Recognition of hand gestures from cyclic hand movements using spatial-temporal features

Huong Giang Doan
MICA, Hanoi University of Science and Technology, CNRS/UMI2954-Grenoble INP
Industrial Vocational College Hanoi
huong-giang.doan@mica.edu.vn

Hai Vu
MICA, Hanoi University of Science and Technology
CNRS/UMI2954-Grenoble INP
hai.vu@mica.edu.vn

Thanh Hai Tran
MICA, Hanoi University of Science and Technology
CNRS/UMI2954-Grenoble INP
thanh-hai.tran@mica.edu.vn

ABSTRACT

Dynamic hand gesture recognition is a challenge field evenly this topic has been studied for a long time because of lack of feasible techniques deployed for Human-Computer Interaction (HCI) applications. In this paper, we propose a new type of gestures which presents a cyclic pattern of hand shapes during a movement. Through mapping of commands (e.g., turn devices on/off; increasing volume/channel) as output of a gesture recognition system, main purposes of the proposed gestures are to provide a natural and feasible way in control alliances in a smart home such as television, light, fan, door, so on. The proposed gestures are represented by both hand shapes and directions. Thanks to cyclic pattern of the hand shapes during performing a command, hand gestures are more easily segmented from video stream. We then focus on several challenges of the proposed gestures such as: non-synchronization phase of the gestures, change of hand shapes along temporal dimension and direction of hand movements. Such issues are addressed using combinations of spatial and temporal features extracted from consecutive frames of a gesture. The proposed algorithms are evaluated on several subjects. Evaluation results confirm that the proposed method obtains accuracy rates at 96% for segmenting a dynamic hand gesture and 95% for recognizing a command, averagely.

CCS Concepts

•**Human-centered computing** → **Interaction techniques; Gestural input;** *Human computer interaction (HCI);*

Keywords

Human Computer Interaction, Dynamic Hand Gesture Recognition, Spatial-Temporal Features, Dynamic Time Wrapping

WOODSTOCK '97 El Paso, Texas USA

DOI: http://dx.doi.org/10.1145/2833258.2833301

1. INTRODUCTION

Dynamic hand gesture recognition has been a very active research topic in the field of computer vision. Hand gestures are the most common and natural way for human to interact and communicate with computer or recently with smart-consumer electronics/home alliances. Many commercial systems have been developed nowadays. For instance, Samsung smart-TV can manipulate TV-functions using dynamic hand gestures. Omron introduces the smart-TV relying on face and hand recognition. PointGrab [5] proposed an unique solution including hand gesture recognition, face and user behavior analytic. Increasingly, in-air gesture recognition is being incorporated into consumer electronics and mobile devices like WiSee system [20]. WiSee extracts gesture information through wireless transmissions. Regarding to vision-based systems, a recent trend involves representing and recognizing hand gestures using Microsoft Kinect sensors [1] (or like-Kinect sensors such as Asus Kinect, Soft-Kinect). Approaching this trend, in this work, we propose a Kinect-based recognition technique with a new type of the gestures. Our main goals aim to create and deploy an efficient and accurate hand gesture recognition system in order to control devices such as televisions, lighting systems in a smart-home.

Although there are many recent works in dynamic hand gesture recognition [2][3][4][8], deploying such works into HCI systems is still existing a gap [27]. On one hand, a hand gesture recognition system has to wisely select feasible techniques in three phrases: detection, tracking and recognition. On the other hand, a hand gesture can be represented as a physical movement of the hands, arms, face and body with the intent to convey information or meaning. Therefore, there are many ways defining gesture representations for device control. In this study, we propose a new type of the hand gestures whose main characteristic is cyclic pattern of the hand shapes during a hand movement. Along a so-called hand-path (trajectory), hand moving direction represents meaning of a gesture or a corresponding command to control device. We utilize the cyclic pattern characteristics in order to easily detect a hand gesture from video stream. By using both depth and RGB image from Kinect sensors, hand regions are extracted more accuracy from background regions. We then analyze spatial features of the hand shapes through a Principal Component Analysis (PCA) model. A hand-path is extracted based on good-features points which

are detected and tracked between frame-by-frame. Our classification scheme aims to label a gesture by using similar measurements of both hand shapes at each state during its performing; as well as directions of the hand movements. The experimental results show that the proposed system obtains a high performance in term of recognition rate and computational time. Therefore, the proposed technique is feasible to deploy real HCI applications.

