EURASIP Journal on Advances in Signal Processing
a SpringerOpen Journal

RESEARCH Open Access

3D hand tracking using Kalman filter in depth space

Sangheon Park[1], Sunjin Yu[2], Joongrock Kim[1], Sungjin Kim[2] and Sangyoun Lee[1*]

Abstract

Hand gestures are an important type of natural language used in many research areas such as human-computer interaction and computer vision. Hand gestures recognition requires the prior determination of the hand position through detection and tracking. One of the most efficient strategies for hand tracking is to use 2D visual information such as color and shape. However, visual-sensor-based hand tracking methods are very sensitive when tracking is performed under variable light conditions. Also, as hand movements are made in 3D space, the recognition performance of hand gestures using 2D information is inherently limited. In this article, we propose a novel real-time 3D hand tracking method in depth space using a 3D depth sensor and employing Kalman filter. We detect hand candidates using motion clusters and predefined wave motion, and track hand locations using Kalman filter. To verify the effectiveness of the proposed method, we compare the performance of the proposed method with the visual-based method. Experimental results show that the performance of the proposed method out performs visual-based method.

Keywords: hand detection, hand tracking, depth information

1. Introduction

Recently, human-computer interaction (HCI) technology has drawn attention as a promising man-machine communication method. Advancements of HCI have been led by associated developments of computing power, various sensors, and display techniques [1,2].

Interest in human-to-human communication modalities for HCI also has been increased. These include movements of human hands and arms. Human hand gestures are non-verbal communication that ranges from simple pointing to complex interactions between people. Main advantage of hand gestures is the ability of communication in the distance [3]. The use of hand gestures for HCI demands that the configurations of the human hand can be measurable by the computer. The performance highly depends on the accuracy of detection and tracking of hand locations. Current hand detection and tracking methods are using various sensors including directly attached to hand, special feature gloves, and color or depth images [4-7].

The hand detection and tracking via image sensor may be done with 2D or 3D information. However, as obtaining 3D information needs high computing power and high cost equipment, 2D methods have been more developed than 3D. In 2D hand detection and tracking methods, the most common method is a visual-based method, which uses information such as color, shape, and edge. Visual-based methods can be categorized as color-based and template-based methods. The color-based method starts by finding a hand region using color information (RGB, HSV, YCbCr). Then, a color histogram is made from the detected hand. Based on this color histogram the region which is similar to hand color can be tracked [8,9]. The template-based method creates an edge image through the color or gray image. The edge image is matched to the trained hand template, and then the hand is tracked [10].

However, hand movements generally occur in 3D space. Then, 2D method only can use 2D information, which eliminates the movement information along the z-axis. This makes the limitation of 2D methods inherently. Recently, the equipment for obtaining 3D information is becoming faster, more accurate, and cost-effective. This equipment includes depth sensors such as ToF cameras

* Correspondence: syleee@yonsei.ac.kr
[1]Department of Electrical and Electronic Engineering, Yonsei University, 134 Shinchon-Dong, Seodaemun-Gu, Seoul, Korea
Full list of author information is available at the end of the article

and PrimeSensor [11]. After the emergence of this equipment, real-time 3D hand tracking methods rapidly developed. For example, Breuer et al. [12] used an infra-red ToF camera to create a near real-time gesture recognition system. Grest et al. [13] proposed a human motion tracking method using a combination of depth and silhouette information.

In this article, we propose a novel real-time 3D hand tracking method in depth space using PrimeSensor with Kalman filter. We generate the motion image from depth image. Then, we detect hand candidates using motion clusters and predefined wave motion, and track hand locations using Kalman filter.

The organization of this article is as follows. In Section 2, related works are briefly reviewed. In Section 3, the preprocessing of depth information and the proposed hand detection and tracking method are described. In Section 4, several experiments of our hand tracking system are performed. Finally, we conclude the article in Section 5.

2. Background

2.1 Visual-based hand tracking

There are two well-known visual hand tracking methods: color- and template-based methods. In color-based methods, after initial hand detection, the color information is extracted from the specified initial region. This color information is made up of RGB-space pixel colors or transformed into HSI-space pixel colors. In [14], the color histogram is made from hue and saturation values of the region. Then, the obtained color histogram is used to hand tracking. In template-based methods, the initial hand is found by matching the whole image with a prepared trained hand template. The template is moved near to the initial hand region, and the matching point of the hand is found. This process is used for every frame [15].

Visual-based methods are natural tracking method. However, visual-based methods are highly affected by the illumination conditions. When using a color histogram or skin color probability density function, RGB, hue, and saturation values may change by illumination. This can make it difficult to find and track the hand. Also, when a specific part of the hand is occluded or shaded by an object, then hand tracking can fail [16,17].

2.2. Depth-based hand tracking

Depth-based hand tracking methods can be categorized into model-based and motion-based. Model-based hand tracking uses the 3D articulation model to fit the hand. The motion-based method uses hand motion in depth space.

Breuer et al. [12] proposed the model-based hand tracking in depth space. In order to estimate location and orientation of the hand, principal component analysis is used with 3D points. These 3D points are subsequently fitted to an articulated hand model for refinement of the first estimation. Also, Oikonomidis et al. [18] proposed a system using model-based full-degree-of-freedom hand model initialization and tracking in near real-time with Kinect. They optimized hand model parameters to minimize discrepancy between the appearance and 3D structure of hypothesized instances of a hand model and the actual hand observations. The tracker based on stochastic meta-descent for optimizations in high dimensional state spaces is proposed by Bray et al. [19]. This algorithm is based on a gradient descent approach with adaptive and parameter-specific step sizes. The hand tracker is reinforced by the integration of a deformable hand model based on linear blend skinning and anthropometrical measurements.

In motion-based hand tracking method, Holte et al. [20] proposed the view invariant gesture recognition system with the ToF camera. This method finds the motion primitives from an accumulated image based on 3D data. It detects movements using a 3D vision of 2D double differencing (subtracting the depth values pixel-wise in two pairs of depth images), thresholding, and accumulating.

2.3. Color information versus depth information

Figure 1 shows the color and depth images under different illumination conditions. Figure 1a, b shows the color and depth images with normal illumination condition. In contrast, Figure 1c, d shows them in low illumination condition. The figures show the sensitivity to illumination changes of color and depth images. As figures showing, the color image is very sensitive to illumination variation.

The ToF camera and the PrimeSensor are currently developed depth image sensors. Both sensors produce depth images that store the real depth value in each pixel. For example, the PrimeSensor stores in each pixel with 16 bits depth information. We have the image with 3D information X, Y, and Z-axis. The depth image also has some drawbacks. First, the depth image includes a lot of noise at the edge of objects. Second, it is hard to find invariant features of objects, because the depth information depends only on distance. Table 1 shows the summary of the advantages and disadvantages of the color and the depth information.

2.4. Kalman filter

Kalman [21] proposed a recursive method to solve the problem of linear filtering of discrete data. Providing many advantages in digital computing, Kalman filter is applied in a variety of research fields and real application areas [22]. The main procedure of Kalman filter is to estimate the state, then refine the state from the error.

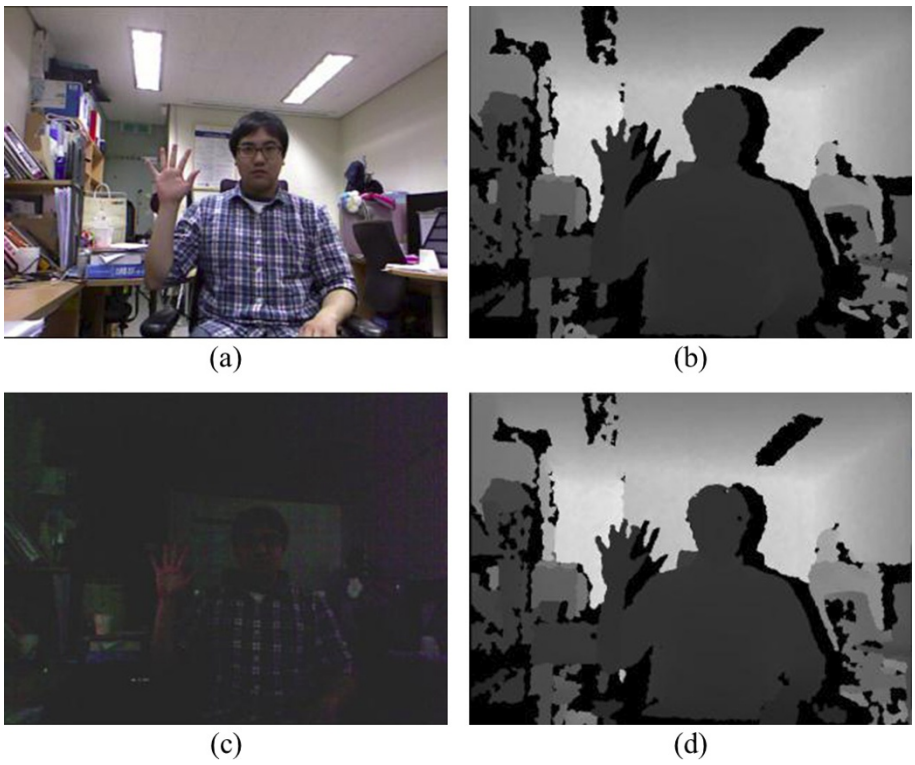

Figure 1 Comparing color and depth images under different illumination conditions. **(a)** color image in normal illumination; **(b)** depth image in normal illumination; **(c)** color image in low illumination; **(d)** depth image in low illumination.

The Kalman filter has two update procedures as shown in Figure 2. One is a control update and the other is a measurement update. In the control update, we estimate the state with the previous state and an action parameter (vector). In the measurement update, the state is corrected by sensor information. The equations of Kalman filter are presented in Table 2.

