Temporal Segmentation of Gestures Using Gradient Orientation of Depth Images

*Tonmoy Roy, *Upal Mahbub, *Md. Shafiur Rahman, *Hafiz Imtiaz, and †Md. Atiqur Rahman Ahad

*Department of Electrical and Electronic Engineering
Bangladesh University of Engineering and Technology, Dhaka-1000, Bangladesh
E-mail: tonmoy_roy@live.com; omeecd@eee.buet.ac.bd; shafeey@live.com; hafiz.imtiaz@live.com
†Department of Applied Physics, Electronics & Communication Engineering
University of Dhaka, Bangladesh
E-mail: atiqahad@univdhaka.edu

***Abstract*—This paper deals with the problem of temporal segmentation present in practical applications of action and gesture recognition. In order to separate different gestures from gesture sequences a novel method utilizing depth information, oriented gradients and supervised learning techniques is proposed in this paper. The temporal segmentation task is modeled as a two-class problem and histogram oriented gradients of the gesture boundary and non-boundary sample frames are incorporated in the feature table as positive and negative training vectors, respectively. The classification task is carried out using both Euclidean Distance based and Support Vector Machine classifiers. A clustering algorithm is employed thereafter to finally locate the temporal boundaries of gestures. Through extensive experimentation it is shown that, the proposed method can provide a high degree of accuracy in temporal gesture segmentation in comparison to a number of recent methods.**

***Index Terms*—gesture recognition, depth image, motion history image, histogram of gradient, k-means clustering.**

I. Introduction

Human gestures/actions are non-verbal bodily actions used to express intentions and can be instantly recognized by people. Gestures are used to depict sign language to deaf people, convey messages in noisy environments, and now a days, interact with machines [1]. Automatic gesture recognition systems that are necessary and integral parts of any human machine interaction system operates in the principle of converting high-bandwidth video data into compact description of the presence and movement of the human subjects in that scene. Many gesture recognition algorithms have been proposed in recent years [2] [3] [4] [5]. However, reliable gesture recognition remains a challenging area due the complexity of human movements, cluttered background, self occlusion, illumination and such other problems [1] [6] [2]. On the other hand, in order to improve the recognition performances, gesture models are often complicated making the recognition process computationally expensive.

For real time gesture recognition systems, one major factor is temporal segmentation of gestures from videos containing a collection of several gestures performed by a subject [7]. The gesture recognition process gets directly hampered if the temporal segmentation process is not robust. This two class problem of detection of the frames depicting start and end of a gesture itself can be modeled as a gesture recognition task which is, in many cases, quite difficult due to irregular movements by human subjects. In recent years, there have been several works on temporal segmentation of gestures, especially on the Chalearn Gesture dataset [8] [9] [1]. However, all these works are more focussed on gesture recognition while leaving the temporal segmentation part poorly attended. It is obvious that perfect in temporal segmentation will surely improve the recognition results of these methods to a great extent.

In this paper, an efficient mechanism is developed to separate different gestures from long test sequences of depth images of human gestures provided in ChaLearn Gesture dataset, which involves the use of depth images obtained utilizing infrared depth sensors. A supervised learning method is used on features extracted from the training videos. The training vector is formed by extracting localized features from depth images based on count of occurrences of gradient orientation. A support vector machine is trained to classify the frame sequences to find the boundary frames for the gestures. Finally a partitioning algorithm is used to cluster the classified frames to find the temporal boundaries of gestures. A satisfactory temporal segmentation is achieved for a wide variety of gestures including sign languages, cluttered background scenarios, partially visible human figure scenarios, etc. is achieved.

II. Proposed Method

In this paper, a method of temporal gesture segmentation from a video of depth images containing a number of gestures is proposed. It is assumed that the gestures will be drawn from a vocabulary of gestures, generally related to a particular task. Though, both RGB image and depth image are available in the dataset, the proposed algorithm focuses only on the depth data obtained from the sensor for the temporal segmentation recognition model.