The rest of paper is organized as follows. Section 2 briefly surveys related techniques and public dataset of the dynamic hand gestures recognition. Section 3 describes the definition and characteristics of the proposed hand gestures. Section 4 describes the proposed techniques. Section 5 reports the experimental results. Finally, Section 6 concludes works and suggests further research directions.

2. RELATED WORKS

There are uncountable solutions for a vision-based hand posture recognition system in literature. Readers can refer good surveys such as [27][21] for technical details. In general, to recognize dynamic hand gestures, two problems need to address: gestures representation and recognition. Some methods represent dynamic hand gestures using motion history image [23], others extract trajectories of keypoints along a video sequence [9]. Regarding with the hand gesture classification, numerous methods for hand gesture recognition have been proposed:such as Neural Network [6], Hidden Markov Models (HMMs) [7], or Conditional Random Fields (CRFs) [26]. The phase synchronization is a particular issue for dynamic gestures, Dynamic Time Warping (DTW) [25] technique is preferred. Concerning Kinect-based hand gesture recognition, existing methods use different types of features: RGB, Depth, Skeleton, or combination of these features in an early or late fusion strategy [10]. However, ability to detect and recognize the human hand gestures posed many challenges due to the complexity of hand shapes, variation of gesture trajectories, cluttered background and light conditions. Consequently, deploying completed HCI applications still needs further investigations. In this work, we take into account a new type of the dynamic gestures. Therefore, we list below related dynamic hand gesture datasets.

A brief survey on existing dynamic hand gestures datasets: Using hands can help people communicate with computers in a more intuitive way. Whereas a hand pose presents a specific meaning in static gestures (e.g., American Sign Language - ASL [19]), hand movements convey very rich information in many ways. For example hands can be used for pointing at a person or at an object, conveying information about space, shape and temporal characteristics. Performance of the dynamic hand gestures strongly depends on the type of dataset used in relevant works. There were many self-defined dynamic hand gestures databases such as [18], [22], and [17]. In [18] a dynamic gesture presents through path in body-face space. Two classes of the gestures: deictic and symbolic are recognized to detect the intention of the user to execute a command. Authors in [22] proposed a dynamic gesture dataset collected from 10 subjects, it contains 10 gestures. In such dataset, the hand shapes depict figural representation or human actions. Recently, [17] shares hand gestures database which was designed according to mouse functionalities: cursor, left click, right click, mouse activation, and mouse deactivation. Moreover, many other works proposed the hand gestures databases that has been collected and widely published for different purposes such as MSR dataset utilized for evaluating human action recognition in [15][14]. The Cambridge-Gesture dataset is often utilized for evaluating performance of a hand detector [12][13]. In summary, the dynamic hand gesture datasets are designed in different characteristics such as repetitive, deictic, iconic, so on in order to intend and convey various messages, communications between human and computer.

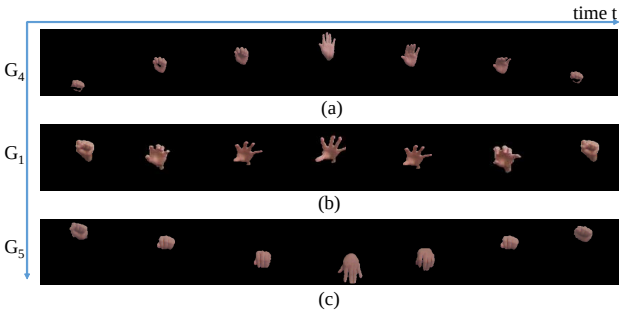


Figure 1: Defined dynamic hand gestures.(a) Increase gesture; (b) Turn On/Off gesture; (c) Decrease gesture.

