(c|nonskin)}>T\ ,$$

C= Skin color
P_{GMM}= the Gaussian probability

$\mathrm{P_{min}(c|Skin)}$ and $\mathrm{P_{hist}(c|nonskin)}$= histogram-based probabilities that c belongs to the skin and nonskin classes respectively.
Some researchers used depth data to improve hand detection from TOF camera. By using position (x, y) and size (w, h) in Open CV function on RGB image, estimate average depth values of the hand.

$$d_{face} = \frac{1}{wh}\sum_{i=x}^{x+w}\sum_{j=y}^{y+h} \acute{I}_{ToF}(i,j)$$

$I_{T\acute{o}F}$= the projected depth image.

If the face is occluded, the previous position and distance is used. By using threshold all objects in front of the face can be detected.

$I_{ToF}(\imath,\jmath)\acute{} > d_{face} + t_s$
t_s = static parameter.

Interaction of hand can be accepted by distance range of hand from the camera. The arm is removed from hand if user wearing short sleeves. The distance from hand to camera is detected as soon as the hand is detected. System overview with the RGB and to F camera is shown in Figure 3.

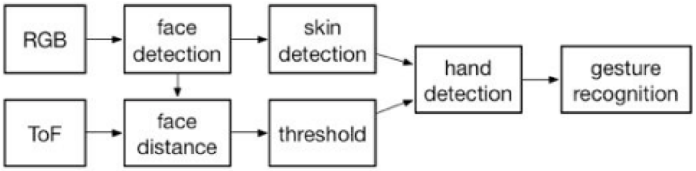


Figure 3 System Overview with the RGB and TOF Camera

2.6 Hand Bounding Box

Representing hand in box used [7]. They used threshold approach to compute bounding box. Depth data which is too close and too far from Kinect are considered as zero with fixing lower and upper thresholds. The function “CV Find Contours ()” was used to find the counter for the object. This technique takes a binary image and returns the number of retrieved contours. By using the Open CV function “CV Convex Hull ()”, its convex hull after obtaining contour could be computed. The external point of hand is represented by points of hull. This set of points is necessary for the function used to identify the fingers: “cv Convexity Defects ()”.In particular, the defects which are closer to the convex hull, determine the fingertip’s position. Other important points are the center of mass of the hand and the two wrist points at the vertices of the convex hull.

2.7 Hand Estimation by Labeled Vertices

Combine both a training stage and estimation stage was used by [27]. RGB and depth images used as an input in training stage and generate random forest classifier and a Baysian classifier. The former is for the per-pixel hand parts classification; and the latter is employed as an object classifier to locate hand. The estimation stage operates only depth image as input. By using depth data, all 2D/3D features are all constructed and send into per-pixel classifier to predict hand pose. By using this method, the background is extracted for the frame 0 and second classifier is employed to initialize the hand object. Figure4 shows hand poses estimation system.

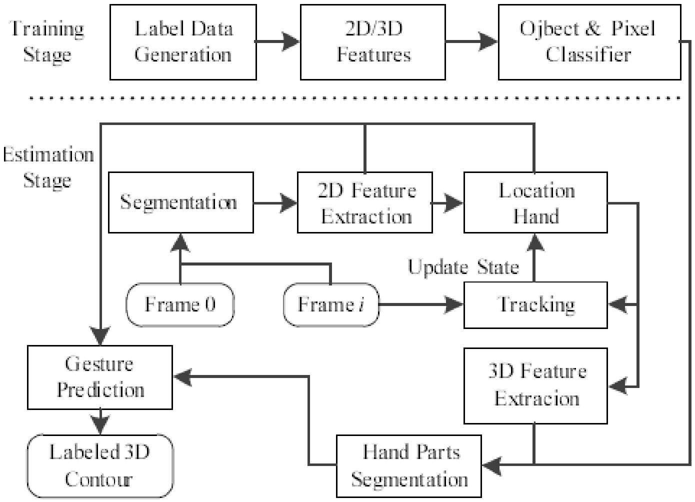


Figure 4 Band Pose Estimation System (Yao)

2.8 Threshold Method

This is a simple method of depth which was used to isolate the hand [2- 3- 12]. Depth threshold determines the hands to be used to those points between some near and far distance thresholds around the Z (depth) value of the expected centroid of the hand – which can be either predetermined and instructed to the user, or determined as the nearest point in the scene. An effective method to reduce susceptibility to noise is to also place bounds on the area of the detected hand – that is, place limits on the number of pixels expected in the blob segmented by depth threshold.

