

# An Improved Approach for 3D Hand Pose Estimation Based on a Single Depth Image and Haar Random Forest

Wonggi Kim<sup>1</sup>, Junchul Chun<sup>1</sup>

<sup>1</sup>Department of Computer Science, Kyonggi University  
Suwon, 443-760, South Korea

[e-mail: bemong1@naver.com, jcchun@kgu.ac.kr]

\*Corresponding author: Junchul Chun

*Received March 8, 2015; revised June 30, 2015; accepted July 29, 2015;  
published August 31, 2015*

---

## Abstract

A vision-based 3D tracking of articulated human hand is one of the major issues in the applications of human computer interactions and understanding the control of robot hand. This paper presents an improved approach for tracking and recovering the 3D position and orientation of a human hand using the Kinect sensor. The basic idea of the proposed method is to solve an optimization problem that minimizes the discrepancy in 3D shape between an actual hand observed by Kinect and a hypothesized 3D hand model. Since each of the 3D hand pose has 23 degrees of freedom, the hand articulation tracking needs computational excessive burden in minimizing the 3D shape discrepancy between an observed hand and a 3D hand model. For this, we first created a 3D hand model which represents the hand with 17 different parts. Secondly, Random Forest classifier was trained on the synthetic depth images generated by animating the developed 3D hand model, which was then used for Haar-like feature-based classification rather than performing per-pixel classification. Classification results were used for estimating the joint positions for the hand skeleton. Through the experiment, we were able to prove that the proposed method showed improvement rates in hand part recognition and a performance of 20-30 fps. The results confirmed its practical use in classifying hand area and successfully tracked and recovered the 3D hand pose in a real time fashion.

---

**Keywords:** 3D hand-pose estimation, Random Forest algorithm, Depth map, Kinect Sensor

---

A preliminary version of this paper appeared in ICONI 2014, Taipei, Taiwan and was selected as an outstanding paper. This version includes a concrete analysis and supporting implementation results on the proposed approach. This research was supported by Kyonggi University research grant 2014.

## 1. Introduction

**H**and pose estimation is an important issue in Human Computer Interaction and its application. Recently, Kinect has been used to achieve real-time body tracking capabilities, which has triggered a new era of natural interface based applications. In this paper we introduce a new method to estimate hand pose using Haar-like features and Random Forest along with single depth image obtained by the Kinect sensor [1]. Random Forest classifier [2] is known to be an effective learning classifier for handling mass storage database because it has higher recognition performance compared to other classifiers and yet at the same time it shows very high calculation speed.

In this work, we first created a database which is necessary to make the hand parts classifier learn and materialized a virtual 3D hand model designed with 23 degree of freedom and hierarchical structure composed of each part of the hand [3]. Through this process, various hand pose data are produced in massive volume. In order to exclude as many unpractical poses and infrequently used poses as possible, this work referred to ASL (American Sign Language) system and created maximum possible poses that are meaningful. Each pose in the database is created by this procedure and is composed of a pair of labeling image and depth image classified based on parts of finger joint and palm is used for building Random Forest classifier. The input depth hand image from the Kinect sensor eventually becomes a labeled hand region through the Random Forest classifier. Furthermore, the efficiency of the proposed method is evaluated by comparing the performance of two types of Random Forest classifier where one uses the conventional pixel previously suggested by Keskin [4] and the other used the Haar-like feature vector suggested in this paper. In the experiment the proposed method showed higher recognition rates and a performance of 20-30 fps confirming its practical use in classifying hand area in real-time fashion.

In Section 2, we review the previous works in 3D hand pose estimation. Section 3 addresses the detailed descriptions of the proposed Haar-like based Random Forest classifier using the Kinect sensor. Section 4 presents the experimental evaluation of classification accuracy by comparing the proposed method with current state of art method using per-pixel classification with RDF. Finally, section 5 includes conclusion and future works.

## 2. Related Work

Recently introduced real-time human body pose simulation technique by Microsoft enhanced the users' attention to the area of hand pose estimation for the applications of human computer interactions and augmented reality because the accurate and efficient hand pose estimation is essential to such higher level tasks [1]. Human body pose estimation techniques regarded as state-of-the-art are data-driven and bottom up approaches in which pixels are independently assigned body part label [5] or a vote for joint locations [6,7,8].

Compared to the techniques of human body estimation, the hand pose estimation techniques are more difficult to develop since the hand has more complex articulations, self-occlusions and multiple view points. Therefore, they require exponentially more data sets to capture the variations and making their application difficult. Moreover, without global regression procedures such as enforcing dependency between local outputs [9] or kinematic constraints [10], obtaining the accurate hand pose estimation cannot be guaranteed in bottom

up approaches. Meanwhile, other approaches are more top-down and global in which hypotheses are generated from a 3D hand model and poses are tracked by fitting the model to the test dataset [11,12].

A number of approaches have been introduced to estimate the hand pose from depth images [13,14,15,18]. Recently by adopting the idea of an intermediate representation for the tracked object based on the approach [5], a depth image based real-time skeleton fitting algorithm for the using an object recognition by part approach [4] was reported efficient for hand pose estimation. However, the overall accuracy of the developed system largely depends on the individual random decision tree and the training dataset. In this work, we follow the approaches of [4,5] while improving the accuracy of the classification ratio of hand parts of the 3D hand model in the training phase of the overall 3D hand pose estimation.

### 3. Proposed Hand Pose Estimation Technique

#### 3.1 Overall Procedure for Hand Pose Estimation

The overall process for estimating the pose of 3D hand in real time is as follows. From the image captured by Kinect a depth image is obtained. After segmenting articulations of the hand, hand pose is estimated by using the Random Forest algorithm. Fig. 1 shows the overall steps for hand pose estimation.