3. Proposed method

In this section, we explain the proposed hand detection and tracking algorithm. Figure 3 shows the steps of the proposed method. First, we get a depth image from the depth sensor, and create a motion image which is the accumulated difference images. Then, we reduce the noise with the spatial filter and the morphological operation. Motion clustering method is proposed to find motion clusters. Then, initial hand detection is performed among the clusters with wave motion. Finally, the Kalman filter is used to track the hand.

Table 1 Advantage and disadvantage of color and depth information

	Color information	Depth information
Advantage	Easy to find feature	Robust to light variation
	Non-intrusive method	Getting real depth value
		Non-intrusive method
Disadvantage	Sensitive to light conditions	Hard to find features
	Occlusion	Noise in edges
		Occlusion

3.1. Preprocessing

The depth image from the depth sensor has various sources of noise such as reflectance and mismatched patterns. Sometimes these noises are detected as real motion information. Therefore, noise reduction should be performed before hand detection. Also preprocessing includes clustering algorithm for initial hand detection.

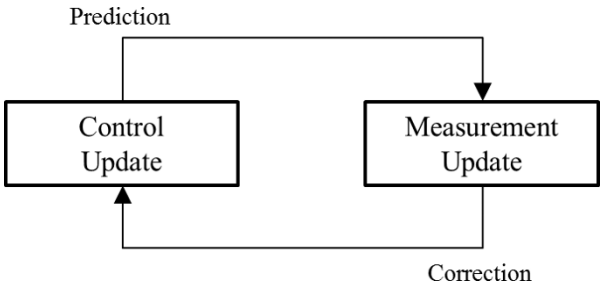

Figure 2 The procedures of Kalman filter.

Table 2 Summary of Kalman filter

Kalman filter	Control	Measurement
Update	$x_t = A_t x_{t-1} + B_t u_t + \varepsilon_t$	$z_t = C_t x_t + \delta_t$
Mean	$\bar{\mu}_t = A_t \mu_{t-1} + B_t u_t$	$\mu_t = \bar{\mu}_t + K_t(z_t - C_t \bar{\mu}_t)$
Covariance	$\bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + R_t$	$\Sigma_t = (I - K_t C_t)\bar{\Sigma}_t$
Kalman gain	-	$K_t = \bar{\Sigma}_t C_t^T (C_t \bar{\Sigma}_t C_t^T + Q_t)^{-1}$

3.1.1. Motion image (accumulated difference image)

We use the motion image which is the accumulated difference image. The process of generating the motion image is shown in Figure 4. First, we store five consecutive images in the chronological order. Then, we obtain the difference image which is the previous frame (i_{t-1}) subtracted from the current frame (i_t), as shown in (1).

$$\text{Diff\_image}_t = i_t - i_{t-1} \tag{1}$$

We accumulate difference images. In this accumulated image, all movement of human, object, and noise are represented. Next, noise reduction, motion clustering, and hand detection procedures are applied to this motion image.

3.1.2. Noise reduction

We use a spatial filtering and a morphological processing for noise reduction. When the noise reduction method is applied to the motion image, real motion can be shown clearly. A 5 × 5 aperture median filter is used for spatial filtering. The median filter replaces the pixel value with the median value of the sub-image with aperture [23]. This median filter provides excellent salt and pepper noise reduction with considerably less blurring. As the noise pattern of the motion image is very similar to salt and pepper noise, the median filter is very effective. We also use morphological processing for noise reduction. We use the opening operation which consists of erosion followed by dilation [23]. The basic effect of the opening operation is to reduce the outer shape of the object by erosion and to expand the outers. Generally, this operation smooths the outers, splits the narrow region, and removes the thin perimeter. Thus, the opening operation removes the randomly generated noise and smooths the original image. The erosion operation slips off the object or particles layer, reducing irrelevant pixels and small particles from the image. The dilation operation does the inverse of the erosion operation. It attaches layers to the object or particles, and it can return the eroded objects or particles to their original size. These operations are highly effective for the depth image noise reduction.

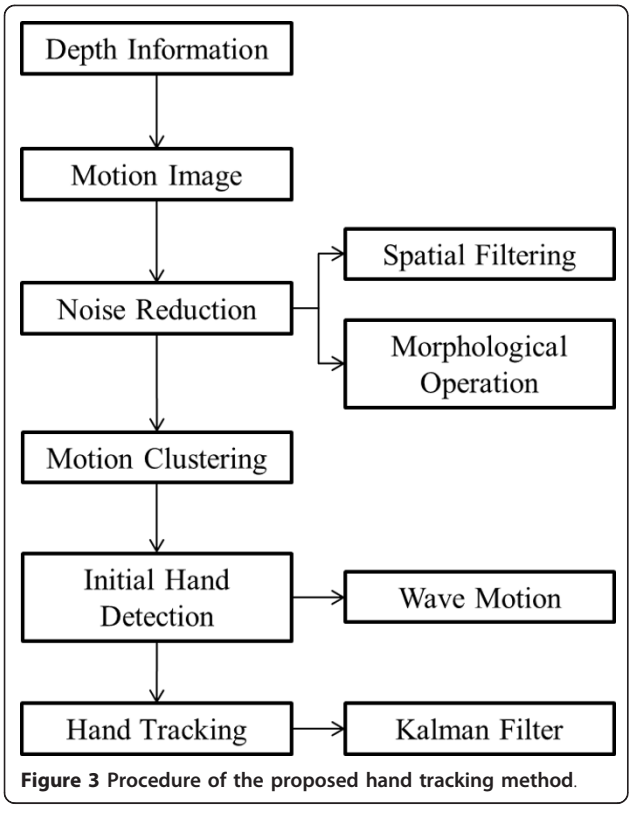

Figure 3 Procedure of the proposed hand tracking method.

Figure 5a shows the original motion image and Figure 5b shows the result of the noise removal methods of the spatial filtering and the morphological processing on our experimental motion image.

3.1.3. Motion clustering

In this section, we describe how to cluster motion regions from the motion image. First we select connected components from the motion image. Then the obtained connected components are clustered. These clusters are possible candidates for the hand. The selected clusters can be either real motion or noise. The noise clusters are usually small or split frequently, so if the size is smaller than some threshold, then we can decide it as a noise cluster, and remove it.

To decide the threshold of the size, we use polynomial regression method. First, we obtain the size of a hand from each distance of 60-750 cm with every 10-cm interval. With the obtained hand size data, we employ the polynomial regression method to fit a curve to the

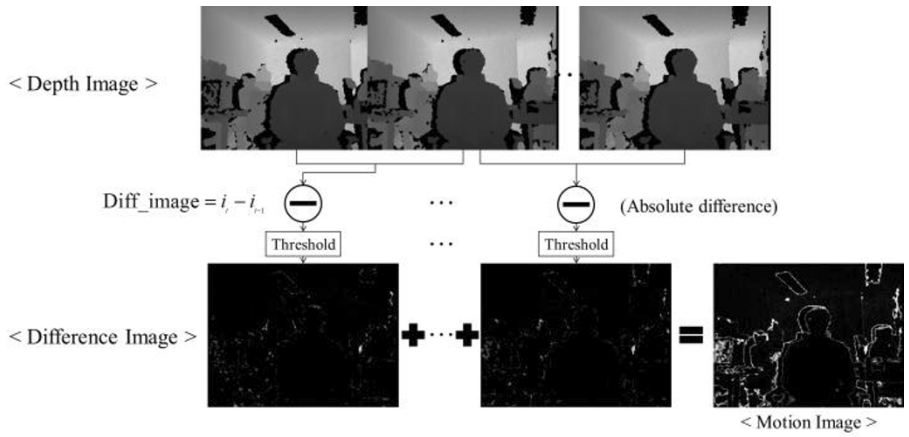

Figure 4 The process of generating the motion image.

dataset [24]. We use the fifth-order polynomial model given by (2)

$$g(\boldsymbol{\alpha}, x) = \boldsymbol{\alpha}^T p(x), \tag{2}$$

where

$$\boldsymbol{\alpha} = [\alpha_1\ \alpha_2\ \alpha_3\ \alpha_4\ \alpha_5\ \alpha_6]^T \tag{3}$$

and

$$p(x) = [\ 1\ x\ x^2\ x^3\ x^4\ x^5\ ]^T, \tag{4}$$

Because the fifth-order polynomial model is enough to model the obtained data. Given m data points, we use the least-squares error minimization objective given by (5)

$$s(\boldsymbol{\alpha}, x) = \sum_{i=1}^{m} [y_i - g(\boldsymbol{\alpha}, x_i)]^2 = [\mathbf{y} - \mathbf{p}\boldsymbol{\alpha}]^T [\mathbf{y} - \mathbf{p}\boldsymbol{\alpha}] \tag{5}$$

where $\mathbf{y} = [y_1,...,y_m]^T$ is the known data which we obtained in the hand size experiment. $\mathbf{p}$ represents the Jacobian matrix of $p(x)$:

$$\mathbf{P} = \begin{bmatrix} 1 & x_1 & x_1^2 & x_1^3 & x_1^4 & x_1^5 \\ 1 & x_2 & x_2^2 & x_2^3 & x_2^4 & x_2^5 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_m & x_m^2 & x_m^3 & x_m^4 & x_m^5 \end{bmatrix} \tag{6}$$

Finally, we can estimate the parameter vector $\boldsymbol{\alpha}$ from Equation (7), and the result is given in Equation (8).

$$\boldsymbol{\alpha} = (\mathbf{p}^T\mathbf{p})^{-1}\mathbf{p}^T\mathbf{y} \tag{7}$$

$$\boldsymbol{\alpha} = \begin{bmatrix} 327.982426870878 \\ -3.28281895300308 \\ 0.0152427222491653 \\ -3.54409649514619e-05 \\ 3.98934813043004e-08 \\ -1.73104221405172e-11 \end{bmatrix} \tag{8}$$

Then, we can find the fitted curve to the hand size dataset at any distance with Equation (9).

$$\hat{y} = \boldsymbol{\alpha}^T \cdot p \tag{9}$$

where $\hat{y}$ denotes the estimated number of pixels at distance p. Figure 6 shows the result of the fitted curve from 60 to 750 cm. In figure, the 'x' represents real hand-size data and the 'o' denotes the hand size estimated by the polynomial regression function.