Depth hand detection was obtained by modifying depth threshold based on the location of other parts of body. On the other hand [4] without assuming the hand that was the nearest object in the scene used OpenCV7 to determine the head location and then estimate candidate hand locations.

This method applied to the depth histogram to segment the hand from the rest of the image. After threshold the image, the pixel co-ordinates$(x, y)$and the corresponding unscaled depth values (d) of the segmented hand region are extracted. Let X is the set of all the points extracted from each frame. Unscaled depthvalues are used to avoid the non-linear behavior of the depth map introduced by the camera. Figure 5 shows the Zcam camera, depth image of the Palm pose and its corresponding3-dimensional view as a heat map[20].

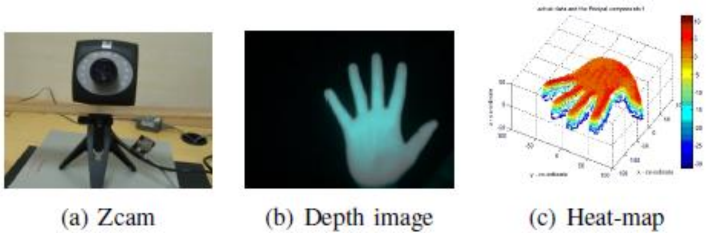

(a) Zcam (b) Depth image (c) Heat-map

Figure 5 Camera and Depth Data

In addition, [16] also applied threshold method with some other technique for hand and arm detection on depth information. They applied depth threshold, keeping pixels with depth$d < T_f - T_0$, where T_0 is a small value that typically represents the minimum distance from the face plane to the waving hand - They found out that a value of T0 T_0=100 is suitable for our dataset. Subsequently, we perform Connected Component Analysis (CCA) and keep the biggest component as the candidate arm. Since depth threshold may actually not perform a perfect segmentation, they further apply Otsu's segmentation algorithm on the final component to clean it from any background noise. Finally, we compute the Minimum Enclosing Ellipsoid (MEE) to find the elongation axis and rotate the arm in a horizontal position, such that the palm is always at the right side. as well [18] used threshold from nearest depth position with certain gap, a rough hand region can obtained. They used use RANSAC to locate the position of the black belt (user wear black belt on gesturing hand belt), and thus, a more precise hand shape can be detected.

2.9 Detection Base on Depth Images and Shape Recognition

Hamester, [8]. Perform foreground segmentation on depth images to reduce the region of interest. After that, edge detection in the foreground depth image provides a set of candidate contours. To support classification and the ability for generalization, Fourier descriptors with 12 complex-valued coefficients1were used to represent contours. These provide desirable invariance properties against common affine transformation (e. g. scale or rotation). Furthermore, the contour information is condensed in these 12 coefficients. Finally soft-margin support vector machines are used to separate hands from non-hands. Based on the color that is enclosed by a hand contour, the parameters of an elliptical boundary model (EBM) of the skin color distribution were found. In all subsequent frames after successful detection by shape, this model is used to retrieve the hand.

2.10 K-mean Clustering Algorithm

In the first stage, [11] applied threshold manually for specifying the depth range and then gestures was recognized. Hand pixels are projected to a 2D space for subsequent analysis. Distance between two pixels

$p_1 = (x_1, y_1)$ and $p_2 = (x_2, y_2)$ is defined as

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

To partial all pixels into two groups used K-mean clustering algorithm. For partitionn partition into k cluster used K-means clustering method. $C_1, C_2, C_3, \ldots, C_k$; Each observation is identified with the nearest mean, $\mu_i(x, y)$, which is calculated as the mean of points inC_i. K-means clustering minimized the within-cluster sum of squares.

$$arg_C^{min} \sum_{i=1}^{k} \sum_{(x,y)\in C} ||p_j(x, y) - \mu_i(x, y)||^2$$

As soon as, there is a change in the input data source can continuously applied k-mean clustering. Pixel belong to each hand can be clustered after using k-mean converges. To cluster can be merged if distance is less than predefined value.