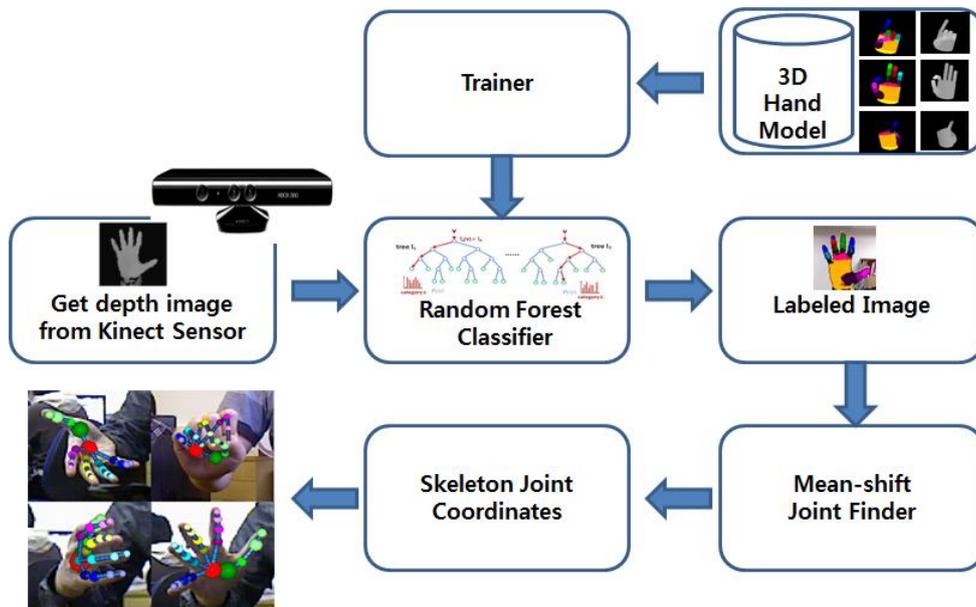
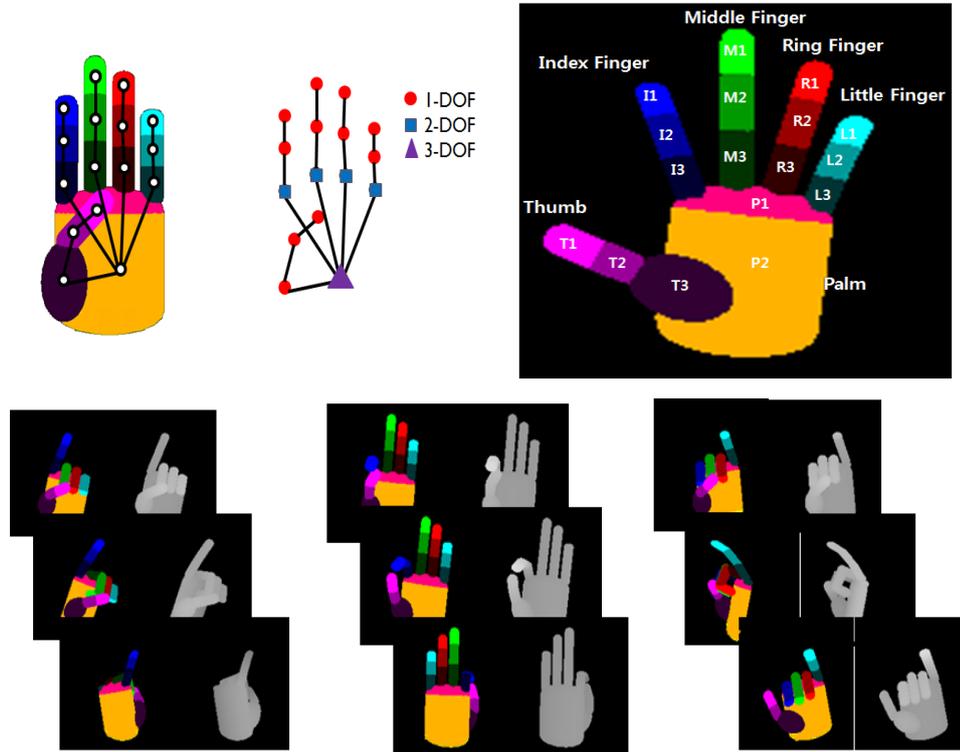


Fig. 1. Overall steps for hand pose estimation

#### 3.2 3D Hand Model for Database

In order to make a classifier to estimate a hand pose, a large numbers of hand pose images are necessary. Therefore, we develop 3D hand models which are 23-DOF (degree of freedom). The 3D hand model consists of 15 different joints and labeled by different colors as illustrated in Fig. 2. The hand model consists of 17 different sub-regions such as 2 palm regions ( $P1, P2$ ) and 15 regions of each finger such as thumb ( $T1, T2, T3$ ), index finger ( $I1, I2, I3$ ), middle

finger ( $M1, M2, M3$ ), ring finger ( $R1, R2, R3$ ) and little finger ( $L1, L2, L3$ ), respectively. Based on this 3D model, various hand poses can be constructed by randomly generating the angles of joints. However, unnatural or implausible hand poses should be excluded from the database in order to improve the hand pose recognition ratio. For this purpose, we exploit ASL (American Sign Language) excluding unnecessary cases of hand pose models.



**Fig. 2.** A developed 3D Hand model with 17 classified sub-regions of the hand and labelled hand poses with their corresponding depth images

### 3.3 Haar-like Feature Selection

Selection of efficient feature vectors of the human hand is important because it is necessary to segment the joints of hand in real time. Extracting feature values is based on the following equation [5].

$$f(I, x) = d_I \left( x + \frac{u}{I(x)} \right) - d_I \left( x + \frac{v}{I(x)} \right) \quad (1)$$

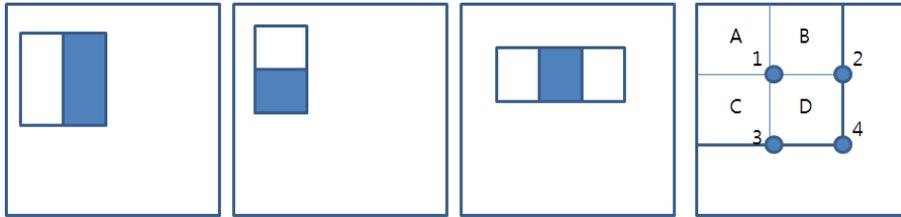
In the formula (1), for a pixel at  $x$  and its neighboring pixels  $u, v$  in an depth image  $I$ , where  $d_I(c)$  gives the depth of the pixel at  $c$ , a feature  $f(I, x)$  is simply the difference value of the two pixels. Thus, the evaluation process for obtaining features becomes very fast. In this paper we use Haar-like features instead of the simple difference value of two pixels in feature selection. A Haar-like feature takes into account adjacent rectangular regions at a specific location in a window, sums up the pixel intensities in each region and calculates the difference between these sums. The difference is subsequently used to categorize the subsections of an

image.