Now, we can choose the threshold by this regression function. Figure 7a shows the result of motion clustering. The noise clusters still remain. Figure 7b shows the result of motion clustering with the threshold by the hand size. In the hand detection process, we find the hand cluster among those clusters.

We reduce the number of clusters by the polynomial regression method. Then, if other motions are overlapped behind a hand, the hand cannot be found, because the near region of the hand in the motion image turns into white. This situation is shown in Figure 8.

In order to find a hand cluster in this situation, we use the concept of bird's eye view image. The bird's eye view is an elevated view of a scene from above. This bird's eye view can be easily generated with 3D depth information. The depth image and the motion image are depicted on the X-Y plane. In the overlapped situation, however, we need to analyze X-Z plane information. The X-Z plane of the scene can be the bird's eye view as shown in Figure 9a. This figure is the X-Z plane of the original depth image. Then we consider this with the motion image above Figure 8b. We extract motion information from the original bird's eye view and generate Figure 9b. We call this figure as the motion bird's eye view. The white regions of Figure 9b represent the motion, which has the same meaning as the white

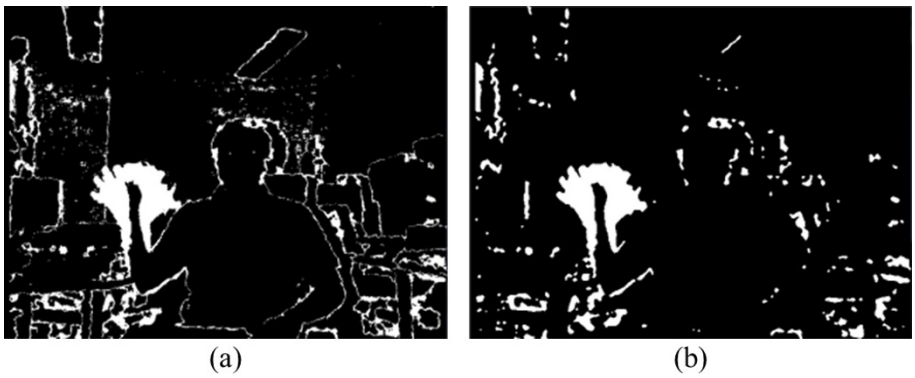

Figure 5 The original motion image and reduced noise motion image. **(a)** The original motion image; **(b)** reduced noise motion image.

regions of motion image. In Figure 9b, the small rectangle represents the front part which is the hand and the big rectangle represents the rear part which is the moving body. Therefore, we can separate the hand part from the moving body like Figure 8a.

3.2. Initial hand detection

In the preprocessing section, we generated the motion image by accumulating difference images, reduced the noise in the motion image, and found the motion clusters. In this section, we find the hand cluster from the remaining clusters in the image shown in Figure 7b.

To find the hand, we set the condition of hand wave motion, which consists of a side-to-side motion sequence. First, we detect the direction of cluster movements using a motion template [25,26]. The motion template is an effective method for tracking general movement, and it is especially useful for gesture

Figure 6 The fitted curve of hand size with the fifth-order polynomial regression function.

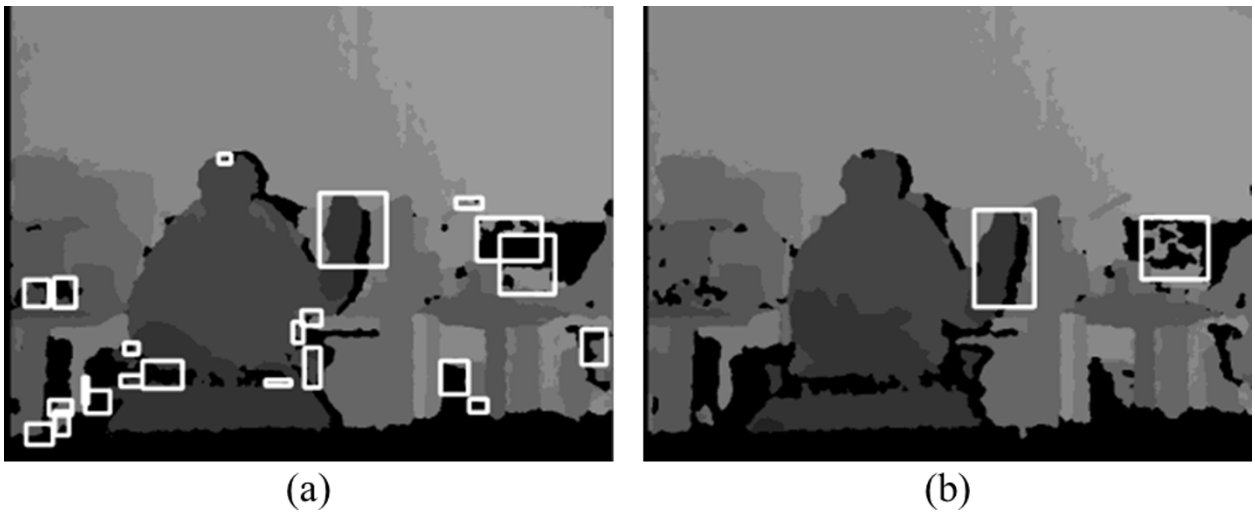

Figure 7 Motion clustering with hand size. **(a)** Before applying the threshold of hand size; **(b)** after applying the threshold of hand size.

recognition. A cluster is needed for using the motion template. We already obtain the clusters from the motion image. Thereafter, we assume that we have a well-segmented cluster which is the white rectangle shown in Figure 10a. This image is referred to as the motion history image. The white region of this image represents that all of the pixels in this region are set to the floating point. As the rectangle moves, a new cluster is calculated from the new current motion image and stacked to the motion history image. In Figure 10b, c, the white rectangle represents the new cluster and the previous cluster of old motions have become darker. The darkest rectangle denotes the oldest motion. And the rectangle is becoming lighter in consecutive order. These sequentially fading rectangles represent the movement of clusters. Figure 10d shows the motion history image in depth space.

From the motion history image, we can derive the direction by taking the gradient. The gradient can be calculated by the Sobel gradient function and the Scharr gradient. Some of gradients calculated from the motion history image are invalid. Those occur when non-movement regions have zero gradients and outer edges of the cluster have large gradients. Since we know the time between frames, we can calculate the range of gradients, and we can remove the invalid gradients. Finally, we can decide the global gradient as the direction. Figure 11 shows the direction of clusters. The line in the circle shows the direction that the clusters are moving toward.

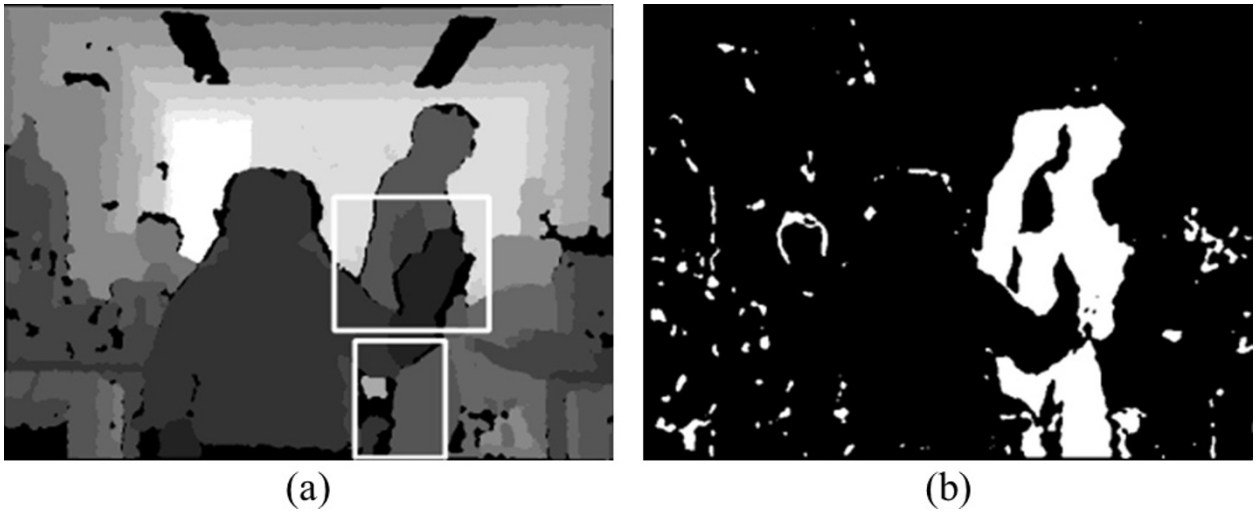

Figure 8 The overlapped situation in depth image and motion image. **(a)** Depth image of overlapped situation; **(b)** motion image of overlapped situation.