3. CONSTRUCTING A NEW DATASET OF DYNAMIC HAND GESTURES

3.1 The proposed hand gesture dataset

In section 2 we have presented several datasets of dynamic hand gestures. Each dataset was designed for a specific application then had its own characteristics. Our work context is using hand gestures for home appliance controlling. Therefore, it is inconvenient to use existing datasets. We have studied and found following five common commands to control all home appliances: Turn on/Turn off; Next; Back; Increase; Decrease. We then design a new dataset of five dynamic hand gestures corresponding to these commands.

To control a device, an user stands in front of the Kinect sensor at a distance of $1.2 - 1.5$ m. A gestural command is composed of three stages: preparation; performing; relaxing. At preparation phase, the user stays immobile. At performing phase, the user raises his/her hand (e.g., right hand) and moves the hand according to a predefined trajectory. Simultaneously, while moving hand, the hand postures/shapes also change following three states: *initial* state, *implementing* state and *ending* state. It is notice that changes of the hand shapes follow a cyclic pattern from: fist shape at *initial* state to opening shape *implementing* state, and fist shape again at *ending* state. We list following descriptions of each gestures/command:

- Turn on/Turn off commands: one's hand doesn't change its direction or without any movements; just changes hand shape. From the initial state, hand opens slowly to fully open and then close slowly to initial state. Performing of these commands is illustrated in Fig.1(b), Fig.2(b).

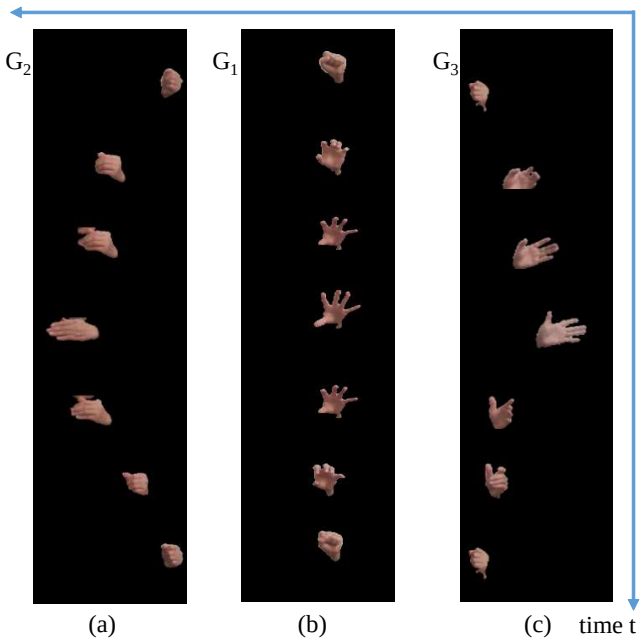


Figure 2: Temporal features of hand gestures. (a) Back gesture, (b) Turn On/Off gesture, (c) Next gesture.

- Increase and Decrease commands: The Increase command is started by the initial state, the right hand move up and opens slowly to fully open, after that the right hand move down and close slowly to the initial state. This gesture is illustrated in Fig. 1(a). The Decrease command is the same the Increase one but hand is changed by down direction, as shown in Fig. 1(c).

- Next and Back commands: States of a next/back gesture are similar to Increase and Decrease gestures. However, hand's movement of next/back command is performed in horizontal directions. Examples of the next/back gestures are shown in Fig.2(a)

Our dataset consists of the gestures collected by 6 subjects who are volunteers. Each subject implements five defined gestures; and each gesture is collected in five times.