The value of a simple rectangle's Haar-like feature is defined as the difference in the sum of the pixels of areas inside the rectangle, which can be at any position and scale within the detected face image. This rectangle features can be computed very rapidly by using summed area table called integral image. **Fig. 3** illustrates examples of two-rectangle features and three-rectangle features, respectively. The integral sum ( $I_{sum}$ ) of location  $l(x, y)$  can be the sum of the pixels above and the left of  $l(x, y)$  and can be expressed by

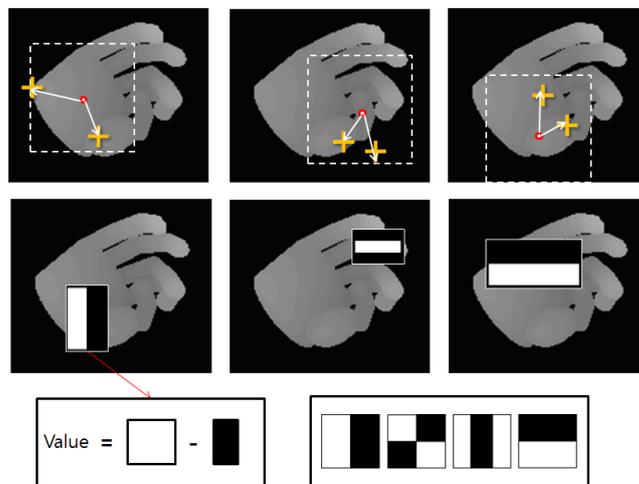
$$I_{sum}(x, y) = \sum_{x' \leq x, y' \leq y} I(x', y') \tag{2}$$

The sum of pixels in the rectangle  $D$  can be computed as  $4 + 1 - (2 + 3)$ . The calculated feature values are used for Random Forest classifier to recognize hand fingers.



**Fig. 3.** Examples of rectangle features (2-rectangle features and 3-rectangle features) and rectangular sum using integral image

Eventhough the Haar-like feature selection uses all pixels values in a selected region, computational burden is minimized because integral image is used in evaluating feature value. **Fig. 4** shows the comparison between using two pixel conventionally used in the previous method and Haar-like feature in feature selection in the proposed method.



**Fig. 4.** Two different feature selection methods by using conventional two pixels (top) and the proposed Haar-like feature (bottom)

### 3.4 Hand Labeling by Random Forest Algorithm

Random Forest algorithm with the images of various poses by 3D hand model and feature vectors can classify the hand region. Random forests are an ensemble learning method for classification that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes produced by individual trees. The algorithm for inducing a random forest was developed by Leo Breiman [2]. Since Random Forest has the advantages of high reliability and low computational burden it is frequently used for human pose estimation[5] or face recognition[16].

In the original bagging algorithm, the training algorithm applies the general technique of bagging or bootstrap aggregating to the tree learners. Given a training set  $X = \{x_1 \cdots x_n\}$  with the responses  $Y = \{y_1 \cdots y_n\}$ , bagging repeatedly selects a bootstrap sample of training set and fits trees to these samples. Given a free parameter  $B$ , for  $b = 1$  through  $B$ :

- ① Sample, with replacement,  $n$  training samples  $X_b, Y_b$  from  $X, Y$ .
- ② Train a decision tree  $f_b$  from  $X_b, Y_b$ .

After training, predictions for unseen examples  $x'$  can be made by taking the majority vote in the case of decision trees:

$$\hat{f} = \frac{1}{B} \sum_{b=1}^B \hat{f}_b(x') \quad (3)$$

Generally, depending on the size and training set several thousand trees are used in the process. When the number of trees is increased the variance of the model tends to decreased without increasing the bias. As a result, the training and test error tend to level off after some number of trees have been fit.

Random forests differ only in one way from the described original bagging scheme. Random forests use a modified tree learning algorithm that selects a random subset of the features at each candidate split in the learning process. Because of the correlation of the trees in an ordinary bootstrap sample: if one or a few features are very strong predictors for the response variable (target output), these features will be selected in many of the  $B$  trees, causing them to become correlated. Fig. 5 shows the Random Forest composed of multiple decision trees.

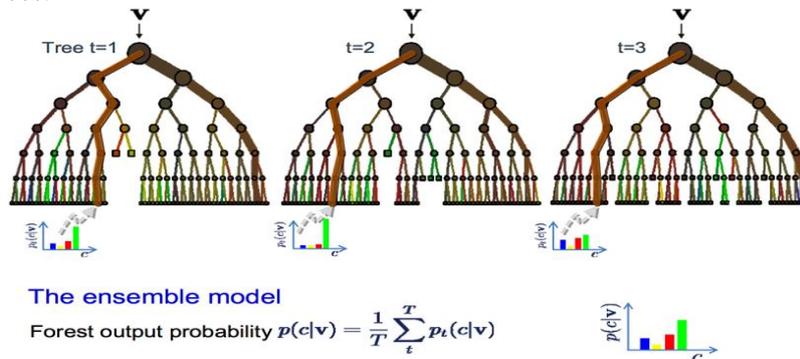


Fig. 5. A Random Forest composed of multiple decision trees and its ensemble model

The nodes which are generated by a set of arbitrary feature vectors and a set of images from 3D hand model consist of random decision tree as shown in Fig. 6. Several trees which undertake the learning process are combined to produce better results. When all learning process is over, the pixels of depth image are tested through each random decision tree and the joint class where the pixels belong to is determined. By connecting the central value of each joint class, we can estimate hand pose.

