# CoMMANTEL'13

21 – 24 January 2013
Ho Chi Minh City, Vietnam

# International Conference on Computing, Management & Telecommunications

Welcome | Keynote | Copyright

Committee | Table of Contents | Search

Sponsors & Supporters | Author Index | Help

IEEE Catalog Number:
CFP1323T-CDR
ISBN: 978-1-4673-2086-3

## Algorithms for Different Applications

**Chair:** Nguyen Huynh Tuong, *Ho Chi Minh City University of Technology, Vietnam*

# Temporal gesture segmentation for recognition

Thi-Lan Le(1), Van-Ngoc Nguyen(1), Thi-Thanh-Hai Tran(1), Van-Toi Nguyen(1,2), Thi-Thuy Nguyen(3)

(1) Computer Vision Department
International Research Institute MICA
HUST - CNRS/UMI - 2954 - INP Grenoble

(2) L3i Laboratory
La Rochelle University
France

(3) Faculty of Information Technology
Hanoi University of Agriculture,
VietNam

***Abstract*— This paper presents a method for temporal gesture segmentation based on the total activity of the video sequence. The new point of this method is that we apply some filters on the sequence and on the total activity plot that makes our method more robust to noise. This method has been shown to be very efficient on a very big data of the new contest CHALEARN on hand gesture recognition. This method can be a good reference for participants to the CHALEARN contest. The method is generic so could be applied for any shot boundary problem.**

***Keywords*— shot boundary detection, filtering, CHALEARN, hand gesture**

## I. Introduction

CHALEARN is a contest on gesture and sign language recognition from video data organized by Microsoft [1]. In this contest, candidates focus on hand gesture recognition. Besides challenges on video data, one of the most challenges of this contest is that only one sample will be used for training the gesture recognizer, knowing that traditional recognition methods require a lot of training data (e.g. SVM, Boosting).

In the video data of the CHALEARN contest, gestures are recorded consecutively. Each video clip contains from 1 to 5 gestures from a small vocabulary of 8 to 12 gestures. To be able to learn a sample of each gesture, it is necessary to temporally segment samples from video. We call each sample a video shot, each contains only one dynamic gesture. This problem returns to the famous shot boundary detection (SBD) problem.

SBD is the process of automatically detecting the boundaries between two shots in video. This problem has attracted much attention since video became available in digital form as it is an essential pre-processing step to almost all video analysis, indexing, summarization, search, and other content-based operations.

The general objective is to segment a given video sequence into its constituent shots, and to identify and classify the different shot transitions in the sequence. Different algorithms have been proposed, for instance, based on simple color histograms [2, 3], pixel color differences [4], color ratio histograms, [5] edges [6], and motion [7 - 9].

In this paper, we present a new method for temporal video segmentation. In particular, we will deal with videos containing dynamic gestures; one has been recorded after another. These videos coming from a very big database are taken by Microsoft Kinect sensor( both RGB and Depth information are provided) and used for training and evaluating one - shot learning algorithms for gesture recognition in the CHALEARN contest that we will present in more detail in the next section.

The problem of shot boundary detection has been investigated over many years and a large variety of techniques have been proposed, and evaluated, from simple comparison of adjacent frames to more complex recognition of patterns of motion vectors in compressed video [10].

As CHALEARN is a contest, the participants do not want to publish theirs methods until the end. Therefore, in the following, we will only present several published methods that have been proposed in the context of gesture segmentation of CHALEARN contest, the same as our objective.

In [11], the authors followed an appearance - based approach for temporal segmentation. They aspired to find the frames that are similar to the beginning and ending frame in the un-segmented video sequence and define them as the interval frames between two gestures in a video. Histogram of Oriented Gradients (HOG) is used as descriptor to represent frame and K Nearest Neighbor (KNN) is used to search for the similar frames.

In [12], the authors compute the difference between two consecutive frames, then accumulate the difference for each image to determine the total activity of the scene. Peaks (local maxima) of the total activity plot have been detected. Each peak and its surrounding correspond to a harsh movement of the hand. The gesture segmentation is then based on the location of peaks and the limitation of surrounding.

While presenting the method, two papers above do not report the temporal segmentation performance. In this paper, we present a method and evaluate its performance on a whole video data of CHALEARN contest based on shot boundary detection evaluation criteria.

The organization of this paper is follows. Section II presents briefly the CHALEARN contest and the video dataset to be processed. Section III describes the proposed method for temporal gesture segmentation. Section IV shows experimental

978-1-4673-2088-7/13/$31.00 ©2013 IEEE

results and evaluation. Section V concludes and give some idea for future works.

