Human Action Recognition using Depth Maps

Vennila Megavannan
IPG R&D Hub,
Hewlett Packard, Bangalore, India
Email: vennila.vyoma@gmail.com

Bhuvnesh Agarwal
Computer Science and Engineering
IIT-Kharagpur, India.
Email: bhuvnesh5261@gmail.com

R. Venkatesh Babu
SERC, Indian Institute of Science
Bangalore, 560012, India
Email: venky@serc.iisc.ernet.in

***Abstract*—In this paper we propose an approach to recognize human actions using depth images. Here, we capture the motion dynamics of the object from the depth difference image and average depth image. The features from the space-time depth difference images are obtained from hierarchical division of the silhouette bounding box. We also make use of motion history images to represent the temporal information about the action. We make use of the translation, scale and orientation invariant Hu moments to represent the features of the motion history image and the average depth image. We then classify human actions using support vector machines. We analyze the representation efficiency of Hu moments and the hierarchical division of bounding boxes separately in order to evaluate the contribution of each of the features. The results show superior performance of over 90% when both features are combined.**

***Index Terms*—Action recognition, Kinect, Depth sensor, Motion History Image**

I. Introduction

In recent years, human actions recognition has been a major concern in computer vision because of its immense application in the field of autonomous video surveillance, video retrieval and human computer interaction which require methods for recognizing human actions in various scenarios.

There have been several methods proposed in the past two decades for recognizing human actions from single and multiple views. In [1], they represent actions using MHI and MEI in different views. The camera is located at 7 different angles to capture actions in different views. Yamato et al. [2] used grid-based silhouette mesh features to form a compact codebook of observations for representing actions using hidden markov model. Extracting silhouettes in real-life videos is challenging and prone to noise. The noisy silhouettes are handled by phase correlation [3], by constructing space-time volume over silhouette images [4], [5]. Optical flow is another major feature used for recognizing actions. Efros et al. [6] calculate optical flow in person-centered images in order to model relative motions among different locations of object. Babu et al., [7] utilized the readily available motion vectors from the compressed video stream for recognizing actions. Shotton et al., [8] have proposed a new method to predict 3D positions of body joints from a single depth image enabling real-time human pose recognition. Poppe [9] and Weinland [10] have presented detailed survey on human action recognition approaches.

The existing benchmark data sets are KTH [11], Weizmann [4] and IXMAS [12]. KTH and Weizmann datasets are captured from a single camera and IXMAS dataset is created using using 5 calibrated and synchronised cameras to capture actions in multiple views. But today with the advancement of camera and video technology, we are able to capture the depth image which gives as information in the 3rd dimension with which we can represent and recognize actions more accurately with lesser computational burden. Hence we have made use of the depth camera to create our database for human action recognition.

In this paper, we propose a method of human action recognition which uses depth images from Kinect. The advantage of this approach is that we are able to recognize actions which have similar RGB images like swim and side wave. i.e the actions which are performed perpendicular to the camera and the actions which are performed parallel to the camera can be distinguished with the help of the depth image which gives us the motion information perpendicular to the camera.

We need to represent the spatial and temporal information about an action in a single image. In order to do this we make use of Motion History Images [1] where the pixel intensity is function of the recency of action. In order to integrate the depth information generated by the Kinect, we have explored the representative power of the average depth image where we take the average of the non zero depth values across N frames for each pixel location.This represents the average depth information for N frames. Hu moments [13] are used as descriptors of the motion history image and average depth image. To distinguish between actions more accurately, we have defined the depth difference image where capture the change in depth for N frames at each pixel location. We then make use of the hierarchical windows of varying sizes to form the feature vector representing the depth difference image to capture the change in depth information of specific spatial regions of the action as well as the whole action effectively.

The paper is organized as follows. Section 2 gives the overview of the proposed approach. Depth normalization and silhouette extraction are explained in section 3. Section 4 details action representation using depth information. Section 5 explains feature extraction and results are discussed in section 6. Section 7 concludes the paper.

II. System Overview

The overview of the proposed approach is illustrated in Fig. 1. In order to represent the variation in depth for the various actions more effectively, we perform depth normalization. We

978-1-4673-2014-6/12/$31.00 ©2012 IEEE

find the maximum and minimum value of depth within the silhouette of the person. We then scale this range i.e. the difference between the maximum and minimum depth value into a 8 bit number which thereby provides us with the finer details of depth in the desired region. The next step is the silhouette extraction. We then extract the silhouette of the person. This is done by finding a suitable threshold depth value for all the subjects above which we classify all the pixels as background. The next step is action representation where we represent the various actions using motion history image, average depth image and the depth difference image. The next step is feature extraction. We represent the features using hu moments and hierarchical bounding box. These features are used to train a model using support vector machine and classify the actions.