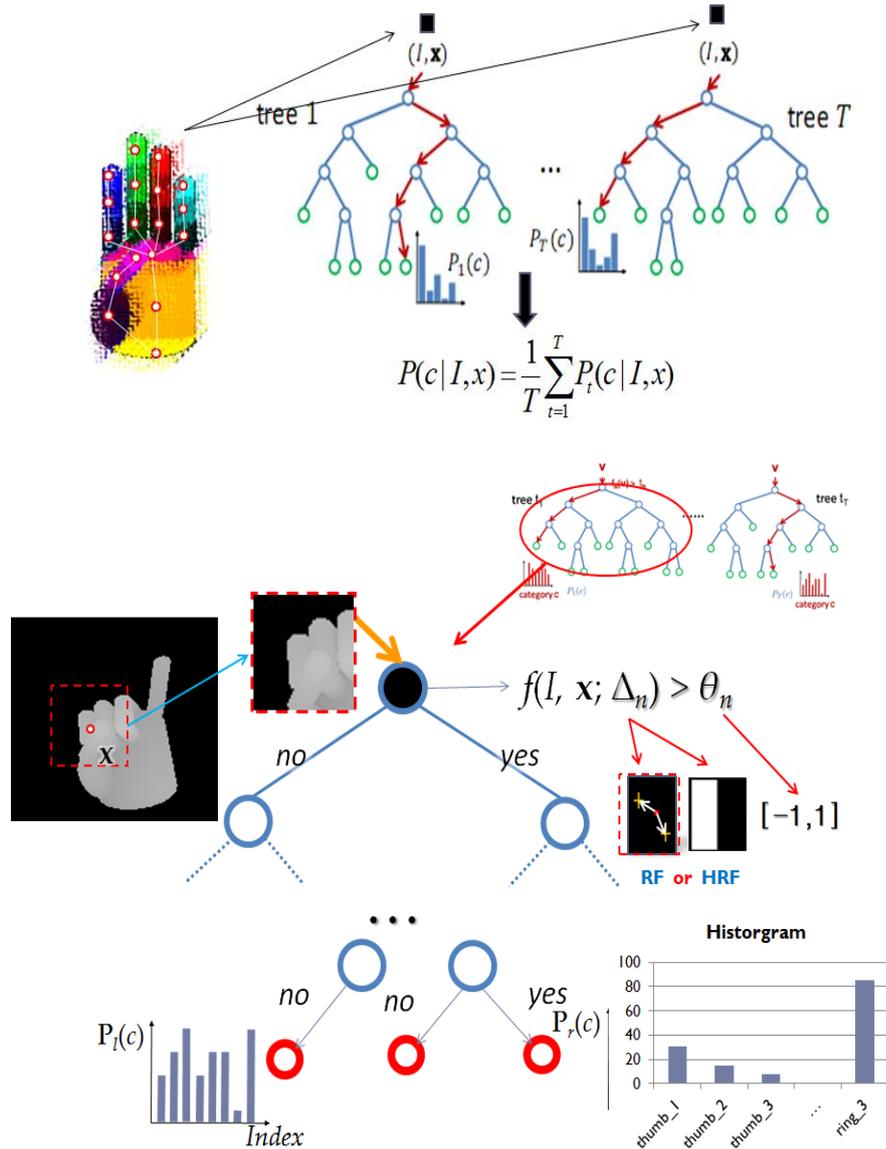


Fig. 6. The procedure of hand region classification by using RDF and HRDF

In the previous research on estimating hand pose using Random Forest(RDF), two pixel values are used as a feature vector in order to classify the pixels of the depth image. However, in this work, Haar-like feature is used for Harr Random Forest classifier (HRDF) instead of a pair of feature values as illustrated in Fig. 7. As described in section 3.3 the main difference

between the conventional RDF and HRDF is in the selection of feature values from the hand region for the classification. Instead of selecting a single pair of pixels and using the difference value of the pixels in the RDF, all pixel values within the rectangle area are evaluated in HRDF. Even though HRDF considers all pixels in the rectangle boundary, however, the feature value  $f(I, x; \Delta_n)$  of HRDF is computed in a constant time by following equation

$$f(I, x; \Delta_n) = \{i(B) - i(C) - i(F) + i(E)\} - \{i(F) - i(E) - i(D) + i(A)\} \quad (4)$$

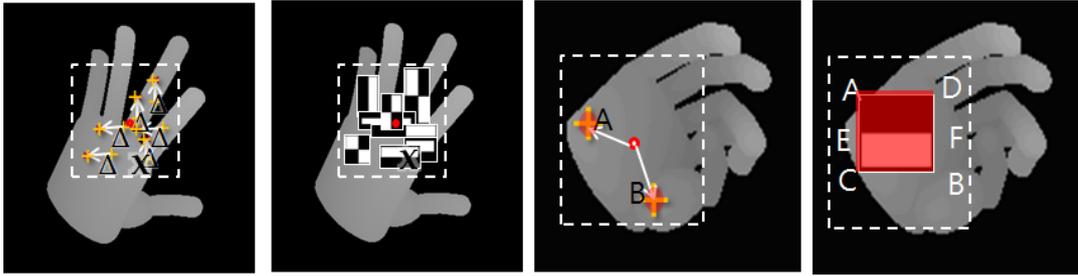


Fig. 7. Feature selection method for both RDF and HRDF

Depending on the comparison result between the feature values  $f(I, x; \Delta_n)$  based on the Harr-like feature and threshold value within  $[-1, 1]$ , the direction of the traversing path on the branch nodes is determined. When the learning process reaches leaf nodes, the probabilities of 17 classes composed of fingers and palm to be labelled are evaluated. The probability of the all decision trees becomes as follows:

$$P(c_i | I, x) = \frac{1}{T} \sum_{t=1}^T P_t(c_i | I, x) \quad (5)$$

In formula (5),  $T$  denotes the numbers of decision trees in the forest,  $c_i$  means the class,  $P_t(c_i | I, x)$  the probability histogram of  $t^{\text{th}}$  decision tree, and  $P(c_i | I, x)$  the accumulated probability histogram from the all decision trees. Depending on the accumulated probability value, input feature of the hand region is labelled to a specific class of the hand which has the highest probability.

### 3.5 Joint Position Estimation by Mean Shift

When each pixel of the depth image is assigned posterior probabilities, these probabilities are used to estimate the joint positions of the hand. In this process the outliers caused by false positives can mislead the centroid of the pixel locations of a particular hand part. Thus, in order to reduce such effect we adopt the mean shift algorithm. The mean shift is known as a procedure for locating the maxima of a density function given discrete data sampled from that function [17].