## II. CHALEARN CONTEST AND VIDEO DATASET

CHALEARN is a new challenge that focuses on recognizing gestures from video data recorded by a Microsoft KinectTM camera, which provides both RGB images and depth images obtained from an infrared sensor. KinectTM. This contest is organized in two rounds, one in conjunction with the CVPR conference (Providence, Rhode Island, USA, June 2012) and another with the ICPR conference (Tsukuba, Japan, November 2012). The results of the first round have been discussed in [1].

The dataset of CHALEARN is very impressive. We summarize here some information about this dataset. They are portraying a single user in front of a fixed camera, interacting with a computer by performing gestures to play a game, remotely control appliances or robots, or learn to perform gestures from an educational software. The they have collected a large dataset of gestures using the Microsoft Software Development Kit (SKD) interfaced to Matlab, which includes:

- 50,000 gestures recorded with the KinectTMcamera, including RGB and depth videos,
- with image sizes 240 x 320 pixels,
- at 10 frames per second,
- recorded by 20 different users,
- grouped in 500 batches of 100 gestures,
- each batch including 47 sequences of 1 to 5 gestures drawn from various small gesture vocabularies of 8 to 12 gestures,
- from over 30 different gesture vocabularies.

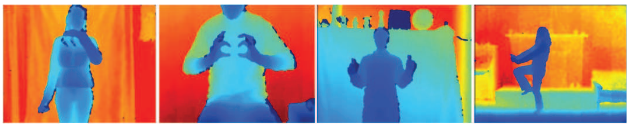

Figure 1. Color rendering of depth images from the gesture challenge database were recorded with a KinectTM camera

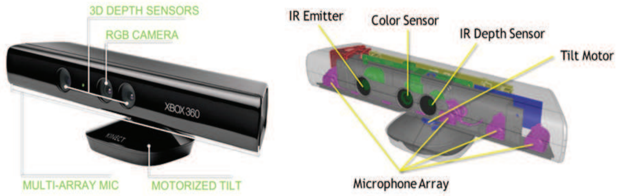


Figure 2. Color rendering of depth images from the gesture challenge database were recorded with a KinectTM camera and its components

Figure 3. Two informations provided by Kinect sensor (Depth – Left, RGB – Right)

To train and test a gesture recognizer, it is necessary to segment each sequence into shots, each shot corresponds to one gesture. As recorded by KinectTM sensor, each video has 2 data: RGB and depth.

## III. PROPOSED METHOD FOR TEMPORAL GESTURE SEGMENTATION

### *A. General framework*

We propose a framework for temporal gesture segmentation from video as in Figure 4. It consists of three main blocks

- **Pre-processing**: This block takes the input sequence and makes some pre - processing on frames.
- **Total activity computing**: This block takes the pre-processed sequence and computes the total activity of the whole sequence.
- **Temporal segmentation:** This block will make decision of segmentation based on the computed total activity.

In the following we will describe in more detail each step in the framework.

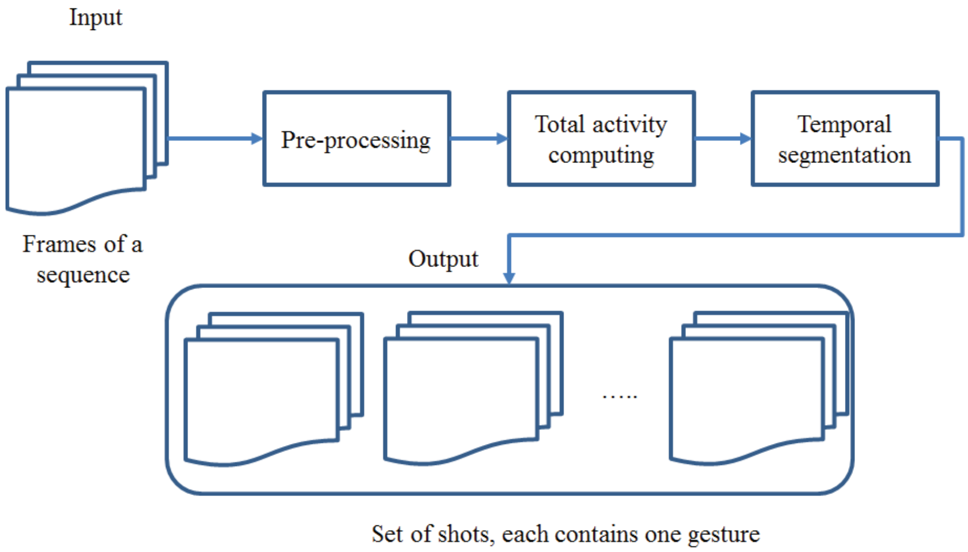


Figure 4. Main steps in the temporal segmentation framework

*B. Pre-processing*

Each sequence in the CHALEARN dataset composes of 2 videos: one is for RGB and the other is for depth. Both are stored in *avi* format. We propose to use depth information because it is invariant to illumination change. Then each depth frame in the sequence will be filtered by a spatial filter to reduce salt and pepper noise. As reported in [11], noise

pair of frames, we have (N-1) value represent the total activity of the sequence.