2.11 Palm Detection

The size and position palm of single hand were calculated [21 after applying threshold depth data. The gravity of the hand image I is calculated:

$$\bar{c} = \frac{1}{|I|} \sum_{\bar{p}\in I} \bar{P}$$

Where pixel of hand is shown by$\overline{\mathrm{P}}$. By obtaining the center of hand. They calculated radius of palm r_{palm} with help of star like profile around the hand center. The profile can be rotated according the hand's orientation, which leading to seven directions as the figure below. In next step, they tried to measure largest distance from the center. As radius, we take the median of the distances scaled by □ = 1:065 to compensate for a small bias of the median towards smaller hand sizes. After obtaining palm by using distance, depth value which is not belong to hand and figures removed.
A point $\overline{\mathrm{P}}$is discarded if:

$$(\Upsilon_{\bar{p}} > \Upsilon_1 \wedge ||\bar{p} - \bar{c}|| > r_{palm} \vee (\Upsilon_{\bar{p}} > \Upsilon_2 \wedge ||\bar{p} - \bar{c} > \eta . r_{palm})$$

Where$\hat{\Upsilon}$=1.75.

This is illustrated in figure below. There is no figures when the range is $\Upsilon_{\vec{\mathrm{p}}} > \Upsilon_1$ and try to remove all figures do not belong to palm. When$\Upsilon_{\vec{\mathrm{p}}} > \Upsilon_2$ shows the region left and right of the hand.

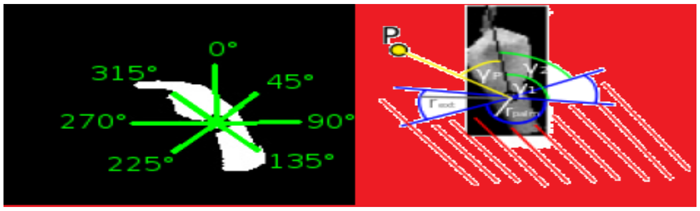


Figure 6 (a) Start Model for Radios of the Palm (b) Segmentation of Refinement Step

In addition [22], proposed a palm center detection method based in palm center coordinate (center of gravity of the hand). The spatial moments of image are computed as:

$$\boldsymbol{m_{i,j}} = \sum_{x,y} (\boldsymbol{f(x,y).x^j.y^i})$$

The central moments:

$$mu_{ij} = \sum_{x,y} (f(x,y).(x.\bar{x})^i.(y - \bar{y})^i)$$

Where $(\bar{x}, \bar{y})$ is the mass center?

$$\bar{x} = m_{10}/m_{00}$$
$$\bar{y} = m_{01}/m_{00}$$

This method improves accuracy and reduces the calculation time. Also, it can ensure the objective of palm center coordinate.

2.12 Randomized Decision Forest

Keskin, [9] applied this method for accurate hand detection and pose estimation with great accuracy result. The pixel location x and depth value image I are the input to an RFD. A set of posterior probabilities is an output for each hand partc_i. Given a depth image(x) , where x denotes location, the following equation feature was used:

$$F_{u,v}(i, x) = I\left(x + \frac{u}{I(x)}\right) - \left(x + \frac{v}{I(x)}\right)$$

Two values, u and v are relative to the pixel in hand and normalized according to the depth value at$\mathrm{x}$. They are not rotation and scale invariant and the features are 3D translations invariant. Each node is linked with the$\mathrm{u}$and$\mathrm{v}$, along with a depth threshold τ. The depth data is divided into two sets as follows.

$$C_L(u, v, \tau) = \{(I, x) | F_{u,v}(I, x) < \tau\}$$

$$C_R(u, v, \tau) = \{(I, x) | F_{u,v}(I, x) \geq \tau\}$$

A set of pixels assigned to left and right children of separate node is manually exclusively by $\mathrm{C_L}$ and $\mathrm{C_R}$.

Each split is scored by the total decrease in the entropy of the label distribution of the data:

$$S(u,v,\tau)=H(C)-\sum_{s\in|L.R|}\frac{|C_s(u,v,\tau)|}{|c|}H\left(C_s(u,v,\tau)\right)$$

Shannon entropy which is $\mathrm{H(K)}$ calculated by using the normalized histogram of the labels in the sample set K, the process ends when the leaf nodes are reached. Each leaf node is then related with the normalized histogram of the labels expected from the pixels reaching it.