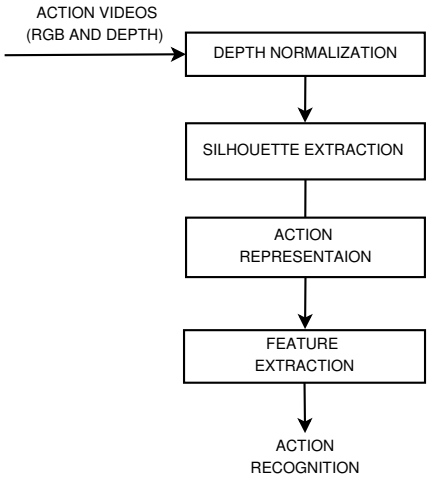


Fig. 1. overview of our approach

III. Preprocessing

A. Depth Normalization

The depth information provided by Kinect is in 11 bits but processing all the images in 11 bits will be computationally too slow. Hence, we have to scale them down to some lower bit numbers. But the problem in doing so is that we lose lot of depth information of desired region. To overcome both the

(a) (b)

Fig. 2. (a) Original depth image (b)Normalized depth image

problems, we found two flexible threshold values in which the all the actions can be described completely for all users. The depth information of the far background or the very close region to camera does not have any contribution in recognising the actions. Now we scale down the depth values in this region to an 8 bit number which thereby provides us with the finer details of depth in the desired region and the processing overload is also reduced. Figure 2 shows the original depth image and the corresponding normalized depth image.

B. Silhouette Extraction

Detection and elimination of the background using only 2-Dimensional(RGB) image is very difficult and inefficient. But with the help of the depth image, the background can be easily identified and removed in our case. We make use of the fact that the subject is always at a particular distance from the background pixels. We find a suitable threshold depth value for all the subjects above which we classify all the pixels as background. Hence we easily get the silhouette of the depth image.

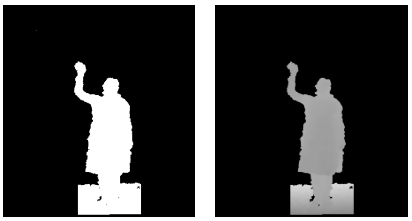
$$D(i,j,t) \quad = \quad \begin{cases} D'(i,j,t), & \text{if } D'(i,j,t) <= \zeta, \\ 0, & \text{otherwise,} \end{cases} \tag{1}$$

where i,j denotes row and column of pixel the image,
t is the time stamp of the temporal frames,
ζ is the background threshold depth value
D' is the depth image,
D is the silhouette of the depth image.

$$B(i,j,t) \quad = \quad \begin{cases} 1, & \text{if } D(i,j,t) > 0, \\ 0, & \text{otherwise,} \end{cases} \tag{2}$$

where (i,j) denotes row and column of the pixel location,
t is the time stamp of the temporal frames,
mask B is the binary silhouette image
D is the silhouette of the depth image.

Figure 3 shows the extracted the binary (B) and depth (D) silhouettes for a frame.



(a) (b)

Fig. 3. (a)Binary silhouette (b) Depth silhouette

IV. Action Representation

We have represented the motion using Motion history image (MHI), average depth image and difference between maximum and minimum depth image.

A. Motion History Image

To represent how the image is moving, we form a motion history image. We consider N consecutive frames at a time to obtain the motion history image. Figure 4(a) illustrates a

motion history image for N frames of swimming action. In our experiments the value of N is set at 16.

$$B_{diff}(i,j,t) = B(i,j,t) - B(i,j,t-1) \tag{3}$$

where, B_{diff} ,denotes the binary difference image.

$$I_{mhi}(i,j,t) = \begin{cases} \tau, \text{if } B_{diff}(i,j,t) = 1, \\ \max(0, I_{mhi}(i,j,t-1) - \tau) \text{ o.w,} \end{cases} \tag{4}$$

where t varies from k to $(k+N-1)$
τ is a constant
I_{mhi} is the motion history image,

Let $I^k_{mhi} = I_{mhi}(i,j,k+N-1)$ represent the motion history image for frames k to $k+N-1$; τ is calculated as $(256/N)-1$. Since we consider 16 frames for building MHI, $\tau = 15$.