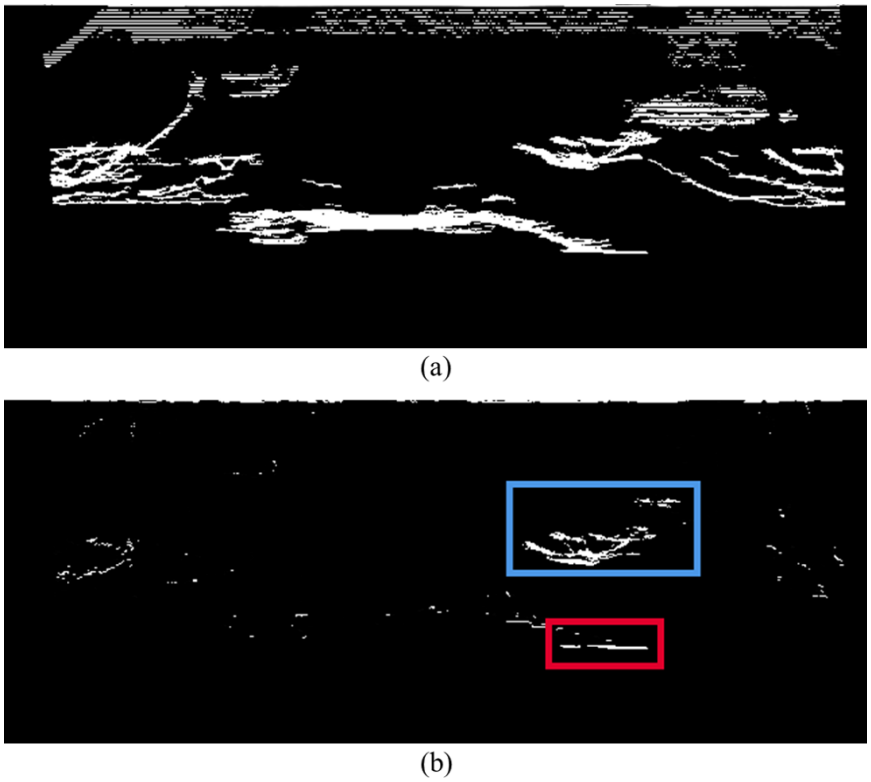

Figure 9 The bird's eye view and the motion bird's eye view. **(a)** The bird's eye view from original depth image; **(b)** the motion bird's eye view from motion image.

Next, we find the hand cluster using wave motion detection. From the movement clusters in Figure 7b, we can calculate their directions. Figure 12 shows the direction of clusters in the motion image.

The method that we use for detection of wave motion is counting the number of direction changes of the cluster. We set the condition of wave number to three times, and count the number of times that clusters move left to right. We also assume that the hand is in the closest position to the camera. With this assumption the hand is the part of the detected hand cluster with the smallest depth value. We use the depth histogram to find the hand in the selected cluster. Figure 13a shows the selected cluster, and Figure 13b shows the depth histogram of the cluster. In the depth histogram, we remove the pixels under 600 mm because PrimeSensor cannot measure the depth less than 600 mm. Therefore, we initialize the hand in the first peak of the depth histogram over 600 mm which is near 1000 mm.

Figure 14 shows the result of the initial hand detection. This detection method is robust to illumination conditions. But the edge noise and reflection noise can be regarded as motion clusters. Sometimes these noise clusters may satisfy the size condition and the wave motion, and this may falsely detected as the hand. We use a tracking method to eliminate possible false detection situation.

3.3. Hand tracking

In [15,27,28], many object tracking methods are explained. Among these, the Kalman filter has the following advantages for hand tracking. The first is computational efficiency; the Kalman filter needs small data storage for previous data in operating the recursive process, because we only need information of the previous state, and not the whole previous frame. The second advantage is that the Kalman filter is suitable for treating a time varying signal. Therefore, we apply the Kalman filter for hand tracking.

The Kalman filter is used for object tracking in many applications. Usually they use the state which is two dimensions for visual images. But as we need to add depth information, the state is designed as three dimensions. We assign depth information as z-axis values. So we make the state with three dimensions in (10).

$$S = \begin{bmatrix} s_x \\ s_y \\ s_z \end{bmatrix} \tag{10}$$

s_x and s_y represent the pixel position of the image, s_z represents the pixel value which is the depth value at position (s_x,s_y) in the depth image. This 3D state can more accurately estimate the hand position. For the

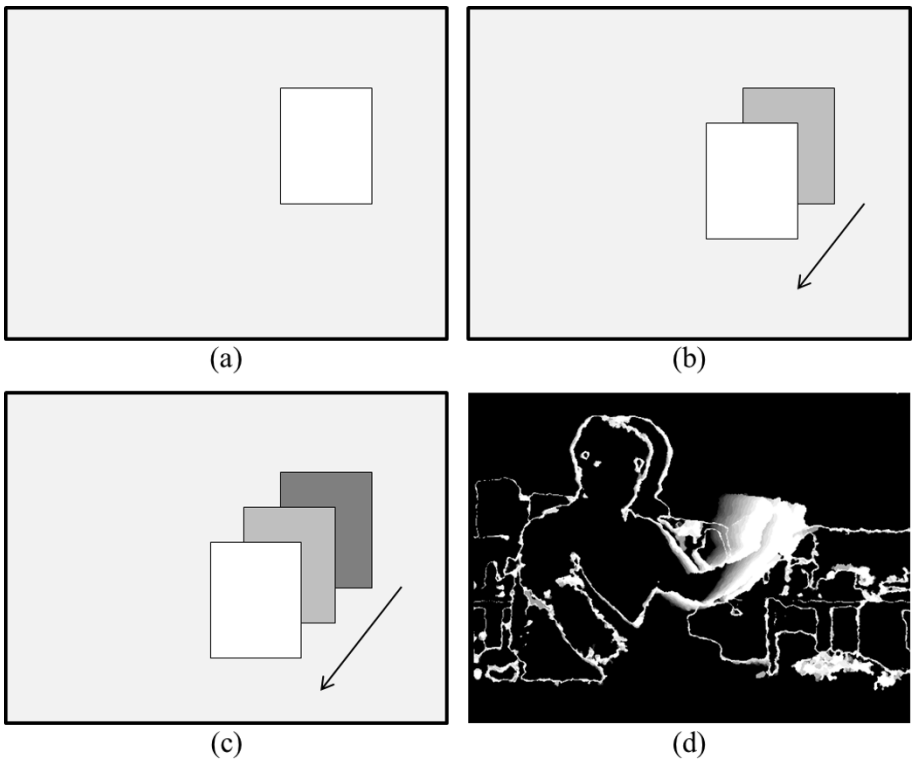

Figure 10 Motion history image and Motion template procedure. Motion history image at time **(a)** t; **(b)** $t + 1$; **(c)** $t + 2$; **(d)** Depth motion history image.

control vector, we use 3D vectors as the velocity of each axis in (11).

$$u = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} \tag{11}$$

For the measurement z, we use the same dimension of state S in (12).

$$z = \begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix} \tag{12}$$

These vectors are the initial setting for the Kalman filter.

The Kalman filter needs hand detection in every frame for tracking. We use the following hand detection method during tracking. First, we define the reference point in the hand. This point is obtained as the central point of an ellipse which fits to the detected hand in the initial hand detection process. The central point is the cross point of the major axis and the minor axis of the fitted ellipse. We use this reference point in tracking. The method of detection is that store the current reference point and cluster, and then find all the motion clusters in the next frame. Comparing with the previous selected cluster, we can choose the current selected clusters with Equation (13) and Equation (14).

$$\begin{aligned} &\text{Current}.x \text{ ¡ Previous}.x + \text{Previous}.width/2, \\ &\text{Previous}.x + \text{Previous}.width/2 < \text{Current}.x + \text{Current}.width. \end{aligned} \tag{13}$$

$$\begin{aligned} &\text{Current}.y \text{ ¡ Previous}.y + \text{Previous}.height/2, \\ &\text{Previous}.y + \text{Previous}.height/2 < \text{Current}.y + \text{Current}.height. \end{aligned} \tag{14}$$

The nominated motion clusters should be fitted to the hand size which is found from the polynomial regression method. This nominated point is now the current hand cluster, and we store the reference point.

Applications of Kalman filter for tracking usually fix the control update as constant. In our algorithm, the velocity of the hand continuously changes. We update the velocity of each axis for every frame. Therefore, the position of the tracked hand is more accurate. We apply the following equations to predict the state S.

$$s_{x,t} = s_{x,t-1} + v_{x,t-1}\Delta t, \tag{15}$$

$$s_{y,t} = s_{y,t-1} + v_{y,t-1}\Delta t, \tag{16}$$

$$s_{z,t} = s_{z,t-1} + v_{z,t-1}\Delta t, \tag{17}$$

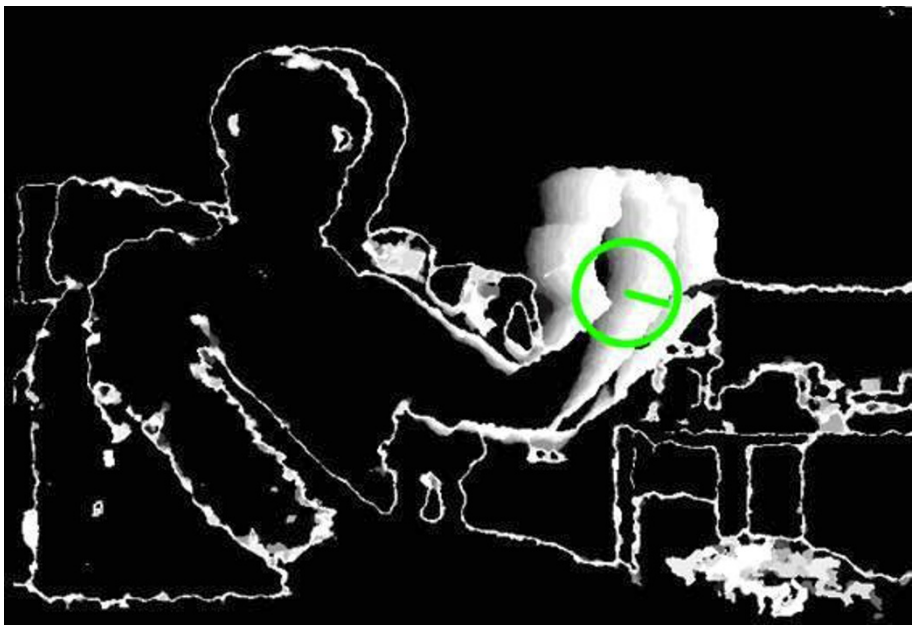

Figure 11 The direction of cluster.