It is an iterative method which starts with an initial estimate  $x$ . When a kernel function  $K(x_i - x)$  be given, this function determines the weight of nearby points for re-estimating the

mean. It estimates the probability density of each class with weighted Gaussian kernel placed on each sample. Each weight is set to be the pixel's posterior probability  $P(c_i|I, x)$  corresponding to the class  $c_i$ , times the square of the depth of the pixel, which is an estimate of the region the pixel covers. Typical Gaussian kernel on the distance to the current estimate is  $K(x_i - x) = e^{-c\|x_i - x\|^2}$ . The weighted mean of the density in the window determined by  $K$  can be expressed by

$$m(x) = \frac{\sum_{x_i \in N(x)} K(x_i - x) x_i}{\sum_{x_i \in N(x)} K(x_i - x)} \quad (6)$$

where  $N(x)$  is the neighborhood of  $x$  and  $K(x) \neq 0$ . The mean shift sets  $x \leftarrow m(x)$ , and repeats the estimation until  $m(x)$  converges. Consequently, the estimated joint locations are used to find a match for the skeleton of the input hand image.

#### 4. Experimental Classification Results and Analysis

For the evaluation of the proposed hand pose estimation system, we first obtain 4,000 hand pose depth images which are a size of 160 by 160 pixels and then construct corresponding 3D hand pose models by OpenGL in the database as shown in Fig. 8.

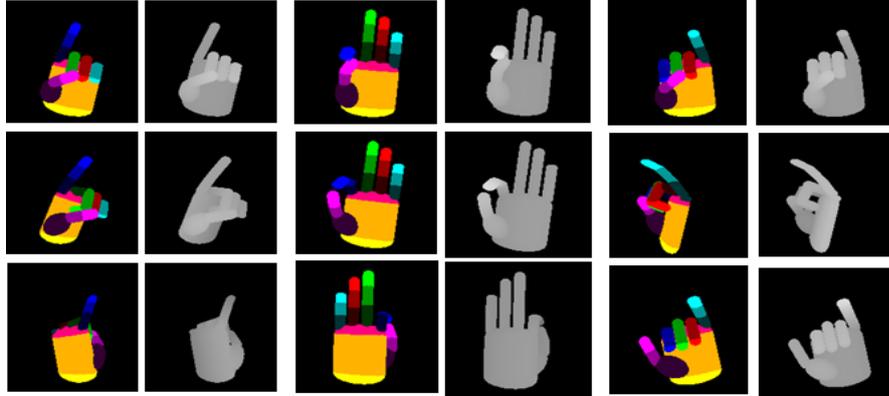
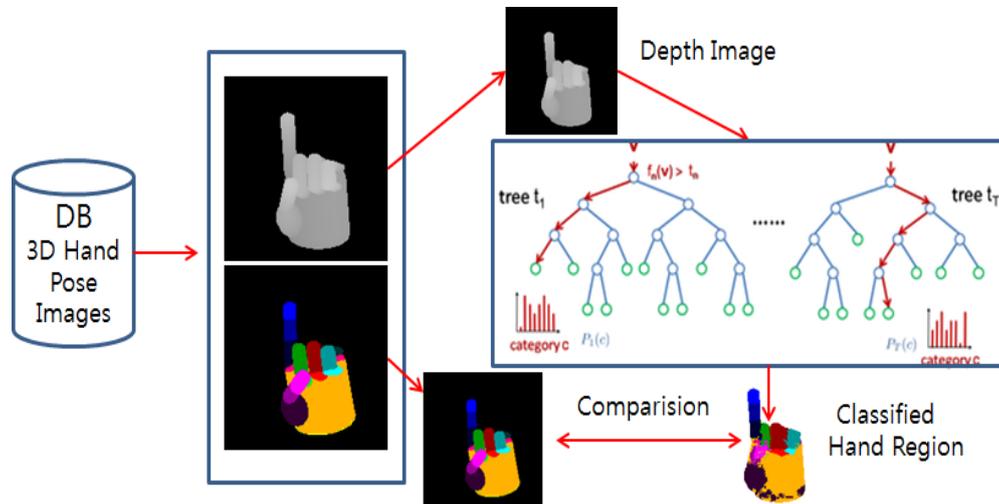


Fig. 8. 3D hand pose models classified and their corresponding depth images

From the 4,000 hand images, we randomly selected both a 400,000 and a 1,600,000 pixel samples for a random forest composed of 20 decision trees in order to learn the Random Forest classifier. The efficiency of the proposed approach is evaluated by comparing the classification ratio of two types of Random Forest classifier. One is RDF proposed by Keskin [4] which uses a pair of pixels as feature vector and the other is the proposed HRDF which utilizes the Haar-like feature. The evaluation process of the proposed method is as illustrated in Fig. 9. As for the performance evaluation, we chose 100 hand poses from the database which contains the hand models and assign the corresponding depth image for each selected hand pose to both Random Forest and Haar-like Random Forest classifiers. The overall classification rate is evaluated by averaging the classification results of 100 sample images.



**Fig. 9.** The evaluation process of the proposed approach

The developing environments of the proposed method are shown in **Table 1**. In order to process Kinect sensor OpenNI libraries are used while 3D hand models are developed by using OpenGL.

**Table 1.** Developing environmanets

| S/W             |                        | H/W                 |                       |
|-----------------|------------------------|---------------------|-----------------------|
| <b>OS</b>       | Windows 7 64bit        | <b>CPU</b>          | Intel i5-3570 3.40GHz |
| <b>Language</b> | C / C++                | <b>RAM</b>          | 8.00GB                |
| <b>Library</b>  | OpenNI, OpenCV, OpenGL | <b>Graphic Card</b> | GeForce GTX 650       |
| <b>IDE</b>      | Visual Studio 2010     | <b>Image Sensor</b> | Kinect                |

**Fig. 10** provides classification results of the hand region based on conventional RDF and the proposed HRDF. As a result of the classification, 17 different pseudo colors are assigned to the corresponding sub-regions (Thumb, Index finger, Middle finger, Ring finger, Little finger and Palm) of the hand as classified in **Fig. 2**. The two graphs in **Fig. 10** show the comparative accuracy rate of the classification based on both the previous method and the proposed HRDF. In both 400,000 and 1,600,000 pixel samples cases, the proposed method shows higher recognition rate in 12 classes out of 17 classes.