B. Average Depth Image

To capture the motion information in the 3rd dimension, i.e the Z direction, we use the average depth image. We consider N frames at a time. We calculate the average of the non zero depth values at each pixel location across N frames. This image represents the average depth image for N frames. Figure, 4(b) shows the average depth image for N frames of swimming action. Equation (5) represents the average depth image for N frames.

$$I^k_{avg} = \frac{\sum_{t=k}^{k+N-1} D(i,j,t)}{\sum_{t=k}^{k+N-1} B(i,j,t)} \tag{5}$$

where, I^k_{avg} is the average depth image for frames k to $(k+N-1)$.

C. Depth Difference Image

To represent the change in depth for the motion in N frames, we find the difference between maximum and minimum of the nonzero values of depth across N frames for each pixel location. Figure, 4(c) illustrates the depth difference image for N frames of swimming action. Here, equation (8) represents the depth difference image for N frames.

$$I^k_{max}(i,j) = \max\{D(i,j,t) : D(i,j,t) \neq 0, \\ \forall\ t \in [k \ldots (k+N-1)]\} \tag{6}$$

$$I^k_{min}(i,j) = \min\{D(i,j,t) : D(i,j,t) \neq 0, \\ \forall\ t \in [k \ldots (k+N-1)]\} \tag{7}$$

$$I^k_{diff} = I^k_{max} - I^k_{min} \tag{8}$$

where, I^k_{diff} represents the depth difference image for frames k to $(k+N-1)$.

V. Feature Extraction

A. Hu Moments

We choose to represent the motion history images and average depth images of an action using the seven translation,scale and orientation invariant Hu moments. We get seven Hu moments each from the motion history image and average depth image. Hence we form a 14-dimensional vector to represent the group of N frames.

(a) (b) (c)

Fig. 4. (a) Motion history image for N frames of swimming action (b) Average depth image for N frames of swimming action(c) Depth difference image for N frames of swimming action

B. Hierarchical division of depth difference image using silhouette bounding box

For the extraction of the features, we first find a rectangular bounding box for a user for a particular action. The bounding box is created such that it can capture the complete motion of an action in it. This bounding box is created for every action and every subject.The bounding box thus created is then hierarchically divided into 54 windows of equal size placed symmetrically with respect to the centre of the window

The maximum value of non-zero values in I_{diff} in each block is computed as feature $F1^b_k$ as given in eqn. (9).The minimum value of the non-zero values in I_{diff} in each block is computed as the second feature $F2^b_k$ as given in Eq. (10)

$$F1^b_k = \max_{(i,j)\in R_b} \{I^k_{diff}(i,j) : I^k_{diff}(i,j) \neq 0\}, \tag{9}$$

$$F2^b_k = \min_{(i,j)\in R_b} \{I^k_{diff}(i,j) : I^k_{diff}(i,j) \neq 0\}, \tag{10}$$

Here, $F1^b_k$ indicates the representative maximum change in depth corresponding to b-th rectangular block in k-th depth difference image I_{diff},and $F2^b_k$ indicates the representative minimum change in depth corresponding to b-th rectangular block in k-th depth difference image, R_b indicates the pixels that belong to b-th rectangle block

Initially the maximum and minimum values of difference are obtained at finer resolution using 36 windows each of size $x \times y$, symmetrically laid about the image center. The remaining features are obtained by merging the windows sequentially to sizes of : $2x \times 2y$ (Fig. 5(b)), $3x \times 3y$ (Fig. 5(c)), $6x \times 3y$ (Fig. 5(d)), $3x \times 6y$ (Fig. 5(e)) and $6x \times 6y$ (Fig. 5(f)) as illustrated.

The representative maximum and minimum change in depth of all the 54 patches are combined as the feature vector of length 54×2 which is 108.

The 108 dimensional feature vector is then combined with the 14 dimensional feature vector obtained from Hu moments to form the final feature vector.These feature vectors are scaled appropriately to keep the real,imaginary values within -1 to 1 range.

VI. Results and Discussions

This section describes the database creation, experimental setup and the results obtained using SVM classifier.