We assume that the movement of the human hand is linear. $s_{x,\ t}$, $s_{y,\ t}$ and $s_{z,\ t}$ are the pixel and depth information at time t position. Δt is the interval time between the previous and the current frames. Equations (15) to (17) predict the position and the depth value from the hand position and the depth value of the previous position and control vector which is updated in the previous process. The $s_{z,\ t}$ has the limit, because the human hand moves within a limited reach range; we use not only X- and Y-axis limits, but also a Z-axis limit. We apply the following equations for updating elements of the control vector, (v_x, v_y, v_z) with

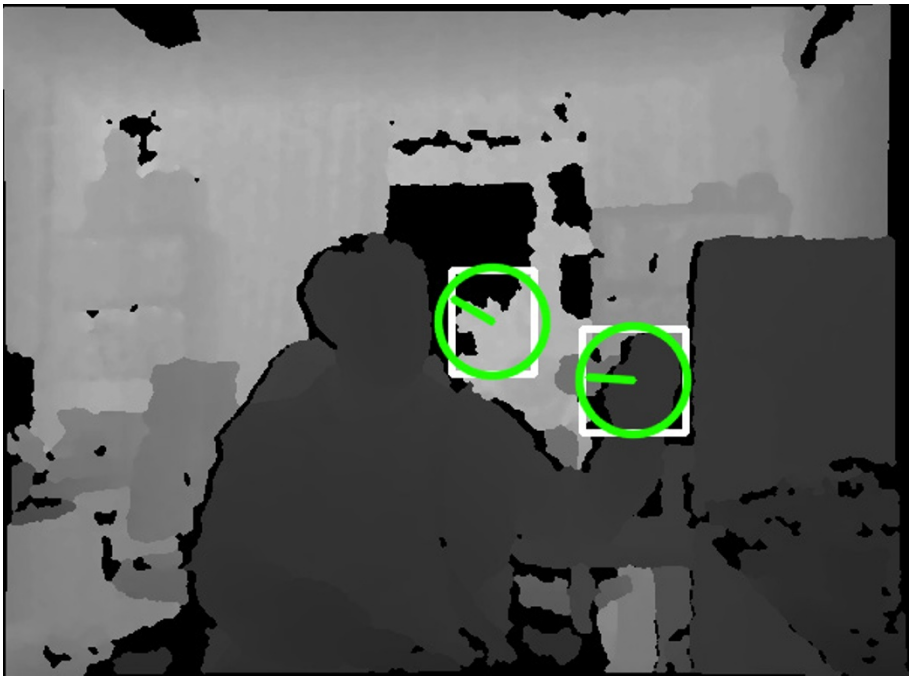

Figure 12 Finding the direction of clusters in motion image.

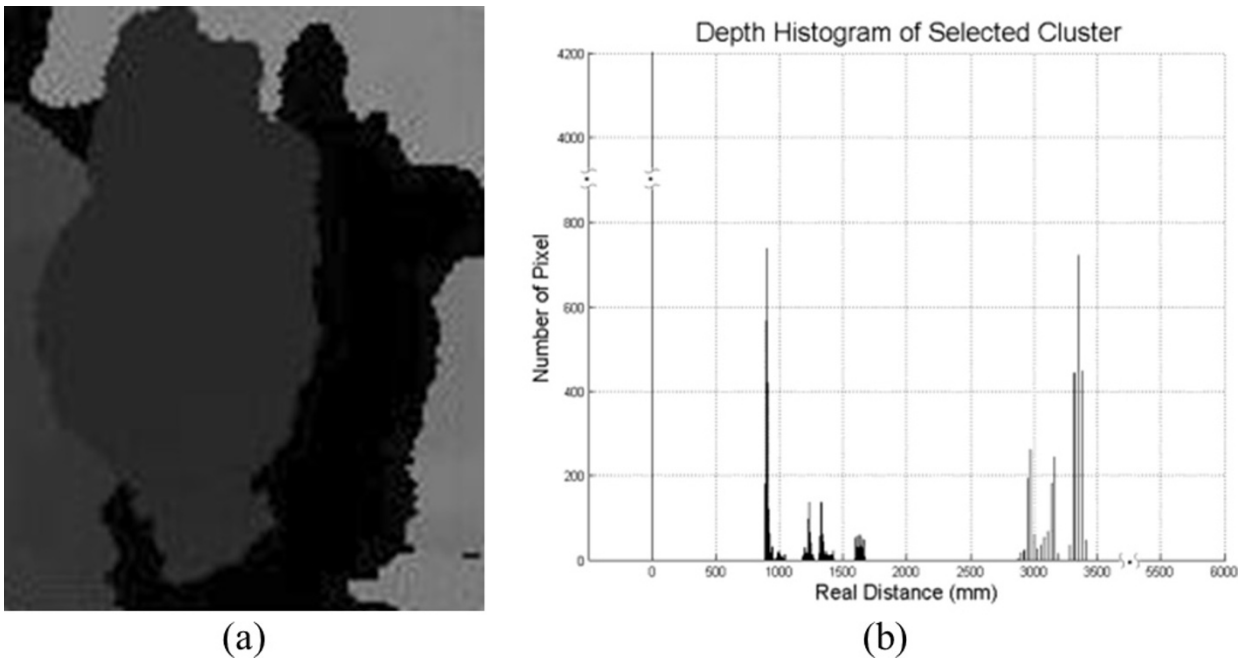

Figure 13 Selected cluster and depth histogram. **(a)** Selected cluster; **(b)** depth histogram of selected cluster.

(18) to (20).

$$v_{x,t} = (s_{x,t} - s_{x,t-1})/\Delta t, \tag{18}$$

$$v_{y,t} = (s_{y,t} - s_{y,t-1})/\Delta t, \tag{19}$$

$$v_{z,t} = (s_{z,t} - s_{z,t-1})/\Delta t, \tag{20}$$

Figure 14 Result of the initial hand detection.

Figure 15 Hand tracking using Kalman filter.

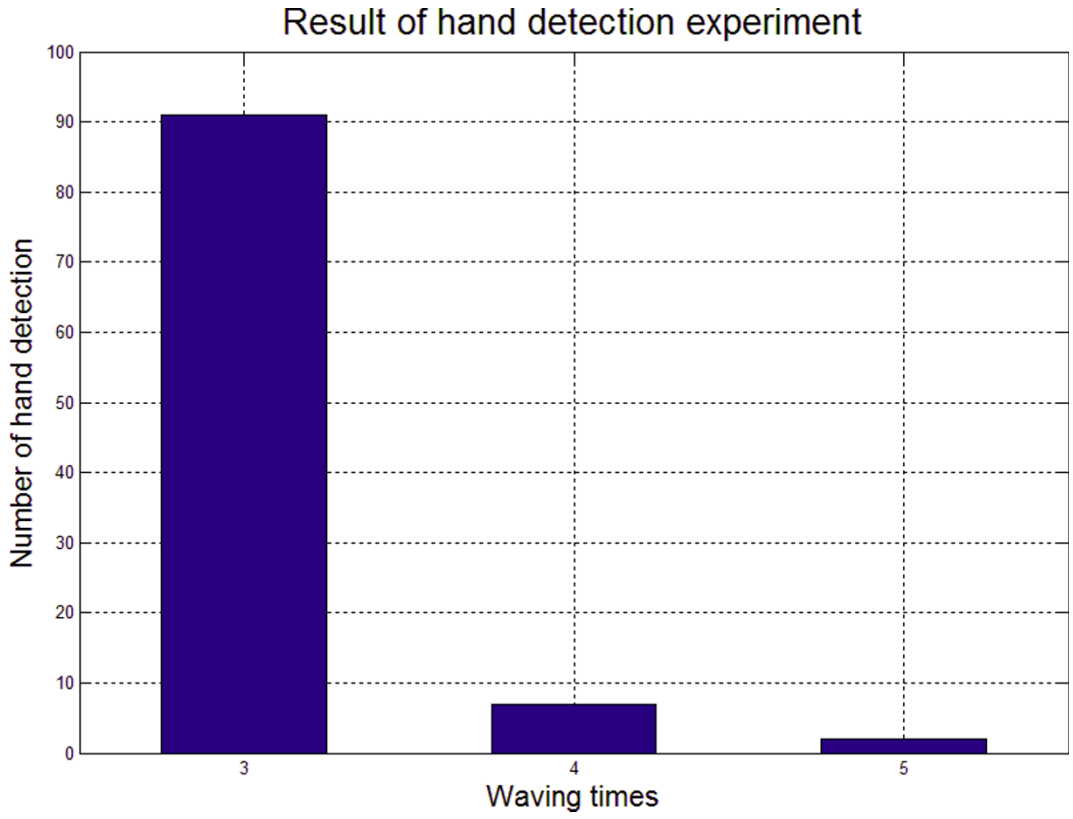

Figure 16 Result of the hand detection experiment.

Using the depth axis in Kalman filter tracking, we can track the hand more accurately and robustly. Figure 15 shows the result of hand tracking with depth information. The white point represents the current hand position and the gray points indicate the previous hand positions.