A. Dataset

There are some clearly-defined hand or body or head gesture datasets, e.g., Cambridge gesture dataset [10], Naval Air

978-1-4799-0400-6/13/$31.00 ©2013 IEEE

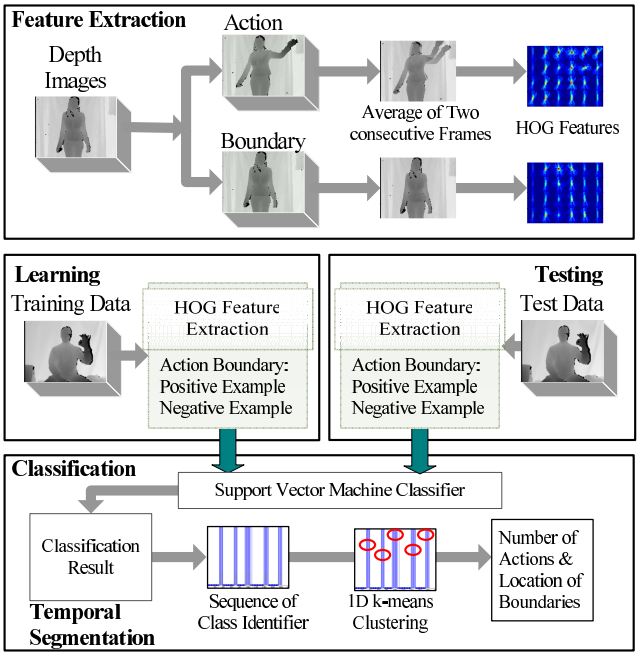

Fig. 1. Schematic of the Proposed Temporal Segmentation Method

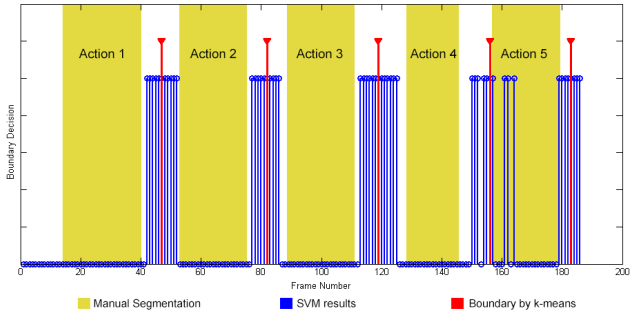

Fig. 2. Classification and Clustering for Temporal segmentation

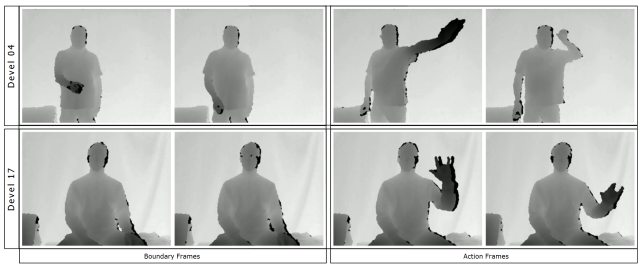

Fig. 3. Frames used for training

Training and Operating Procedures Standardization (NATOPS) aircraft handling signals database [11], Keck gesture dataset [12], Korea University Gesture (KUG) database [13], etc. However, most of these datasets address a particular type of gestures limited to very few classes and application domains. Therefore, for the simulation purpose of the proposed method, a very rich, but extremely complicated dataset, namely, the ChaLearn Gesture Dataset (CGD2011), is considered in this paper [8].

The ChaLearn Gesture database contains nine categories of gestures corresponding to various settings or application domains. Some frames of the video samples of the ChaLearn gesture dataset are shown in Fig. 3. The dataset consists of videos from RGB and Depth cameras of a Microsoft Kinect^{TM} Sensor. In the videos, a single user is portrayed in front of a fixed camera, interacting with a computer by performing gestures. The complexities in this dataset are introduced by wide variation of the types of actions, environment, and position of the performer. The performers were instructed to return to the initial rest position once between each gesture, however, in many cases the performers either did not follow it properly or the gesture itself consists of positions very much similar to the rest position, making the temporal segmentation problem very much difficult.

B. Feature Extraction and Training

The depth data provided in the dataset are in RGB format and therefore, the mean of the three channels is taken instead of the individual channel values. The depth data is a true representation of object distance from the camera by varying intensity of pixels from dark to bright for near to far away objects, respectively. Thus, the depth image contains three dimensional information of the user.

The proposed method calculates histogram oriented gradients (HOG) of the depth images to obtain the feature vector. This vector is used with a traditional supervised learning method namely, support vector machine (SVM), to find the gesture boundary frames of the test videos [14] [15]. To make the boundary location usable, the classification result is partitioned with a clustering algorithm to partition all the classified frames at the end of a gesture into a single frame location. The whole process is diagrammatically represented in Fig. 1.

The proposed method takes sample frames from the beginning and end of the training videos of a data batch. These frames are labeled as 'Action Boundary: True'. According to the description of the ChaLearn Gesture Dataset, it is expected that the user will start from a initial position and come back to the original initial position after performing each action. From each training videos containing a single gesture only, it is assumed that discarding the first 30% and the last 30% frames will result in frames that do not have the user in the initial position. Frames were uniformly selected from this middle portion and labeled as 'Action Boundary: False'. Random selection ensured low probability of a body position in the frame being similar to the position in initial frames.