2.13 Hand Detection Based on Distance

Based on the distance between the hand and the IR camera, depth image displays. The following transformational equation was used to obtain depth data.

$$dst(I)=src(I)+(shift_0,shift_1,\dots)$$

The shadow of infrared imaging noise grayscale still is 0, and the original image grayscale greater than 45 points changed to 255. Binarization process math model is:

$$g(x,y)=\begin{cases}0\; iff(x,y)>thresh\\ \quad f(x,y)other\end{cases}$$

Using above equation destination image realizes binarization process:

$$h(x,y)=\begin{cases}value if g(x,y)>thresh\\ 0 \qquad\qquad\qquad other\end{cases}$$

This method has great advantages such as fast segmentation, accurate extraction; hand segmentation based on depth image which was used by [22].

2.14 Multi Layered Randomized Decision Forest Network

Keskin, [10] Novel method Introduced to tackle the complexity problem. The idea is to reduce the complexity of the model by dividing the training set into smaller clusters, and to train PCFs on each of these compact sets. Thus, the part classification forest (PCF) need to model only a small amount of variation, requiring smaller memory. These experts a accurately model a specific subset of the data, and infer significantly better pose estimates. The main challenge is to direct the input towards the correct experts, which can be done by training a shape classification forest (SCF) on the clusters. The SCF assigns a cluster label to each pixel in an input image. This information can be used in two different ways: I) a pose label for the entire image can be estimated via voting; II) individual pixels can be sent to the corresponding expert PCFs according to their labels. These techniques are the Global Expert Network (GEN) and Local Expert Network (LEN) respectively. These networks are illustrated in Figure 4. The training of the multi–layered model requires three steps: I) clustering of the training data, II) training an SCF with the clusters as shapes, III) Training separate PCFs on each cluster.

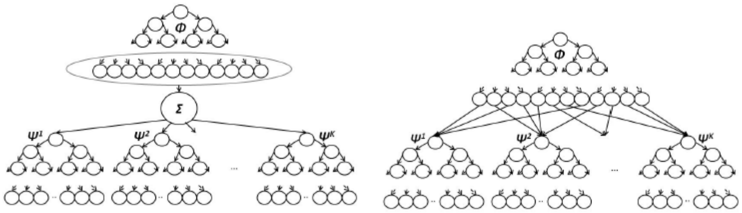


Figure 7 (a) Global Expert Network, (b) Local Expert Network (Cem)

2.15 Detection of Hand Region Using Skeleton Model

Most of researchers try to detect hand and head by using Kinect skeleton which can detect and track hand and head easily [26] and [28] used skeleton model for hand detection. They usually crop hand based on coordinate x, y which obtains from the skeleton. However, when the subject's hand is far from the sensor, the captured hand is small in the captured image. Therefore, optimum window size for cropping hand region have to be determined so that it does not involve the un-hand pixels such as body or head as soon as possible. Field of view depends on horizontal and vertical size of the camera. Resolution of depth image is based on distance between the sensor and object from of camera, so in this method, tried to measure distance of depth pixel value from camera. Then, the range that each pixel expresses is calculated by:

$$Z_{world}=0.00354\times Z_{world}$$

Where $\mathrm{Z_{world}}$

Depth is value from sensor and $\mathrm{X_{world}}$ is length per pixel. The hand size in real-world is defined as 250 × 250 mm. The window size of hand region is described by

$$K_{size}=\frac{250}{x_{world}}$$

3.0 CONCLUSION

In this paper, the previous research work on the hand detection by depth data has been reviewed. Our discussion has focused on different techniques using depth data by Kinect camera, summarized in Table 1. The result of this study shows that most of these techniques have some limitations for hand detection in real time. Hand detection has a crucial role in human computer interaction. These limitations have decreased the application performance. Based on the previous limitations such as detecting hand in only specific distance, seeing the hand as a single object in the scene, having box boundary around hand, missing the detection of hands when they change their place. In future, the researchers should focus and develop such techniques to improve these limitations in order to detect hand by depth data which can be used for human computer interaction applications. It is clear that future research in the area of hand detection is necessary to be improved to realize the ultimate goal of human detection with machines in the natural terms.