The figure 5 shows a plot of total activity computed for one sequence. High values correspond to the strong movement while small values correspond to small movement of the hand.

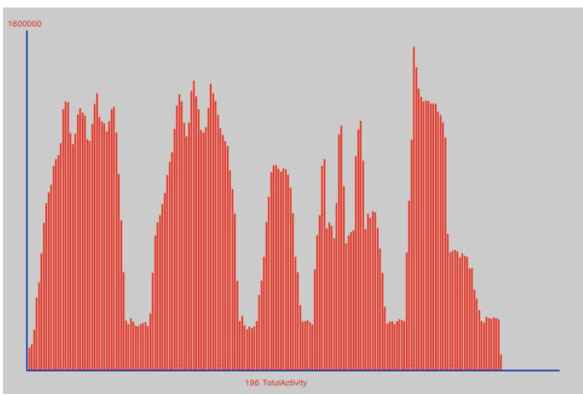

Figure 5. Plot of total activity of a sequence. Horizontal axe presents frame number, vertical axe represents the total activity of the current frame with the beginning frame

reduction improves 9% performance of the hand gesture recognition.

*C. Total activity computing and processing*

The total activity of a sequence is defined as follows: Given a sequence having N frames, first we compute the difference between the each pair of frames (the current frame and the beginning frame of the sequence). Then we calculate the total of absolute values of the differences. After processing with all

Values do not change in between several frames correspond to a transition between two hand gestures.

We found that the plot is quite noise. We then apply a Gaussian filter to smooth it. In our experiment, the Gaussian filter has sigma value equals to 7. After applying the filter, the plot is smoothed (figure 6).

*D. Temporal segmentation decision*

To segment a sequence, we will find local minima of its smoothed plot of total activity that could correspond to a cut position. Then we discard false alarms by keeping only positions at which the local minima satisfying following criteria:

- The value of local minimum needs to be smaller than the value of 3 surrounding.
- The value of local minimum needs to be smaller than a threshold. The threshold is not fixed a prior but must be adapted to the considered sequence.

As we can see in the figure 6, only four true local minima are kept. Their positions correspond to frame numbers at which we will cut into shots.

Figure 6. Plot of total activity and local minima of a sequence after applying a Gaussian filter

## IV. Experimental results

### A. Performance evaluation measures

To evaluate the performance of temporal segmentation algorithms, in [13], the authors have analyzed some quality criteria:

- **Accuracy:** In [14], a simple expression of accuracy has been proposed, which is equivalent to a measure used for the evaluation of speech recognition systems

$$\text{Accuracy} = \frac{N_T - (N_D + N_I)}{N_T} = \frac{N_C - N_I}{N_T} \quad (1)$$

  Where $N_T$, $N_D$, $N_I$, $N_C$ are respectively the number of actual transition effects present in the video database, the number of transition effects deleted, inserted and correctly found by the tested system.

- **Error rate:** Accuracy measure does not take into account the complexity of the video sequence nor its size, in [15], the author proposes a measure that evaluates the error rate (insertion and deletion off transitions effects) over the whole results of the segmentation algorithm.

$$\text{Error rate} = \frac{N_D + N_I}{N_T + N_I} = \frac{N_D + N_I}{N_C + N_D + N_I} \quad (2)$$

- **Recall and precision:** The measures are similar to information retrieval evaluations [16]:

$$\text{Recall} = \frac{N_C}{N_C + N_D} \quad (3)$$

$$\text{Precision} = \frac{N_C}{N_C + N_I} \quad (4)$$

  Apart from these measures, CHALEARN has provided ground truth of temporal segmentation on 20 batches, each contains 47 sequences. We will compare the results obtained by our method with this ground truth.

### B. Results

The results of temporal gesture segmentation are presented in Table 1. We apply our proposed method for 500 batches of the CHALEARN data. Each batch contains 47 sequences. We count the number of correct gestures in each sequence $N_C$, the number of deleted gestures $N_D$, the number of inserted gestures $N_I$, $N_T$ is the number of gestures in each sequence provided by the ground truth. In fact, the dataset provides ground truth for only 20 batches. For the 480 remaining batches (22560 sequences) we have to manually count $N_T$ by ourselves. Then we compute recall, precision, accuracy and error rate based on equations (1-4).

From the Table 1, we can see that the method that we proposed for temporal segmentation algorithm gives the best performance. It is better than the method propose by [12] . In addition, using the depth information gives lightly better performance than using the RGB information. The result is that the RGB information is more sensible to noise than the depth information.