4. Experimental results

The experimental environment is a PC with Intel® Core™ i5 CPU 750 @ 2.67 GHz 2.66 GHz, and to obtain depth information we used Primesense's PrimeSensor development kit. The sensor obtained the depth image as follows. The IR light of PrimeSensor scatters the IR pattern, and the depth camera gets the pattern and creates the depth image. It also supports the color image. The resolution of the depth image is VGA (640 × 480), and the maximum frame rate is 60 fps. The resolution of the color image is UXGA (1600 × 1200). The operating range is 0.6-3.5 m [23]. In the proposed method, we use only the depth image.

4.1. Hand detection experiment

We perform hand detection experiment using the proposed initial hand detection method. We use the initial hand motion condition for finding the initial hand position. The wave motion is used as the initial hand motion. The experiment is performed 100 times and, we set three times of hand waving as the detecting condition. When the number of wave motion is 3, we assume hand detection is performed. Figure 16 shows the result of the hand detection experiment. The result says that the count of three times satisfying the above condition is 91%. And we can detect initial hand 100% at most five times waving motion of hand. In the experiment, the waving motion is continued until the system detects the hand.

4.2. Depth-based hand tracking experiment

The first hand tracking experiment is finding *X*- and *Y*-axis errors of the hand tracking at three distances. We manually gave the central hand position for the ground truth and compared it with the result of the proposed hand tracking.

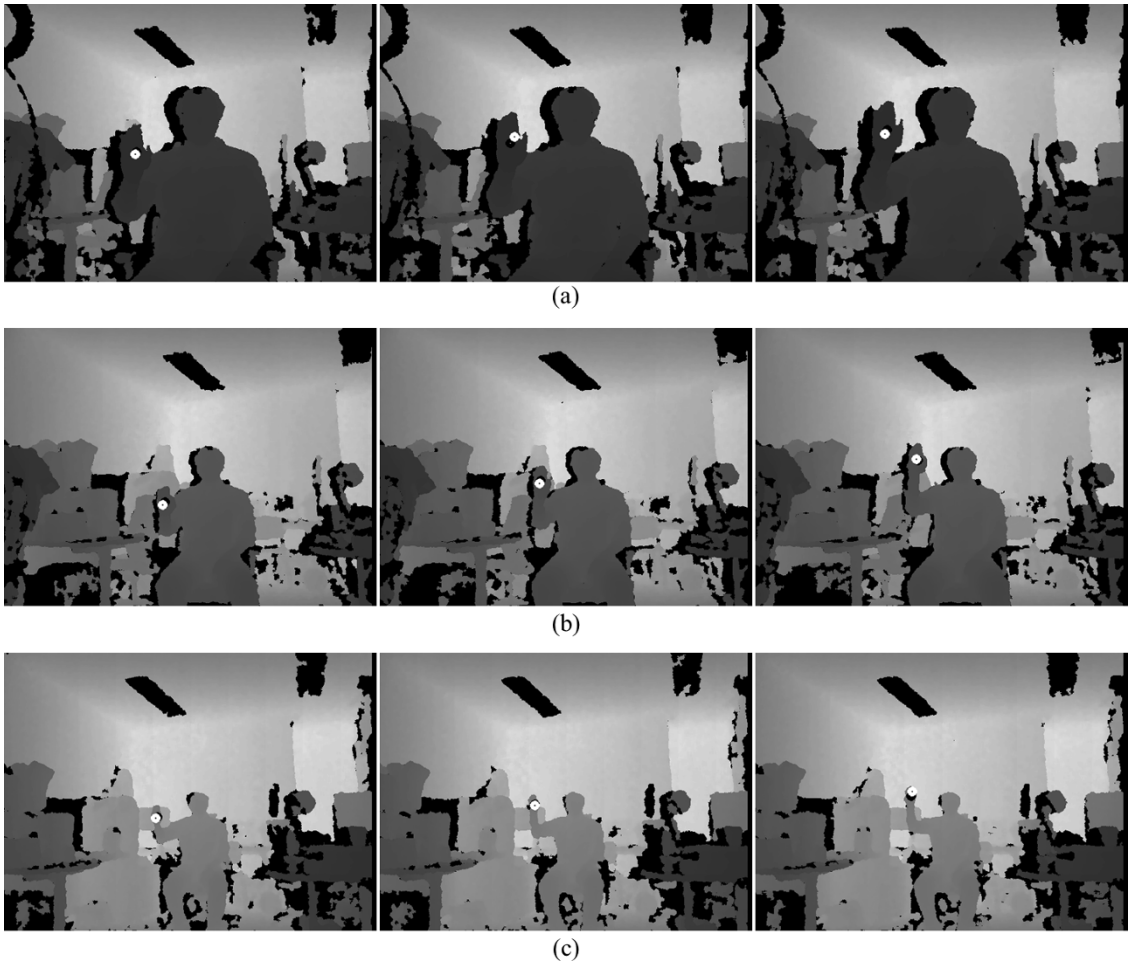

Figure 17 Result of the hand tracking experiment. Hand tracking **(a)** at 1 m distance; **(b)** at 2 m distance; and **(c)** at 3 m distance.

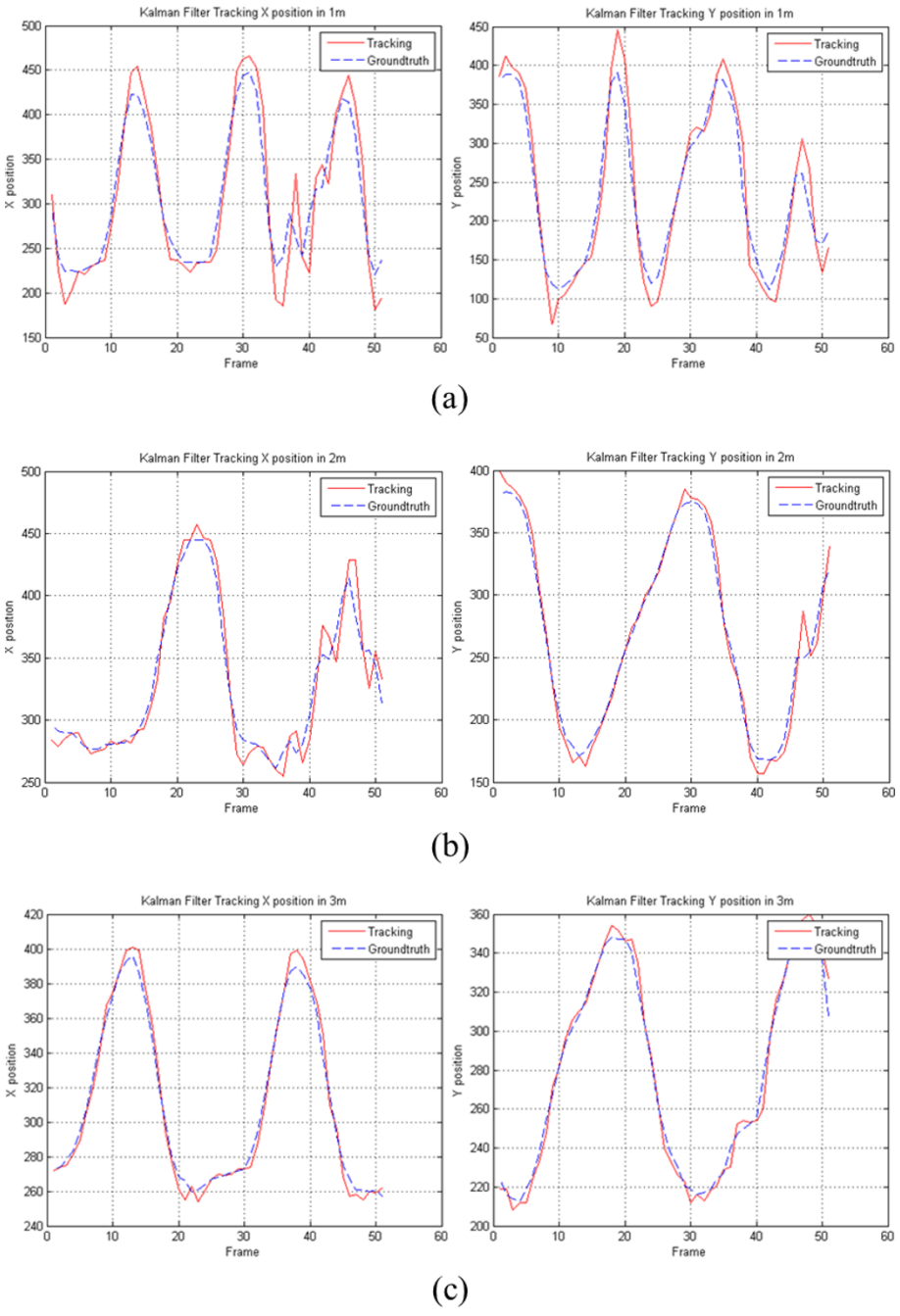

Figure 18 Result of the hand tracking experiment in 2D. *X* and *Y*-axis tracking result at **(a)** 1 m; **(b)** at 2 m; **(c)** at 3 m.

The movement of hand is the square and triangle shapes, as shown in Figure 17. In Figure 18, we show the measured distance error at each axis, *X* and *Y*. The dashed line represents the ground truth which is manually detected and the solid line represents the results of tracking using the Kalman filter. The results show that when tracking in the long distance smaller tracking error is observed compared to when tracking in the short distance.

Table 3 The result of tracking error in 2D

	1 m	2 m	3 m
Error (Pixel)	18.5623	7.6045	3.6696

The 2D errors of the hand tracking experiment are shown in Table 3 where the error unit is denoted by pixel. The largest error occurred in the 1-m experiment, and the smallest error in the 3-m tracking experiment.