3.2 Characteristics of the proposed dataset

The defined gestures are discriminated in both characteristics: hand shape and movement of hand directions. Hand shapes represent a cyclic pattern of a gesture, whereas second one represents meaning of gestures. They share following characteristics:

- For each gesture,hand shapes at initial and ending stages are identical (Fig. 3). We will utilize this feature to segment/cut a gesture from video stream (as described in Section 4.3).

- Gestures belonging a certain class could be different in length (e.g., as shown Fig. 3, number of postures in a type of gestures are different due to velocity of hand movement).

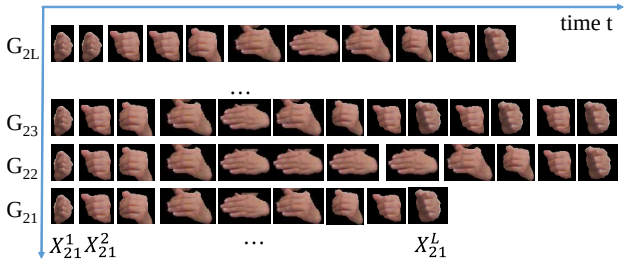


Figure 3: Differences of G_2 (Back) gesture in term of change speed performing. This illustration is collected from a subject

- The stages of gestures belonging a certain class could be non-synchronized phase. A state of the first gesture is more longer or shorter than second one. For example, initial stage of two gestures G_{21}, G_{23} in Fig. 3 are non-synchronized. Gestures G_{21} spends 5 frames to represent this stage, whereas G_{23} spends only 3 frames to express.

To easily describe the proposed technique, we denote following parameters. Let N is number of class of the proposed hand gestures to be considered. The proposed gestures are denoted by a set $G = \{G_i | i \in [1, N]\}$ (N equals 5 as defined in Sec. 3.1). In which, G_1 encodes Turn on/Turn off command; G_2, G_3 encodes Back/Next command; , G_4, G_5 is Increase/Decrease command, respectively. Each gesture G_{ij} includes consecutive frames: $G_{ij} = \{X_{ij}^k | k = [1, L]\}$. Dataset of a gesture G_i is: $G_i = \{G_{ij} | j = [1, M]\}$. Details of the proposed techniques are described in section below.

4. GESTURE RECOGNITION USING SPATIAL TEMPORAL FEATURES

In this section, two main issues are addressed. First, we must determine start and stop frames of gesture, this problem refers to temporal segmentation/boundary detection. Second, given a sequence of consecutive frames, we have to determine the label of gesture. We propose a framework that composes of the following main components (Fig. 4). Pre-processing step calibrates depth and RGB images. Hand segmentation step detects and segments hand region from the calibrated frames. Dynamic hand gestures are represented by spatial-temporal features. For classification, a K-NN (K-Nearest Neighbour) classifier is applied to predict the label of gesture.

4.1 Pre-processing

Depth and RGB data captured from Kinect sensor are not measured from the same coordinate system. In the literature, the problem of calibrating depth and RGB data has been mentioned in several works for instance [21]. In our work, we utilize the calibration method of Microsoft due to its availability and ease to use. The result of calibration is showed in Fig. 5.

4.2 Hand segmentation

As sensor and environment are fixed, we firstly segment human region using background subtraction technique. Both depth and RGB images can be used for the background subtraction. However, depth data is less sensitive with illumi-