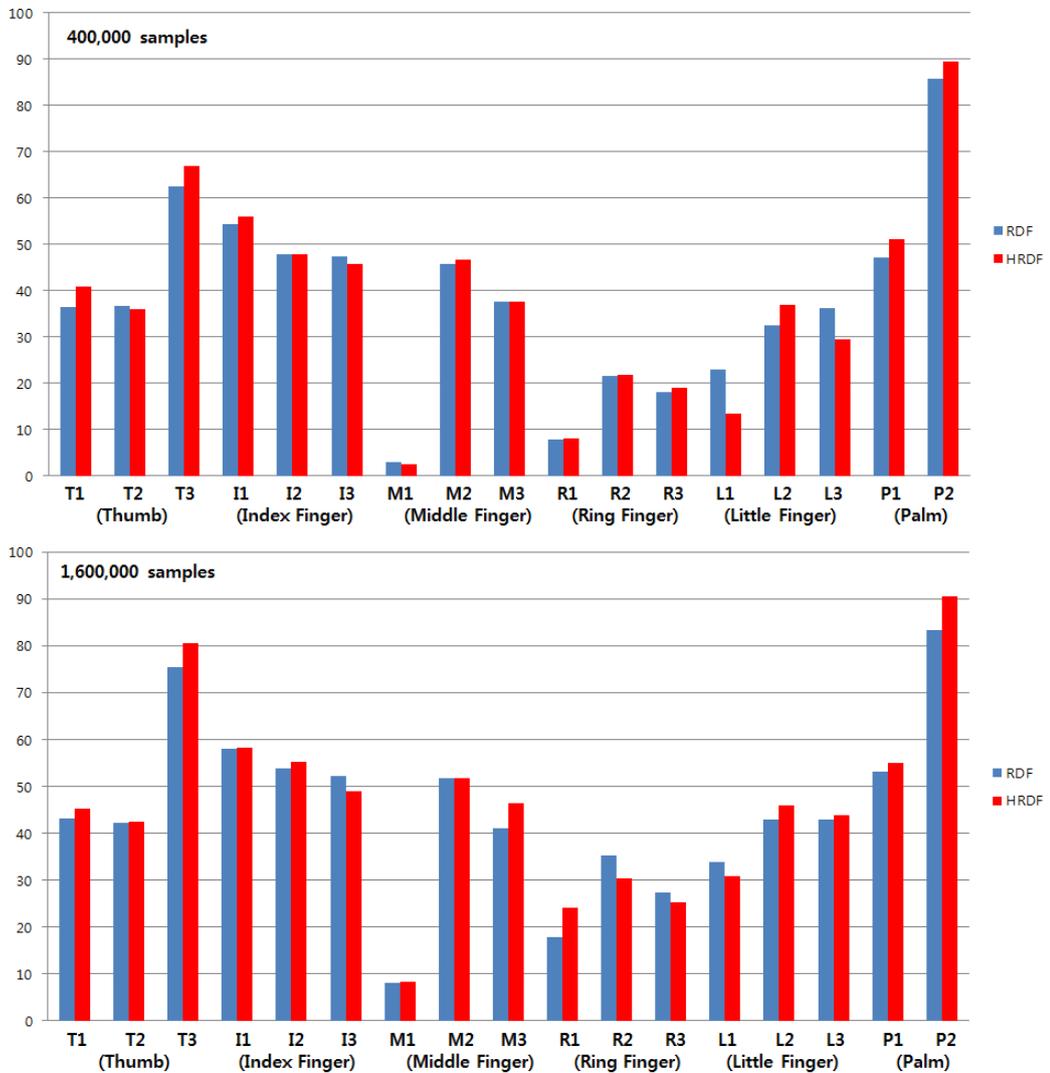
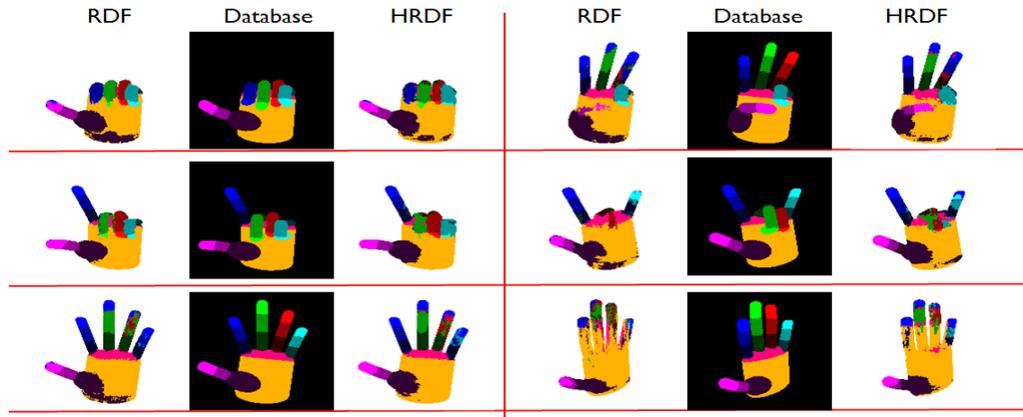


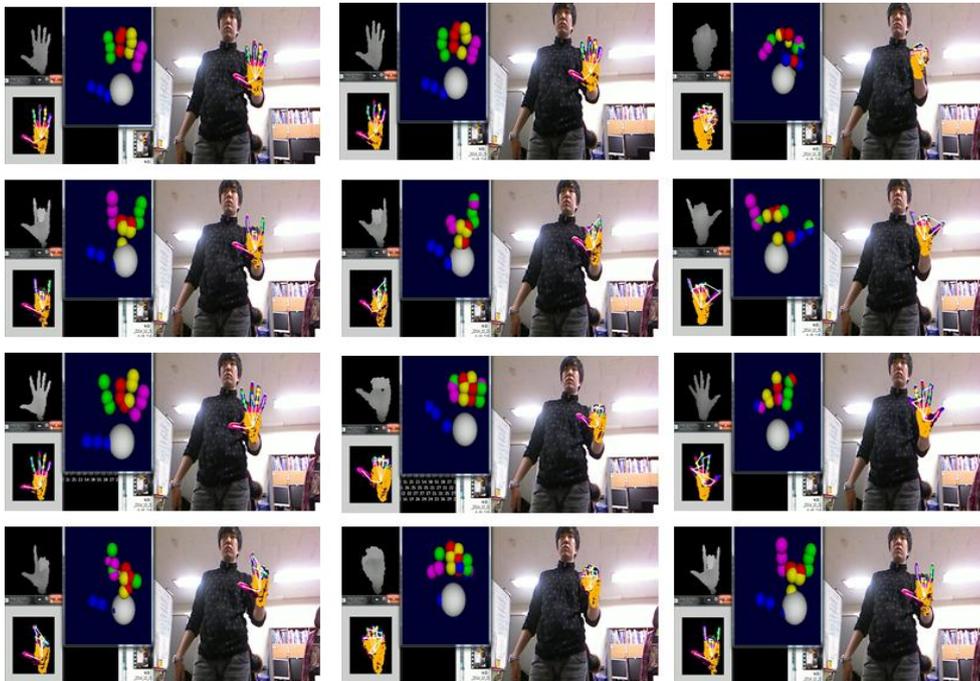
Fig. 10. Labelled results of hand region based on conventional RDF and the proposed Haar-like feature based RDF(HRDF) and classification accuracy of 17 hand regions of 400,000 samples and 1,600,000 samples