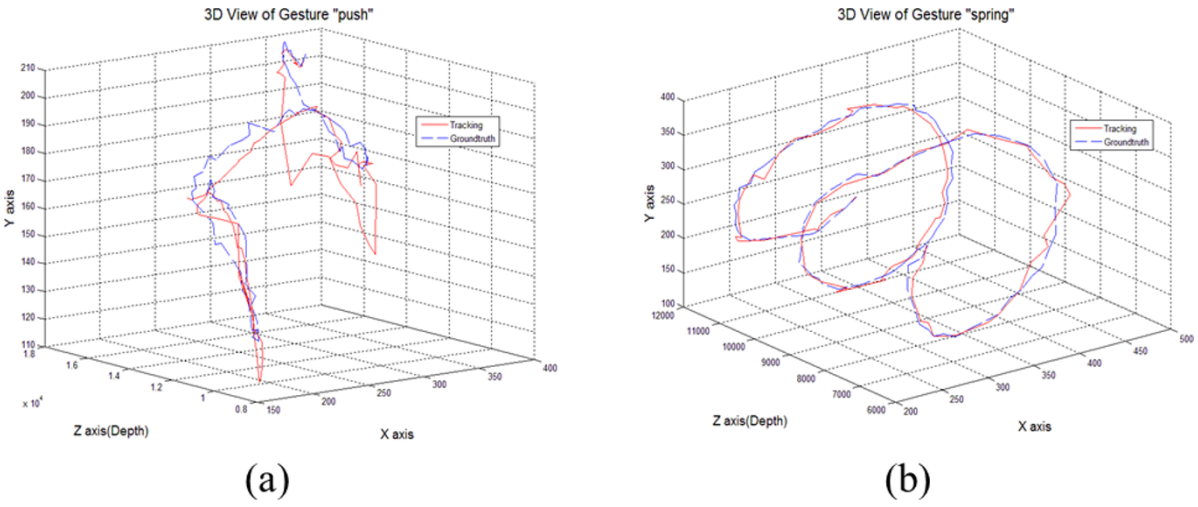

Figure 19 Result of the hand tracking experiment in 3D. 3D view of **(a)** push motion; **(b)** spring motion.

Because the hand motion should be smaller at the longer distance on each axis, the 3-m tracking result is more accurate than the 1-m tracking result.

The second hand tracking experiment is finding the error of the hand tracking in *Z*-axis. For this experiment, we use two types of motions, the one is a push motion and the other is a spring motion which draws a circle in a push motion. For the push motion, we push two times in one experiment. For the spring motion, we draw three circles. We obtained the data sets of each experiment from 12 persons with 10 times for each person.

Figure 19 a shows the result of 3D view of the push motion and Figure 19b shows the result of 3D view of the spring motion. Table 4 shows the average mean square error of each axis for 120 trials. The unit of *X*- and *Y*-axis error is pixel and unit of *Z*-axis is a millimeter. The error of *Z*-axis is refined by Kalman filter and hence the error is low in 13-30 mm.

4.3. Depth-based hand tracking and color-based hand tracking

We compare the performance of depth-based and color-based hand trackings. We used the Camshift [29] for the color-based hand tracking. After the initial hand detection, the hand is tracked by the proposed method with depth information, and independently by the Camshift with color information. For the Camshift tracker, the 5 × 5 window center is set to the initial hand point in order to extract the color histogram.

Table 4 The result of tracking error in 3D

	X-axis (pixel)	*Y*-axis (pixel)	*Z*-axis (mm)
Push motion	3.8867	6.4345	30.812
Spring motion	5.3170	5.0516	13.628

The ground truth is measured by color information with a marker which is attached to the hand. The depth and color information are calibrated. Therefore, we used the point of the ground truth for each tracking method.

The gestures of hand used for the experiment are the alphabet shapes such as 'a', 'b', 'c', and basic shapes 'square', 'triangle', and 'circle'. Each experiment is performed at distances of 1, 2, and 3 m. The datasets of each experiment are obtained from 10 persons with 10 times for each person. Figure 20 shows the result of each gesture at 3 m. The solid line of each figure is the result of the proposed depth-based hand tracking method, the dotted line represents the result of Camshift tracking method and the dashed line means the ground truth of hand.

The depth-based hand tracking method usually tracked the shapes well. The color-based method with Camshift fails when the hand overlapped an object or a face of similar hue intensity. Table 5 shows the average pixel error of the proposed depth-based tracking on *X* and *Y*-axis for 100 trials. Table 6 shows the average error of the color-based tracking on *X* and *Y*-axis for 100 trials. The distance of the ground truth for each tracking method is calculated in pixel units since Camshift cannot obtain depth information. The depth-based hand tracking is demonstrated to show the better performance than the color-based Camshift in the same bright light conditions. Under dark light conditions, the proposed method can track the hand well but the Camshift cannot extract the hue sample from the color image. Figure 21 shows the 3D plot of the proposed depth-based hand tracking result. Several points of results are sparked, but the Kalman filter refined those errors.

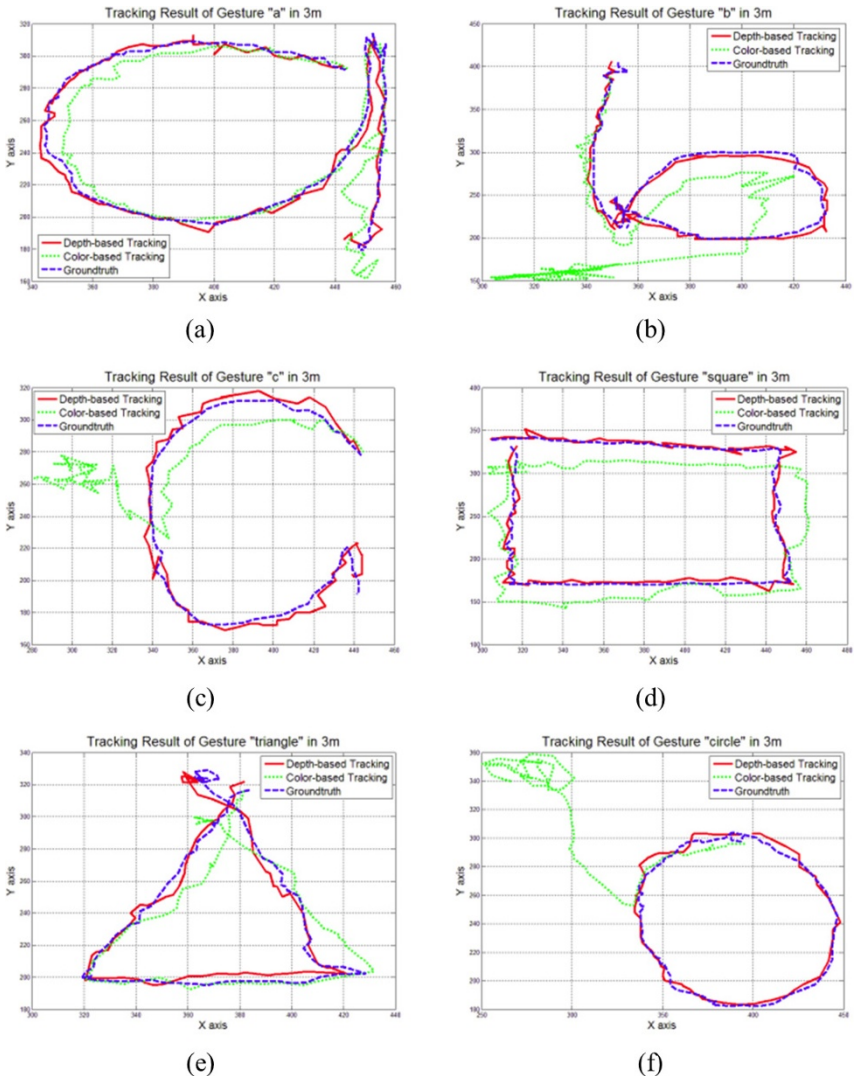

Figure 20 Result of the depth-based and color-based hand tracking. Result of gesture **(a)** 'a' tracking; **(b)** 'b' tracking; **(c)** 'c' tracking; **(d)** 'square' tracking; **(e)** 'triangle' tracking; and **(f)** 'circle' tracking.