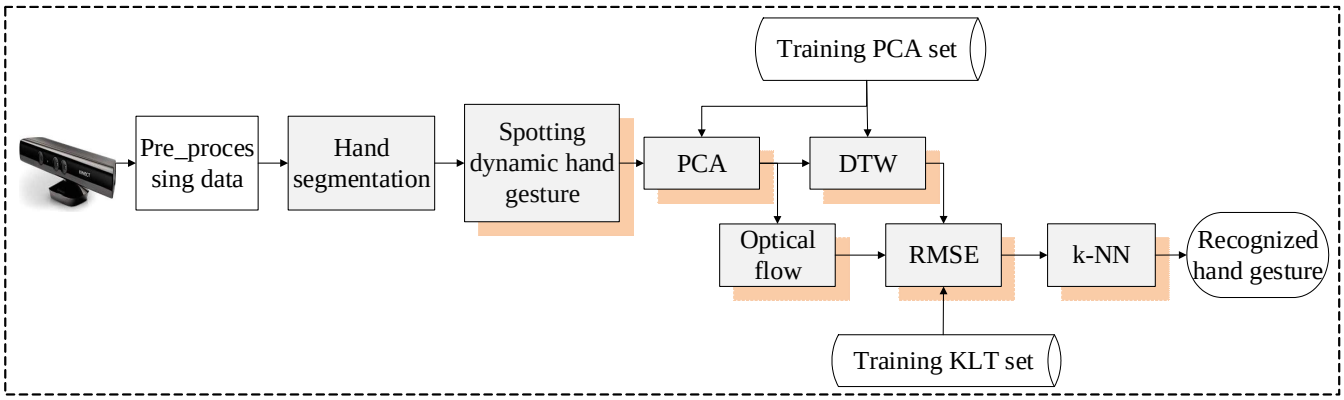


Figure 4: The proposed framework for detection and recognition dynamic hand gesture.

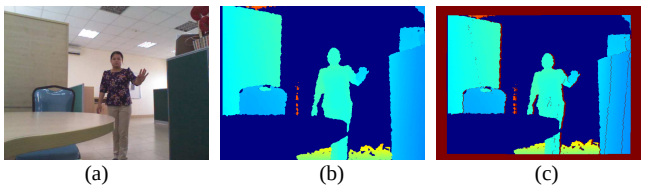


Figure 5: Calibration RGB-D images from the Kinect sensor.(a) RGB image, (b) Original Depth image, (c) Calibrated Depth image.

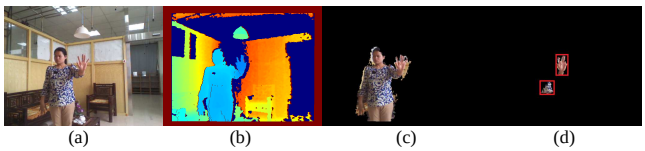


Figure 6: Body and Hand region detection.(a) RGB image; (b) Depth image; (c) Body extraction; (d) Candidates of hand region.

nation. Therefore, in our work we use depth images for background subtraction. Among numerous techniques of background subtraction, we adopt Gaussian Mixture Model (GMM) [24] because this technique has been shown to be the best suitable for our system. Figure 6(a-c) shows results of background subtraction. Given a region of human body (as shown in Fig. 6(c)), we continuously extract candidates of the hand (as shown in Fig. 6(d)) and a hand segmentation result X (as shown in Fig. 7) is selected after pruning hand region. Detail of this technique was presented in [10].

4.3 Gesture spotting

For a real application, the frames comes continuously. It's necessary to detect the start and boundary points of a gesture while overlooks the rest. Spotting dynamic hand gesture is detection and segmentation a dynamic hand gesture candidates from sequence frames that are captured from hand segmentation step. It is input of dynamic hand gesture recognition step and therefore it decide accuracy of recognition. In our context, as analyzed previously section, in all gestural commands, hand postures are similar at starting and stopping times. Moreover, hand shapes represent a

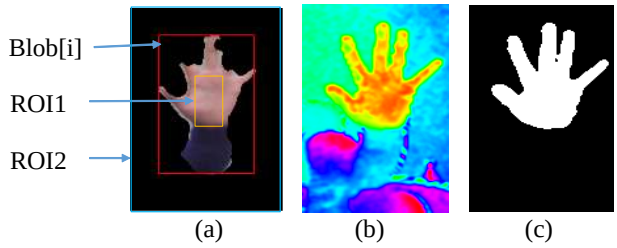


Figure 7: Hand segmentation.(a) A candidate of hand, (b) Malahanobis distance, (c) Hand detection.