Table 1 Summary of all the relevant work done on hand detection

No.	Year	Authors	Methodology/ Technique
1	2004	Liu and Fujimura	Threshold Method
2	2006	Mo and Neumann	Threshold Method
3	2007	Breuer, Eckes, and Müller	Threshold Method
4	2011	Chen, Chen, Lee, Tsai, and Lei	Hand region growing techniques
5	2011	Ren, Meng, Yuan, and Zhan	Adaptive Skin Color and Depth Information
6	2011	Van den Bergh and Van Gool	Adaptive Skin Color and Depth Information
7	2011	Frati and Prattichizzo	Hand Bounding Box
8	2011	Biswas and Basu	Threshold Method
9	2011	Uebersax, Gall, Van den Bergh, and Van Gool	Palm Detection
10	2011	Van Bang Le and Zhu	Palm Detection
11	2011	Van Bang Le and Zhu	Hand Detection Based on Distance
12	2012	Keskin, Kıraç, Kara, and Akarun	Multi Layered Randomized Decision Forest Network
13	2012	Xiao, Mengyin, Yi, and Ningyi	Detection of Hand Region Using Skeleton Model
14	2012	Zainordin, Lee, Sani, Wong, and Chan	Detection of Hand Region Using Skeleton Model
15	2012	Suau, Ruiz-Hidalgo, and Casas	Cluster merging and filtering
16	2012	Park, Hasan, Kim, and Chae	Adaptive hand detection
17	2012	Yao and Fu	3D hand model with label vertices
18	2012	Yao and Fu	Hand Estimation by Labeled Vertices
19	2012	Cerlinca and Pentiuc	Threshold Method
20	2012	Li	K-mean Clustering Algorithm
21	2013	Jirak, and Wermter	Detection Base on Depth Images and Shape Recognition
22	2013	Keskin, Kıraç, Kara, and Akarun	Randomized Decision Forest
22	2014	Poularakis and Katsavounidis	Threshold Method

References

[1] Barczak, Andre LC, and Dadgostar, Farhad. 2005. Real-time Hand Tracking Using a Set of Cooperative Classifiers Based on Haar-Like Features.

[2] Biswas, K. K., and Basu, Saurav Kumar, 2011. Gesture Recognition using Microsoft Kinect®. Paper presented at the Automation, Robotics and Applications (ICARA). 2011 5th International Conference on.

[3] Breuer, Pia, Eckes, Christian, and Müller, Stefan. 2007. Hand Gesture Recognition with a Novel IR Time-Of-Flight Range Camera–A Pilot Study Computer Vision/Computer Graphics Collaboration Techniques. Springer. 247-260.

[4] Cerlinca, Tudor Ioan, and Pentiuc, Stefan Gheorghe. 2012. Robust 3D Hand Detection for Gestures Recognition Intelligent Distributed Computing. Springer. 259-264.

[5] Chen, Chia-Ping, Chen, Yu-Ting, Lee, Ping-Han, Tsai, Yu-Pao, and Lei, Shawmin. 2011. Real-time Hand Tracking on Depth Images. Paper presented at the Visual Communications and Image Processing (VCIP), 2011IEEE.

[6] Dadgostar, Farhad, and Sarrafzadeh, Abdolhossein. 2006. An Adaptive Real-time Skin Detector Based on Hue Thresholding: A Comparison on Two Motion Tracking Methods. *Pattern Recognition Letters*. 27(12): 1342-1352.

[7] Frati, Valentino, and Prattichizzo, Domenico. 2011. Using Kinect for Hand Tracking and Rendering in Wearable Haptics. Paper presented at the World Haptics Conference (WHC), 2011 IEEE.

[8] Hamester, Dennis, Jirak, Doreen, and Wermter, Stefan. 2013. Improved Estimation of Hand Postures Using Depth Images. Paper presented at the Advanced Robotics (ICAR). 16th International Conference on. 2013.

[9] Keskin, Cem, Kıraç, Furkan, Kara, Yunus Emre, and Akarun, Lale. 2012. Hand Pose Estimation and Hand Shape Classification Using Multi-Layered Randomized Decision Forests Computer Vision–ECCV. Springer. 852-863.

[10] Keskin, Cem, Kıraç, Furkan, Kara, Yunus Emre, and Akarun, Lale. 2013. Real Time Hand Pose Estimation Using Depth Sensors Consumer Depth Cameras for Computer Vision. Springer. 119-137.

[11] Li, Yi. 2012. Hand Gesture Recognition Using Kinect. Paper presented at the Software Engineering and Service Science (ICSESS), 3rd International Conference on. 2012 IEEE.