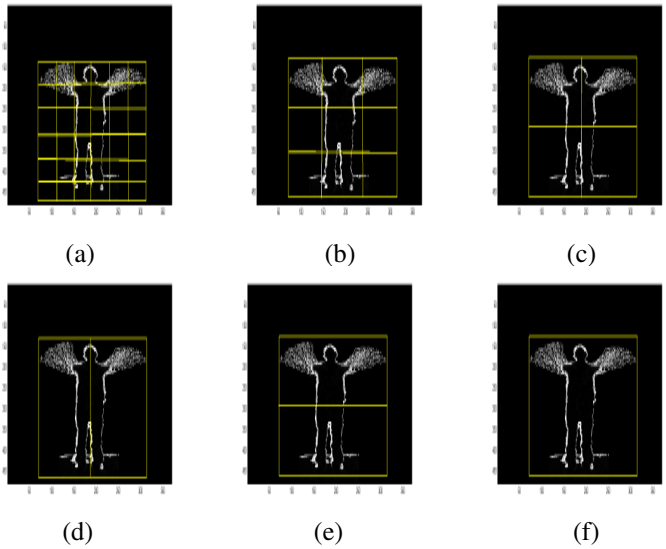


Fig. 5. Feature Extraction: Fine to coarse strategy (a) First level Fine Image patches (No. patches=36) (b) second level patches (No. of patches=9) (c) 3rd level coarse patches (No. of patches = 4) (d)-(f) other coarse resolution patches (No. of patches=5.)

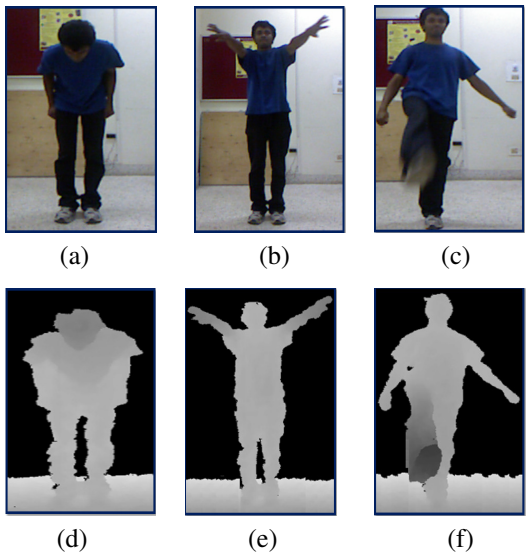


Fig. 6. Some of the actions considered in our database (a)-(c) RGB images of bend,swim, kick (d)-(f) Corresponding depth images

A. Database Creation

We created our database using the Kinect. Both the depth and RGB image were recorded at a rate of 30 frames per second. The depth images were saved as 11 bit images. The resolution of the depth and RGB images were 640×480. We placed the Kinect at a fixed height from the floor so as to capture the subject's entire body. All actions were performed at a fixed distance from the Kinect. Our database consists of the following eight actions: swimming, bending, waving,kicking, bowling, jumping, boxing and stretching. Each action was performed by 8 subjects for three times. All actions were performed facing the camera. Figure 6 shows snapshots of some of the actions considered.

B. Experimental Setup

In our Database, each action is captured 3 times and on an average we have 350 to 400 frames of depth images representing each action per subject. We consider a time window of 16 frames ($N = 16$) to generate the motion history image, average depth image and depth difference image. We consider an overlap of 12 frames to ensure continuity of motion. A Feature vector of length 122 is generated for each 16 frame time window. The training set includes the feature vectors of all the actions of all the subjects except one which is used for testing. Hence the testing is person independent i.e. the person used for testing is not included in training. We have used the well known support vector machines to classify the actions. LIBSVM [1] was used to train our model and test the performance of the model. Each instance in the training set contains one target value (i.e. the class labels) and several attributes (i.e. the features or observed variables). The SVM classifier first requires the data to be scaled down. The main advantage of scaling is to avoid attributes in greater numeric ranges dominating those in smaller numeric ranges. Then we have used RBF kernel which non-linearly maps samples into a higher dimensional space so it, unlike the linear kernel, can handle the case when the relation between class labels and attributes is non-linear.

C. Performance Evaluation

We have evaluated the representation efficiency of Hu moments and hierarchical bounding box based features separately. The 14-dimensional feature vector formed using Hu-moments are able to represent the actions with an accuracy of 46% on an average. The 108 dimensional feature vector formed using the hierarchical division of bounding box (HBB) is found to represent the actions with an accuracy of 68% on an average.

[1]LIBSVM: www.csie.ntu.edu.tw/~cjlin/libsvm/

TABLE I
ACTION RECOGNITION RESULTS - CONFUSION MATRIX (TEST)

Action	Swimming	Bending	Waving	Stretching	Boxing	Bowling	Jumping	Kicking
Swimming	100	0	0	0	0	0	0	0
Bending	0.22	77.23	4.01	14.06	2.45	0	0	2.23
Waving	0	5.38	94.62	0	0	0	0	0
Stretching	0	9.75	9.52	81.23	0	0	0	0
Boxing	0	02.33	0	0	97.66	0	0	0
Bowling	0	0	0	0	0	97.94	2.059	0
Jumping	0	0.2	0.1	0	0	11.94	87.75	0
Kicking	0	0	0	0	0	0	16.56	83.33