**Fig. 11** shows real-time classification result of the hand region using depth images from the Kinect sensor based on the proposed HRDF. The labelling using HRDF show high performance in terms of calculation speed and recognition rate with just depth images.



**Fig. 11.** Real-time hand region classification of depth images from the Kinect sensor

**Fig. 12** shows real-time 3D hand pose estimation by fitting of hand skeleton from input images from the Kinect sensor.



**Fig. 12.** Real-time 3D hand pose estimation by fitting of hand skeleton from input images from the Kinect sensor

We have tested for two cases of samples: 400,000 and 1,600,000 using both regular pixel feature selection and Haar-like feature selection in applying Random Forest to hand pose estimation. In both cases, Random Forests are composed of 20 trees. The experimental results are shown in Fig. 13. When 400,000 samples were used, the average classification rate and variance of RDF are 48.59% and 39.81 meanwhile those of proposed HFDF are 49.05% and 24.69 respectively. The results mean eventhough the small amount of recognition ratio is increased however the stability of the system is increased. When 1,600,000 samples are used, the average classification rate and variance of RDF are 48.86% and 69.90. Meanwhile, those of HFDF are 55.23% and 21.12 respectively. This shows that the recognition ratio of the classifier has increased by 6.83% and the classifier is more stable in both the performance and the learning process. Through the experiential cases we can prove that the proposed Random Forest with Haar-like feature shows improved efficiency in hand pose estimation. In the point of view of the decision tree, both RDF and HRDF show high recongition ratio when the number of decision tree is 14 in the first case. In second case, using 18 decision trees for RDF and 9 decision trees results in high accuracy.

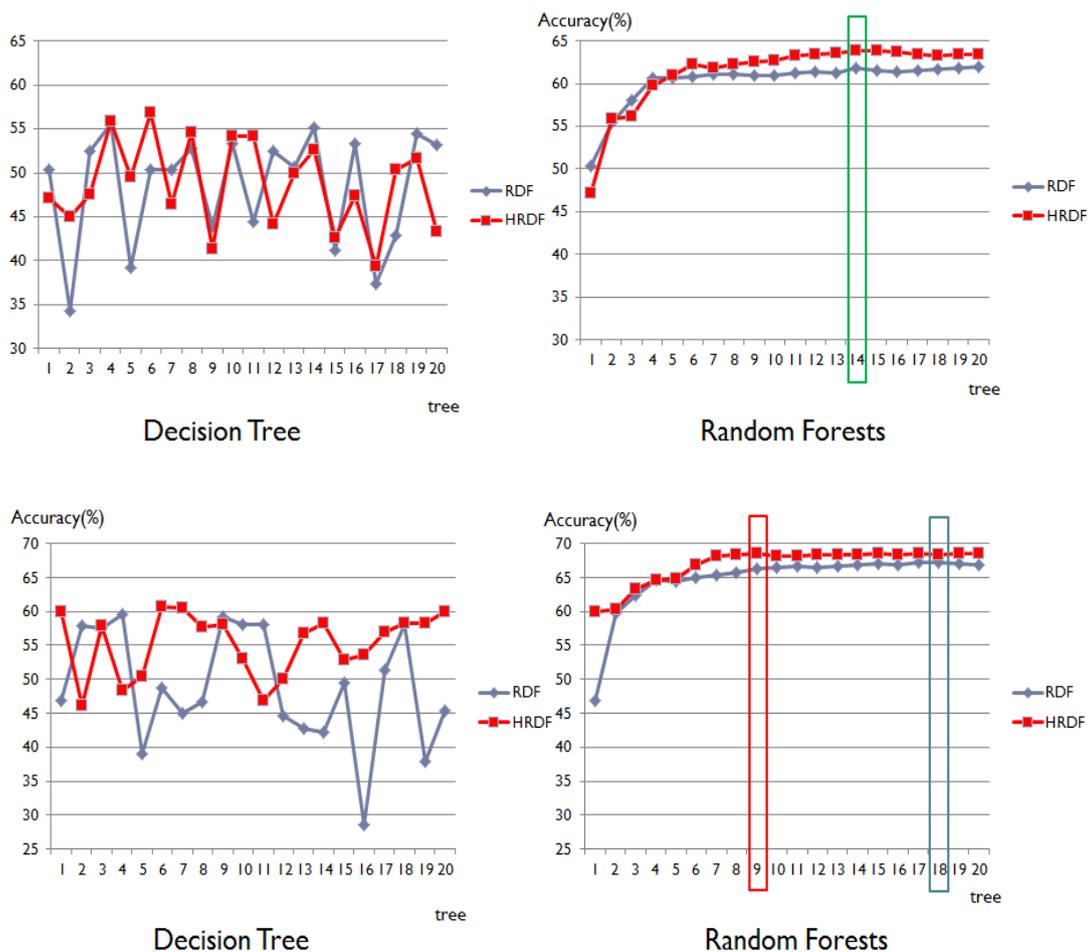


Fig. 13. The comparative classification results of hand pose both using RDF and HRDF (top: 400,000 samples, bottom: 1,600,000 samples)

## 5. Conclusion

In this paper, we present a novel approach for tracking and recovering the 3D orientation of human hand using the Kinect sensor. The main idea of the proposed method is to solve an optimization problem that minimizes the discrepancy in the 3D shape between an observed actual hand by Kinect and a hypothesized 3D hand model. In this work, we utilize Haar-like feature rather than using conventional two pixel values for feature selection and apply the selected features to Random Forest algorithm on training the synthetic depth images generated by animating the developed 3D hand model. We can prove from the experiments that the proposed approach showed higher hand part classification rates than when per-pixel classification with Random Forest was used. In a practical point of view, the proposed system guarantees the performance of 20-30 fps in hand pose estimation of real-time application.