[12] Liu, Xia, and Fujimura, Kikuo. 2004. Hand gesture Recognition Using Depth Data. Paper presented at the Automatic Face and Gesture Recognition, 2004. Proceedings. Sixth IEEE International Conference on.

[13] Mo, Zhenyao, and Neumann, Ulrich. 2006. Real-time Hand Pose Recognition Using Low-Resolution Depth Images. Paper presented at the CVPR. 2.

[14] Ong, Eng-Jon, and Bowden, Richard. 2004. A Boosted Classifier Tree for Hand Shape Detection. Paper presented at the Automatic Face and Gesture Recognition, 2004. Proceedings. Sixth IEEE International Conference on.

[15] Park, MS, Hasan, Md Mehedi, Kim, Jaemyun, and Chae, Oksam. 2012. Hand detection and tracking using depth and color information. Paper presented at the Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV'12).

[16] Poularakis, Stergios, and Katsavounidis, Ioannis. 2014. Finger Detection and Hand Posture Recognition Based on Depth Information. Paper presented at the Acoustics, International Conference on. Speech and Signal Processing (ICASSP), 2014 IEEE.

[17] Ren, Zhou, Meng, Jingjing, Yuan, Junsong, and Zhang, Zhengyou. 2011. Robust Hand Gesture Recognition with Kinect Sensor. Paper presented at the Proceedings of the 19th ACM international conference on Multimedia.

[18] Ren, Zhou, Yuan, Junsong, and Zhang, Zhengyou. 2011. Robust Hand Gesture Recognition Based On Finger-Earth Mover's Distance with a Commodity Depth Camera. Paper presented at the Proceedings of the 19th ACM international conference on Multimedia.

[19] Suau, Xavier, Ruiz-Hidalgo, Javier, and Casas, Josep R. 2012. Real-time head and Hand Tracking Based on 2.5 D Data. Multimedia, IEEE Transactions on. 14(3): 575-585.

[20] Suryanarayan, Poonam, Subramanian, Anbumani, and Mandalapu, Dinesh. 2010. Dynamic Hand Pose Recognition Using Depth Data. Paper presented at the Pattern Recognition (ICPR), 2010. 20th International Conference on.

[21] Uebersax, Dominique, Gall, Juergen, Van den Bergh, Michael, and Van Gool, Luc. 2011. Real-time Sign Language Letter and

Word Recognition from Depth Data. Paper presented at the Computer Vision Workshops (ICCV Workshops), International Conference on. 2011 IEEE.

[22] Van Bang Le, Anh Tu Nguyen, and Zhu, Yu. Hand Detecting and Positioning Based on Depth Image of Kinect Sensor.

[23] Van den Bergh, Michael, Koller-Meier, Esther, Bosché, Frédéric, and Van Gool, Luc. 2009. Haarlet-based Hand Gesture Recognition for 3D Interaction. Paper presented at the Applications of Computer Vision (WACV). Workshop on. 2009.

[24] Van den Bergh, Michael, and Van Gool, Luc. 2011. Combining RGB and ToF Cameras for Real-Time 3D Hand Gesture Interaction. Paper presented at the Applications of Computer Vision (WACV). 2011 IEEE.

[25] Weng, JJ, and Cui, Y. 1998. Recognition of Hand Signs from Complex Backgrounds. Computer Vision for Human-Machine Interaction.

[26] Xiao, Zheng, Mengyin, Fu, Yi, Yang, and Ningyi, Lv. 2012. 3D Human Postures Recognition Using Kinect. Paper presented at the Intelligent Human-Machine Systems and Cybernetics (IHMSC), 4th International Conference on. 2012.

[27] Yao, Yuan, and Fu, Yun. 2012. Real-time Hand Poses Estimation from RGB-D Sensor. Paper presented at the Multimedia and Expo (ICME), International Conference on. 2012 IEEE.

[28] Zainordin, Faeznor Diana, Lee, Hwea Yee, Sani, Noor Atikah, Wong, Yong Min, and Chan, Chee Seng. 2012. Human Poses Recognition Using Kinect and Rule-Based System. Paper presented at the World Automation Congress (WAC), 2012.

[29] Reza, A. and S.F.S.Asrari. 2016. DIGITAL IMAGE OF WATERMARKING: A SURVEY. *Jurnal Teknologi*. 78(1): 209-216.