5. Conclusion

We proposed a novel hand detection and a tracking method using depth information. We make the motion image, as a basic source of the proposed hand tracking system, which is the accumulated difference image from depth image sequences. In the preprocessing stage, we perform noise reduction, applying a spatial filtering and a morphological processing, and motion clustering, obtaining the moving region from the motion image. We detect the hand from this motion clusters using waving motion. We also suggest three-axis Kalman filter for tracking. Comparing the proposed method with

Table 5 The mean pixel error of depth-based hand tracking

		Gestures					
		a	*b*	*c*	Square	Triangle	Circle
Distance (pixel)	1 m	8.506	7.557	6.369	7.798	6.831	6.421
	2 m	5.837	6.544	5.053	5.218	5.119	6.223
	3 m	4.822	6.398	4.122	3.885	3.776	4.263

Table 6 The mean pixel error of color-based hand tracking

		Gestures					
		a	*b*	*C*	Square	Triangle	Circle
Distance (pixel)	1 m	51.571	11.921	10.059	33.482	69.155	63.262
	2 m	11.393	55.969	20.909	39.999	27.950	90.738
	3 m	13.887	38.549	69.643	14.707	14.949	77.681

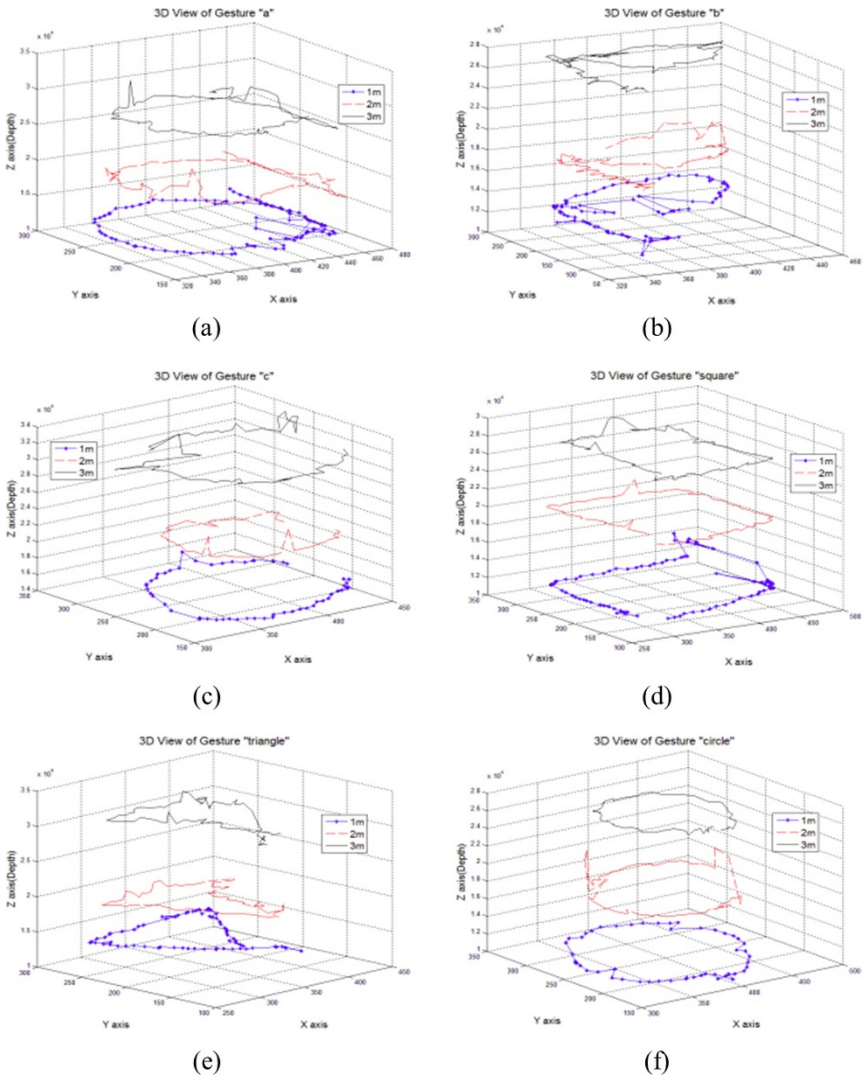

Figure 21 3D view of depth-based hand tracking result. 3D view of gesture **(a)** 'a'; **(b)** gesture 'b'; **(c)** gesture 'c'; **(d)** gesture 'square'; **(e)** gesture 'triangle'; **(f)** gesture 'circle'.

color-based method, we can see the effectiveness of the proposed method. Especially, the depth information-based method is very robust to the light variation environment. As for the future work, in order to improve the accuracy of tracking, more effective noise reduction methods or other tracking methods such as Unscented Kalman filter or particle filter can be considered.

Acknowledgements
This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (No. 2011-0016302). And this research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2010-0011472).

Author details
[1]Department of Electrical and Electronic Engineering, Yonsei University, 134 Shinchon-Dong, Seodaemun-Gu, Seoul, Korea [2]Future IT Convergence Lab, LG Electronics Advanced Research Institute, 221, Yangjae-Dong, Seocho-Gu, Seoul, Korea

Competing interests
The authors declare that they have no competing interests.

Received: 1 June 2011 Accepted: 17 February 2012
Published: 17 February 2012

References

1. VI Pavlovic, R Sharma, TS Huang, Visual interpretation of hand gestures for human-computer interaction: a review. IEEE Trans Pattern Anal Mach Intell. **19**(7), 677–695 (1997). doi:10.1109/34.598226
2. A Just, S Marcel, A comparative study of two state-of-the-art sequence processing techniques for hand gesture recognition. Comput Vis Image Understand. **113**(4), 532–543 (2009). doi:10.1016/j.cviu.2008.12.001
3. B Ionescu, D Coquin, P Lambert, V Buzuloiu, Dynamic hand gesture recognition using the skeleton of the hand. EURASIP J Appl Signal Process. **2005**, 2101–2109 (2005). doi:10.1155/ASP.2005.2101
4. DJ Sturman, D Zelter, A survey of glove-based input. IEEE Comput Graph Appl. **14**(1), 30–39 (1994)

5. RY Wang, J Popovic, Real-time hand-tracking with a color glove. ACM Trans Graph. **28**(3), 1–8 (2009)
6. C Shan, T Tan, Y Wie, Real-time hand tracking using a mean shift embedded particle filter. Pattern Recogn. **40**(7), 1958–1970 (2007). doi:10.1016/j.patcog.2006.12.012
7. Z Li, R Jarvis, Real time hand gesture recognition using a range camera, in *Proceedings of Australasian Conference on Robotics and Automation (ACRA)*, Sydney, Australia, (December 2009)
8. R Kjeldsen, J Kender, Toward the use of gesture in traditional user interfaces, in *Proceedings of the Second International Conference on Automatic Face and Gesture Recognition*, Killington, VT, USA, pp. 151–156 (October 1996)
9. K Imagawa, S Lu, S Igi, Color-based hands tracking system for sign language recognition, in *IEEE International Conference on Automatic Face and Gesture Recognition*, Nara, Japan, pp. 462–467 (April 1998)
10. B Stenger, A Thayananthan, PHS Torr, R Cipolla, Model-based hand tracking using a hierarchical bayesian filter. IEEE Trans Pattern Anal Mach Intell. **28**(9), 1372–1384 (2006)
11. PrimeSensor http://www.primesense.com
12. P Breuer, C Eckes, S Muller, Hand gesture recognition with a novel IR time-of-flight range camera–a pilot study. Lecture Note Comput Sci. **4418**, 247 (2007). doi:10.1007/978-3-540-71457-6_23
13. D Grest, V Kruger, R Koch, Single view motion tracking by depth and silhouette information, in *Proceedings of the Scandinavian Conference on Image Analysis*, Aalborg, Denmark, pp. 719–729 (June 2007)
14. C Manresa, J Varona, R Mas, F Perales, Hand tracking and gesture recognition for human-computer interaction. Electron Lett Comput Vis Image Anal. **5**(3), 96–104 (2005)
15. A Yilmaz, O Javed, M Shah, Object tracking: a survey. ACM Comput Surv. **38**(4), 1–45 (2006)
16. TB Moeslund, A Hilton, V Kruger, A survey of advances in vision-based human motion capture and analysis. Comput Vis Image Understand. **104**(2-3), 90–126 (2006). doi:10.1016/j.cviu.2006.08.002
17. H Lu, KN Plataniotis, AN Venetsanopoulos, A full-body layered deformable model for automatic model-based gait recognition. EURASIP J Adv Signal Process. **2008**, 13 (2008)
18. I Oikonomidis, N Kyriazis, AA Argyros, Efficient model-based 3D tracking of hand articulations using Kinect, in *British Machine Vision Conference*, Dundee, UK, pp. 101.1–101.11 (August 2011)
19. M Bray, E Koller-Meier, P Muller, LV Gool, NN Schraudolph, 3D hand tracking by rapid stochastic Gradient Descent using a skinning model, in *1st European Conference on Visual Media Production*, London, UK, pp. 59–68 (March 2004)
20. MB Holte, TB Moeslund, P Fihl, Fusion of range and intensity information for view invariant gesture recognition, in *IEEE Computer Society Conference on Computer Vision & Pattern Recognition Workshops*, Anchorage, AK, USA, pp. 1–7 (June 2008)
21. RE Kalman, A new approach to linear filtering and prediction problems. Trans ASME J Basic Eng. **82**, 34–45 (1960)
22. G Bishop, G Welch, An introduction to the Kalman filter, in *SIGGRAPH 2001, Course 8*, Los Angeles, CA, USA (August 2001)
23. RC Gonzalez, RE Woods, *Digital Image Processing*, 3rd edn. (Prentice Hall, Upper Saddle River, NJ, 2008)
24. KA Toh, QL Tran, D Srinivasan, Benchmarking a reduced multivariate polynomial pattern classifier. IEEE Trans Pattern Anal Mach Intell. **26**(6), 740–755 (2004). doi:10.1109/TPAMI.2004.3
25. GR Bradski, JW Davis, Motion segmentation and pose recognition with motion history gradients. Mach Vis Appl. **13**(3), 174–184 (2002). doi:10.1007/s001380100064
26. R Munoz-Salinas, R Medina-Carnicer, F Madrid-Cuevas, A Carmona-Poyato, Depth silhouettes for gesture recognition. Pattern Recogn Lett. **29**(3), 319–329 (2008). doi:10.1016/j.patrec.2007.10.011
27. FL Lewis, *Optimal Estimation: With An Introduction to Stochastic Control Theory* (Wiley, NY, 1986)
28. RG Brown, PYC Hwang, *Introduction to Random Signals and Applied Kalman Filtering* (Wiley, NY, 1997)
29. GR Bradski, Computer vision face tracking for use in a perceptual user interface, in *IEEE workshop on Applications of Computer Vision*, (Princeton, NJ, USA, 1998), pp. 214–219

doi:10.1186/1687-6180-2012-36
Cite this article as: Park *et al.*: 3D hand tracking using Kalman filter in depth space. *EURASIP Journal on Advances in Signal Processing* 2012 **2012**:36.