Two sets of Histogram Oriented Gradient (HOG) features were calculated for the two sets of training data. The HOG were calculated from the depth images using spatial bins of width of $sBin$ pixels and $oBin$ orientation bins [16] [17] [18]. The reason behind using the depth images to calculate the HOG features were that the depth images lacked texture. This was a beneficial quality that ensured that the gradients of an object which is separated from the background or other objects by depth could be calculated accurately without worrying about lighting or texture. The HOG features are expected to be distinctly different between temporal boundary frames and non-boundary frames.

As suggested in [16], a linear SVM was trained using the HOG features. After training the SVM with the training videos, the classification operation was run on the testing videos of the same data batch. Each frame was labeled '0' corresponding to 'Action Boundary: False' or '1' corresponding to 'Action Boundary: True' as per the classification results. This labeling however was not enough for temporal action segmentation. There were frames labeled '1' in the middle of an action where the user may have gone too close to the initial position. Again slight movement by the user at the end of an action meant that there were frames labeled '0' at ends of action. The '1' labels were considered to be points in a 1 dimensional (time) space and therefore, to classify the labels as being part of the same action a simple clustering algorithm namely, the k-means clustering, is used. The k-means clustering returned the centers of the clusters as shown in Fig. 2. This means that the cluster centers obtained were most likely to be part of a boundaries.

1) Histogram Oriented Gradients: A histogram of gradients is calculated with each gradient quantized by its angle and weighted by its magnitude. The centered gradient is used except at boundaries. Uncentered gradient is used for the boundaries. Trilinear interpolation is used to place each gradient in the appropriate spatial and orientation bin. For each resulting histogram with $oBin$ number of orientation bins, w different normalization are computed using adjacent histograms, resulting in an $w \times oBin$ length feature vector for each region. Boundary regions are discarded. The computed features are NOT identical to [16]. Specifically, there is no Gaussian spatial window.

Even without precise knowledge of edge positions or intensity gradients for an object, the visual aspect and shape can often be distinguished fairly accurately using the distribution of local intensity gradients. For implementation this is achieved by dividing the image into spatial regions and a local one dimensional histogram is constructed for each region by accumulating the intensity gradients of all the pixels in the region. The combined histogram entries (one for each region) forms the desired representation. For better invariance to illumination the local responses are contrast normalized before use. To obtain this contrast normalized a measure of local histogram energy over larger spatial regions is accumulated. These results are used to normalize all the smaller regions in the larger blocks.

In our experiments [19] was used to calculate the HOG feature.

2) Classification: For classification of the extracted HOG features, we propose utilization of a Euclidean distance (ED) based similarity measure and a support vector machine (SVM) based similarity measure. [20] Given N-dimensional feature vector for the class i of the m-th sample depth image are $\gamma^m(1), \gamma^m(2), ..., \gamma^m(N)$ and the t-th test sample depth image frame with a feature vector $\kappa_t(1), \kappa_t(2), ..., \kappa_t(N)$, a similarity measure between the t-th test sample of the unknown class and the depth image of the class i is defined as

$$C_f^i = \sum_{p=1}^{N} |\kappa_i^m(p) - \kappa_t^m(p)| \tag{1}$$

where a particular class represents a frame being an action frame or an action boundary frame. Therefore, according to 2, given the t-th test action image, the frame action is classified as the class j among the 2 classes when

$$C_t^i < C_t^s; \forall i \neq s, \forall s \in 1, 2, ...r. \tag{2}$$

SVM can also be used for action classification using the proposed features [14] [15]. After extraction of the features as explained in, previous sections, SVM is used to train the system with some randomly picked frames for action and frames from near the beginning and the end for temporal action boundary. In our experiments of the ChaLearn Gesture Challenge Database [8] a linear kernel with a soft margin produced results more superior than a Euclidean distance-based classifier.

3) k-means clustering: k-means clustering is a method of cluster analysis which aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean. This results in a partitioning of the data space into Voronoi cells [21] [22]. K-means is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori. The main idea is to define k number of centroid, one for each cluster. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early grouping is done. At this point the k new centroids are re-calculated as barycenters of the clusters resulting from the previous step. After having these k new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated therefore, an as a result the change in the location of the k centroids can be noticed step by step until no more changes are done.

If classification results are considered as points on a line representing time, it will be a special case of only one dimension. The k-means was used here only to find clusters of points on the line. For this the number of centroids or k was assumed to be from 1 to 5, representing the number of boundaries in the videos (discarding the portion before the first

gesture). With these values of k, the centroids were calculated and then a cost function was used to penalize too many points in a cluster, points that are not directly connected via another point to the centroid and number of centroids. This way, the optimal number of centroids were achieved that produced the minimum cost.