*TABLE 1. COMPARISON OF PERFORMANCE*

| Methods | Accuracy | Error Rate | Recall | Precision |
|---|---|---|---|---|
| Our method using Depth | 93% | 6% | 94% | 98% |
| Our method using RGB | 91% | 8% | 93% | 98% |
| Method proposed by Keskin *et al.* [12] using Depth | 88% | 10% | 89% | 95% |
| Method proposed by Keskin *et al.* [12] using RGB | 87% | 10% | 89% | 94% |

## V. Conclusions

We presented a simple method for temporal gesture segmentation. This method has been validated as a very efficient for segmenting gestures from CHALEARN dataset. In the future, we will use this method as pre-processing to extract samples for hand gesture recognition training. This method is different from [11] and [12] at two points: we make the total activity by differencing the current frame with the first frame; this one is more robust to noise than [12], that computed the difference of two consecutive frames. In addition, both our method and [11] are based on the fact that the frames that are similar to the beginning and ending frame in the unsegmented testing video sequence and define them as the interval frames between two gestures in a video. But [11] used HOG (Histogram of Oriented Gradient) descriptor to describe each frame and KNN (K - Nearest Neighbor) to search for the similar frame. This method is more complex than ours. Neither [12] nor [11] evaluate quantitatively the segmentation algorithm. Then, the method that we presented in this paper can be a good reference for people working in the domain of temporal segmentation in general as well as participants to the CHALEARN contest in particular.


## ACKNOWLEDGEMENTS

This work is supported by the project "Some Advanced Statistical Learning Techniques for Computer Vision" funded by the National Foundation of Science and Technology Development, Vietnam under the grant number 102.01-2011.17.


## REFERENCES


[1] I. Guyon, V. Athitsos, P. Jangyodsuk *et al.*, "ChaLearn Gesture Challenge: Design and First Results," in Workshop on Gesture Recognition and Kinect Demonstration Competition, 2012.

[2] M. J. Swain, and D. H. Ballard, "Color indexing," *International Journal of Computer Vision,,* vol. 7, no. 1, pp. 11-32, 1991.

[3] A. Nagasaka, and Y. Tanaka, "Automatic video indexing and full-video search for object appearances," *Visual Database Systems II, Elsevier*, 1992.

[4] H. Zhang, A. Kankanhalli, and S. Smoliar, "Automatic partitioning of full-motion video," *Multimedia systems,* vol. 1, pp. 10-28, 1993.

[5] D. A. Adjeroh, and M. C. Lee, "Robust and efficient transform domain video sequence analysis: an approach from the generalized color ratio model," *Journal of Visual Communication and Image Representation,* vol. 8, no. 2, pp. 182-207, 1997.

[6] R. Zabih, J. Miller, and K. Mai, "A feature-based algorithm for detecting and classifying production effects," *Multimedia Systems,* vol. 7, no. 2, pp. 119-128, 1999.

[7] J. D. Courtney, "Automatic video indexing via object motion analysis," *Pattern Recognition,* vol. 30, no. 4, pp. 607–625, 1997.

[8] P. Bouthemy, M. Gelgon, and F. Ganansia, "A unified approach to shot change detection and camera motion characterization,," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 9, no. 7, pp. 1030-1044, 1999.

[9] S. Dagtas, W. Al-Khatib, A. Ghafoor *et al.*, "Models for motion-based video indexing and retrieval," *IEEE ransactions on Image Processing,* vol. 9, no. 1, pp. 88-101, 2000.

[10] A. F. Smeaton, P. Over, and A. R. Doherty, "Video Shot Boundary Detection: Seven Years of TRECVid Activity," *Elsevier*, 2009.

[11] D. Wu, F. Zhu, and L. Shao, "One shot learning gesture recognition from RGBD images," in IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW),, 2012, pp. 7-12.

[12] C. Keskin, F. Kırac, Y. Emre Kara *et al.*, "Randomized Decision Forests for Static and Dynamic Hand Shape Classification," in CVPR Workshop on Gesture Recognition 2012.

[13] R. Ruilob, P. Joly, S. Marchand - Maillet *et al.*, "Towards a Standard Protocol for the Evaluation of Video to Shots Segmentation Algorithms " in First European Workshop on Content-Based Multimedia Indexing, 1999.

[14] P. Aigrain, and P. Joly, "The automatic real time analysis of film editing and transition effects and its applications," *Computers and Graphics,* vol. 18, no. 1, pp. 93-103, 1994.

[15] J. M. Corridoni, and A. D. Bimbo, "Film semantic analysis," in CAIP, Czech Republic, 1995.

[16] J. S. Boreczky, and L. A. Rowe, "Comparison of video shot boundary detection techniques," in The International Society for Optical Engineering, 1996, pp. 170-179.