TABLE II
PERFORMANCE OF EACH FEATURE

Action	Recognition Accuracy (%)		
	Combined	Only HBB	Only Hu
Swimming	100	72.34	34.8
Bending	77.23	68.75	67.30
Waving	94.61	60.78	26.43
Stretching	81.23	43.37	43.37
Boxing	97.66	79.13	30.83
Bowling	97.94	93.80	90.0
Jumping	87.75	63.45	46.80
Kicking	83.33	59.37	25.30

The combination of both the the 14 dimensional vector formed from Hu moments and the 108 dimensional feature vector from the hierarchical division of bounding box boosts the accuracy to 90% on an average. This is because Feature vectors obtained from both methods (Hu moments and HBB) are complementary to each other. Features obtained from Hu moments only capture the information about the spatial motion flow whereas HBB features captures the dynamics of the motion during the action. The recognition accuracy for the combined Hu and HBB features along with its independent performance is given in table II. The confusion matrix corresponding to the combined feature is shown in table I. The major confusion is in recognition of bending action. Action of bending is confused with boxing, stretching (moving hands forward and up). The is because of the fact that there is similar change in depth in one particular region of the silhouette. Waving is confused with stretching because of the fact that they have similar motion history image and average depth image at few instants of the action. The proposed method is able to recognize actions such as swimming, bowling, boxing and waving with very high accuracy.

VII. CONCLUSION

In this paper we have proposed an action recognition system using only the depth information obtained from the Kinect sensor. The depth variation of the silhouette regions that are represented using hierarchically divided regions are used for representing the actions along with Hu moments of MHI and average depth image. The combined feature set provided an accuracy of about 90% for the considered 8 actions. The silhouette extraction could be easily extracted using the depth image. Thus reducing the complexity of silhouette extraction and increases the robustness of action representation and recognition.

REFERENCES

[1] Aaron F. Bobick and James W. Davis, "The recognition of human movement using temporal templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 3, pp. 257–267, 2001.

[2] Junji Yamato, Jun Ohya, and Kenichiro Ishii, "Recognizing human action in timesequential images using hidden markov model," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, June 1992, pp. 379–385.

[3] Abhijit S. Ogale, Alap Karapurkar, and Yiannis Aloimonos, "View-invariant modeling and recognition of human actions using grammars," in *Revised Papers of the Workshops on Dynamical Vision (WDV05 and WDV06), Lecture Notes in Computer Science*, May 2007, p. 115126.

[4] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 12, pp. 2247–2253, Dec. 2007.

[5] A. Yilmaz and M. Shah, "A differential geometric approach to representing the human actions," *Computer Vision and Image Understanding*, vol. 119, no. 3, pp. 335–351, 2008.

[6] Alexei A. Efros, Alexander C. Berg, Greg Mori, and Jitendra Malik, "Recognizing action at a distance," in *Proceedings of the International Conference on Computer Vision*, Oct. 2003, vol. 2, pp. 726–733.

[7] R. Venkatesh Babu, B. Anantharaman, K. R. Ramakrishnan, and S. H. Srinivasan, "Compressed domain action classification using HMM," *Pattern Recognition Letters*, vol. 23, no. 10, pp. 1203–1213, 2002.

[8] Jamie Shotton, Andrew Fitzgibbon, Mat Cook, Toby Sharp, Mark Finocchio, Richard Moore, Alex Kipman, and Andrew Blake, "Real-time human pose recognition in parts from a single depth image," in *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 2011.

[9] Ronald Poppe, "A survey on vision-based human action recognition," *International Journal of Computer Vision*, vol. 28, no. 2/3, pp. 976–990, 2010.

[10] Daniel Weinland, Remi Ronfard, and Edmond Boyer, "A survey of vision-based methods for action representation, segmentation and recognition," *Computer Vision and Image Understanding*, pp. 224–241, Feb. 2011.

[11] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: A local svm approach," in *Proceedings of International Conference on Pattern Recognition*, Aug. 2004, pp. 32–36.

[12] Daniel Weinland, Remi Ronfard, and Edmond Boyer, "Free viewpoint action recognition using motion history volumes," *Computer Vision and Image Understanding*, vol. 104, no. 2-3, pp. 249 – 257, 2006, Special Issue on Modeling People: Vision-based understanding of a person's shape, appearance, movement and behaviour.

[13] Ming-Kuei Hu, "Visual pattern recognition by moment invariants," *IEEE Transactions on Information Theory*, vol. 8, no. 2, pp. 179–187, Feb. 1962.