III. Experimental Results

For the purpose of analysis, the first 20 lexicons, namely $Devel01$ to $Devel20$ from ChaLearn Gesture Dataset [8], [23] are considered. In this dataset, each lexicon has 8 to 15 training videos consisting of only one action. These videos were used to extract the training features. According to the description of the dataset the users are supposed to return to their initial posture after performing each gesture. As a result, frames from the start and the end of the training sequences contain frames with the users in positions that would similar to the position at the temporal action boundary. Both boundary and non-boundary training frames were selected using a uniform distribution to minimize the probability of false positive error for classification of a frame as a boundary frame.

The performance of the proposed method is evaluated using the average length deviation of the number of predicted actions from the actual number of actions in the video sequences, which is expressed as

$$TeLen = \frac{F + M}{A}. \tag{3}$$

Here, F is the number of actions detected that were not in actuality an action, M is the number of actions that were present but was not detected and A is total number of actions in the video sequence

In the experiments it is seen that due to low frame rate of the videos, taking average of two consecutive frames produces the best result. After doing the averaging the resulting depth images are used to calculate the HOG features. These features contain all the information required to describe the human body position. To calculate the HOG features, the frames are divided in non-overlapping windows of size $sBin$ and $oBin$ number of orientation in the range $[0^\circ, 180^\circ]$. The hog feature is rearranged as a single vector and used with the classification method to classify each frame of the test videos as either being a temporal action segment boundary or a non-boundary. Through extensive experimentation it is found that the optimum value for $sBin$ is 8 and the optimum value for $oBin$ is 9, which is also consistent with the findings of [16].

In Table I the performance of the proposed temporal segmentation technique employing two separate classifiers, namely, the Euclidean Distance (ED) based classifier and the Support Vector Machine (SVM) classifier, are shown in terms of average length deviation. It can be observed that the support vector method provides more accurate results. The k-means clustering technique is employed in each of these cases after classification to perform cost minimization operation until only one boundary per cluster is obtained while discarding unwanted aberrations. It is to be noted that the frames were scaled to 20% of the original size before performing the segmentation operation to produce the results in Table I.

TABLE I
Classification result for action boundaries

Devel	01	02	03	04	05
ED	2.22	17.05	15.22	33.33	31.52
SVM	1.11	11.36	6.52	7.78	9.78
Devel	06	07	08	09	10
ED	11.11	0.00	11.24	2.20	2.20
SVM	7.78	1.10	5.62	3.30	1.10
Devel	11	12	13	14	15
ED	11.96	6.74	3.41	4.35	4.35
SVM	7.61	6.74	3.41	6.52	5.43
Devel	16	17	18	19	20
ED	6.90	1.09	2.22	6.59	20.00
SVM	3.45	1.09	1.11	8.79	18.89

TABLE II
Performance Comparison

Method	TeLen
HOG + SVM (ours)	5.98%
3DHOF + GHOG [9]	9.03%
Template Matching [9]	15.12%
Manifold LSR [1]	6.24%

The results of our proposed method is compared with few other methods of temporal segmentation proposed by other researchers in Table II. It can be observed from the table that the proposed method outperforms more complex methods such as $3D$HOF with $GHOG$ and LSR Manifold and is surely better than simple template matching algorithms in which correlation based template matching is done.

IV. Conclusion

In this paper, an approach for temporal gesture segmentation is proposed. It employs a combination of distribution of local intensity gradients, a supervised classifier and partitioning algorithm to separate gestures from a sequence of depth images. In the proposed method, intelligent reasoning and statistically uniform distributions are utilized to get the training images. A feature named histogram of oriented gradients that is basically the local gradients of small regions in the image is extracted from the training images. Traditional supervised classifiers are used on the training features to separate the gesture boundary and non-boundary frames. Finally k-means clustering algorithm is used partition the classifier results and separate the gestures from the videos. The dataset, namely, the ChaLearn Gesture Dataset 2011, targeted for experimental evaluation is a rich, but difficult dataset to handle having a lot of variation and classes. This fact presented unprecedented obstacles which had to be overcome systematically with different

methods. Through extensive evaluation it is proved that the proposed combination of features can provide a satisfactory level of temporal gesture segmentation for one of the most complex dataset of gestures to date.

REFERENCES

[1] Y. M. Lui, A least squares regression framework on manifolds and its application to gesture recognition, in: Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on, 2012, pp. 13 –18. doi:10.1109/CVPRW.2012.6239180.