## References

- [1] Microsoft Corp. Redmond WA. Kinect for xbox 360.
- [2] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no.1, pp. 5-32, 2001. [Article\(Cross Ref Link\)](#)
- [3] I. Oikonomidis, N. Kyriazis, and A. Argyros, "Markerless and efficient 26-DOF hand pose recovery," in *Proc. of 10<sup>th</sup> Asian conference on Computer Vision-Vol. Part III*, pp. 744-757, 2011. [Article\(CrossRef Link\)](#)
- [4] C. Keskin, F. Kirac, Y. Kara, L. Akarun, "Real Time Hand Pose Estimation using Depth Sensors," in *Proc. of 13<sup>th</sup> IEEE International Conference on Computer Vision, ICCV 2011*, pp. 1228-1234, 2011. [Article\(CrossRef Link\)](#)
- [5] J. Shotton, A. Fitzgibbon, M. Cook et al, "Real-time human pose recognition in parts from single depth image," *Communication of the ACM*, vol. 56, no. 1, pp. 116-124, 2013. [Article\(CrossRef Link\)](#)
- [6] R. Girshick, J. Shotton, P. Kohli, A. Criminisi, and A. Fitzgibbon, "Efficient regression of general-activity human pose from depth images," in *13<sup>th</sup> Proc. of International Conference on Computer Vision, ICCV 2011*, pp. 415-422, 2011. [Article\(CrossRef Link\)](#)
- [7] G. Pons-Moll, J. Taylor, J. Shotton, A. Hertzmann, and A. Fitzgibbon, "Metric regression forest for human pose estimation," in *Proc. of 24<sup>th</sup> British Machine Vision Conference, BMVC 2013*, pp. 4.1-4.11, 2013. [Article\(CrossRef Link\)](#)
- [8] J. Taylor, J. Shotton, T. Sharp, and A. Fitzgibbon, "The vitruvian manifold: Inferring dense correspondence for one-shot human pose estimation," in *Proc. of 12<sup>th</sup> IEEE International Conference on Computer Vision and Pattern Recognition, CVPR 2012*, pp. 103-110, 2012. [Article\(CrossRef Link\)](#)
- [9] M. Sun, P. Kohil, and J. Sotton, "Conditional regression forest for human pose estimation," in *Proc. of 12<sup>th</sup> IEEE International Conference on Computer Vision and pattern Recognition, CVPR 2012*, pp. 3394-3401, 2012. [Article\(CrossRef Link\)](#)
- [10] T. Dang, T.-H. Yu and T.-K. Kim, "Real-time articulated hand pose estimation using semi-supervised transductive regression forests," in *Proc. of 15<sup>th</sup> IEEE International Conference on Computer Vision, ICCV 2013*, pp. 3225-3231, 2013. [Article\(CrossRef Link\)](#)
- [11] L. Ballan, A. Taneja, J. Gall et al, "Motion capture of hands in action using discriminative salient points," in *Proc. of 12<sup>th</sup> European Conference on Computer Vision, ECCV 2012, Lecture Notes in Computer Science 7577*, pp. 640-653, 2012. [Article\(CrossRef Link\)](#)
- [12] M. de La Gorce, D. Fleet, and N. Paragios, "Model-based 2D hand pose estimation from monocular video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 9, pp. 1793-1805, 2011. [Article\(CrossRef Link\)](#)
- [13] S. Malassiotis and M. Srinatzis, "Real-time hand posture recognition using range data," *Image and Vision Computing*, vol. 26, no. 7, pp. 1027-1-37, 2008. [Article\(CrossRef Link\)](#)
- [14] Z. Mo and U. Neumann, "Real-time Hand Pose Recognition Using Low-Resolution Depth

- Images,” in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2006*, vol. 2 pp. 1499-1505, 2006. [Article\(CrossRef Link\)](#)
- [15] A. Erol, G. Bebis, M. Nicolescu et al, “Vision-based hand pose estimation: A review,” *Computer Vision and Image Understanding*, vol. 108, pp. 52-73, 2007. [Article\(CrossRef Link\)](#)
- [16] Fanelli G., Gall J. and Van Gool L., “Real time head pose estimation with random regression forests,” in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011*, pp. 617-624, 2011. [Article\(CrossRef Link\)](#)
- [17] Cheng, Yizong, “Mean Shift, Mode Seeking, and Clustering,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 8, pp. 790–799, 1995. [Article\(CrossRef Link\)](#)
- [18] Wenkai Xu and Eung-Joo Lee, “A Novel Method for Hand Posture Recognition Based on Depth Information Descriptor,” *KSII Transaction on Internet and Information Systems*, vol. 9, no. 2, pp. 763-774, 2015. [Article\(CrossRef Link\)](#)



**Wonggi Kim** received MS degree from Kyonggi University South Korea majoring computer science. He also received BS degree from Kyonggi University majoring mathematics and computer science in 2013. His major research interests are vision-based face/hand pose estimation, and augmented reality.



**Junchul Chun** is currently a professor in the department of computer science and a principal investigator in the Graphics and Image Processing Research Laboratory at Kyonggi University, Suwon South Korea. He received a B.S. degree from Chung-Ang University majoring in computer science. He also received M.S. and Ph.D. degrees from the department of computer science and engineering at the University of Connecticut, CT., U.S.A, in 1992 and 1995, respectively. He was a visiting research scholar in PRIP (Pattern Recognition and Image Processing) at Michigan State University and University of Colorado at Boulder, 2001 and 2009 respectively. He is currently a chief director of CCSRC (Contents Convergence Software Research Center) at Kyonggi University. He has been currently served as the present of the Korean Society of Internet Information since January 2015. His major research fields are augmented reality, vision-based animation, and human computer interaction.