[2] M. A. R. Ahad, Computer Vision and Action Recognition: A Guide for Image Processing and Computer Vision Community for Action Understanding, Atlantis Ambient and Pervasive Intelligence, Atlantis Press, 2011.

[3] S. Mitra, T. Acharya, Gesture recognition: A survey, Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on 37 (3) (2007) 311 –324. doi:10.1109/TSMCC.2007.893280.

[4] H. Wang, M. M. Ullah, A. Kläser, I. Laptev, C. Schmid, Evaluation of local spatio-temporal features for action recognition, in: BMVC, British Machine Vision Association, 2009.

[5] M. A. R. Ahad, J. K. Tan, H. Kim, S. Ishikawa, Motion history image: its variants and applications, Mach. Vision Appl. 23 (2) (2012) 255–281. doi:10.1007/s00138-010-0298-4.

[6] M. A. R. Ahad, J. Tan, H. Kim, S. Ishikawa, Analysis of motion self-occlusion problem for human activity recognition, Vol. 5, 2010, pp. 36–46.

[7] L. Shao, L. Ji, Y. Liu, J. Zhang, Human action segmentation and recognition via motion and shape analysis, Pattern Recogn. Lett. 33 (4) (2012) 438–445. doi:10.1016/j.patrec.2011.05.015.

[8] ChaLearn Gesture Dataset (CGD2011), ChaLearn, California, 2011.

[9] G. M. F. O. Sean Ryan Fanello, Ilaria Gori, One-shot learning for real-time action recognition.

[10] T.-K. Kim, R. Cipolla, Canonical correlation analysis of video volume tensors for action categorization and detection, IEEE Trans. Pattern Anal. Mach. Intell. 31 (8) (2009) 1415–1428. doi:10.1109/TPAMI.2008.167.

[11] Y. Song, D. Demirdjian, R. Davis, Tracking body and hands for gesture recognition: NATOPS aircraft handling signals database., in: FG, IEEE, 2011, pp. 500–506.

[12] Z. Jiang, Z. Lin, L. Davis, Recognizing human actions by learning and matching shape-motion prototype trees, IEEE Trans. Pattern Anal. Mach. Intell. 34 (3) (2012) 533–547. doi:10.1109/TPAMI.2011.147.

[13] B.-W. Hwang, S. Kim, S.-W. Lee, A full-body gesture database for automatic gesture recognition, in: Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition, FGR '06, IEEE Computer Society, Washington, DC, USA, 2006, pp. 243–248. doi:10.1109/FGR.2006.8.

[14] J. Suykens, J. Vandewalle, Least squares support vector machine classifiers, Neural Processing Letters 9 (1999) 293–300.

[15] M. Awad, X. Jiang, Y. Motai, Incremental support vector machine framework for visual sensor networks, EURASIP J. Appl. Signal Processing 2007 (2007) 222–222.

[16] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, Vol. 1, 2005, pp. 886 –893 vol. 1. doi:10.1109/CVPR.2005.177.

[17] V. Chandrasekhar, G. Takacs, D. M. Chen, S. S. Tsai, Y. Reznik, R. Grzeszczuk, B. Girod, Compressed histogram of gradients: A low-bitrate descriptor, Int. J. Comput. Vision 96 (3) (2012) 384–399. doi:10.1007/s11263-011-0453-z.
URL http://dx.doi.org/10.1007/s11263-011-0453-z

[18] Q. Zhu, M.-C. Yeh, K.-T. Cheng, S. Avidan, Fast human detection using a cascade of histograms of oriented gradients, in: Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on, Vol. 2, pp. 1491–1498. doi:10.1109/CVPR.2006.119.

[19] P. Dollár, Piotr's Image and Video Matlab Toolbox (PMT), http://vision.ucsd.edu/~pdollar/toolbox/doc/index.html.

[20] U. Mahbub, H. Imtiaz, M. A. R. Ahad, An optical flow based approach for action recognition, in: Computer and Information Technology (ICCIT), 2011 14th International Conference on, 2011, pp. 646–651. doi:10.1109/ICCITechn.2011.6164868.

[21] J. B. MacQueen, Some methods for classification and analysis of multivariate observations, in: L. M. L. Cam, J. Neyman (Eds.), Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability, Vol. 1, University of California Press, 1967, pp. 281–297.

[22] G. A. F. Seber, Cluster Analysis, John Wiley & Sons, Inc., 2008, pp. 347–394. doi:10.1002/9780470316641.ch7.
URL http://dx.doi.org/10.1002/9780470316641.ch7

[23] ChaLearn Gesture Dataset. [online], ChaLearn, California, 2011.
URL http://www.kaggle.com/c/GestureChallenge