cyclical pattern of a gesture. We then reply on these properties to detect and segment dynamic hand gestures. We propose an 1D function $G = f(x)$ representing area evolution of hand region with x is hand image that is presented in (1):

$$f(x) = \sum_{\forall i \in x} \delta \; with \; \delta = \begin{cases} 1 & \text{if } i \text{ belongs hand region} \\ 0 & \text{otherwises} \end{cases} \tag{1}$$

Red curve in Fig. 8 shows an example of the processing of a sequence including 389 consecutive frames. The area signal of dynamic hand gestures in this consecutive sequence are performed by one person. We observes that, areas of hand region for one dynamic hand gesture increase and decrease again. The local peaks of the signal correspond to cycles of dynamic hand gesture. To segment a dynamic hand gesture from continuous frames, we propose a procedure consisting of some main steps. Firstly, we smooth area signal $f(x)$ by a Gaussian function to remove noise and dummy local peaks that is shown by blue curve in Fig. 8. The next is applying morphological operator (opening operator) on the $f(x)$ function $O(f(x))$ as (4) that is shown by pink one in Fig. 8. The opening operator consists of a dilation operator $\delta_B(f(x))$ as (3) following an erosion operator $\varepsilon_B(f(x))$ as (2) on the smoothed signal $f(x)$ by the flat structuring element B which are defined as (2), (3), (4) follow formulas:

$$\varepsilon_B(f(x)) = (inf[f(x-t)])_{t \in B} = min\{f(x-t) - B(t) \| t \in B\} \tag{2}$$

$$\delta_B(f(x)) = sup[f(x-t)])_{t \in B} = max\{f(x-t) - B(t) \| t \in B\} \tag{3}$$

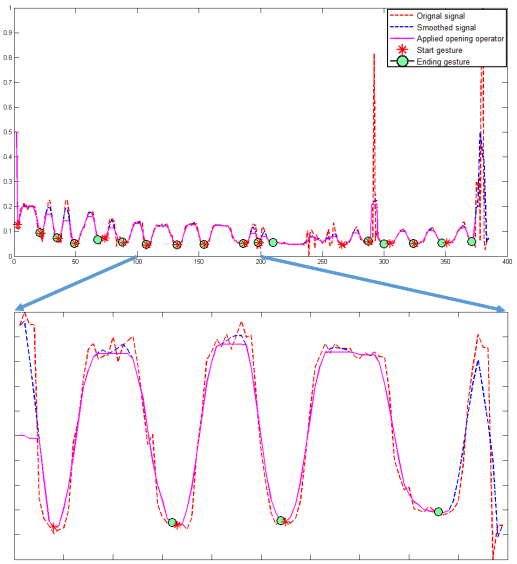


Figure 8: The finding of local peak from the original area signal of the hands.

$$O(f(x)) = \delta_B[\varepsilon_B(f(x))] \tag{4}$$

The final step is searching local peaks on the processed $O(f(x))$. As a result of this process, potential dynamic hand gestures are located within windows of length w which is number of frames. To satisfy the descriptions of a dynamic hand gesture, w can be adjusted in the context of changing values of sigma σ of the Gaussian function and length of the structuring element B. The σ is converted from full width at half maximum values is $2\sqrt{2ln2}\sigma$. Those values are to control the number of nearest neighbor frames for smoothing function. These values are set along the transit time. Fig. 8 shows two instances of local maximal peaks after applying opening operators. Red $(*)$ is the first minimize of local peak that is start of a dynamic hand gesture. Green $(\circ)$ is second minimize of local peak that is ending point of this dynamic hand gesture. This Fig. 8 shows sixteen dynamic hand gesture candidates from 389 consecutive frames.

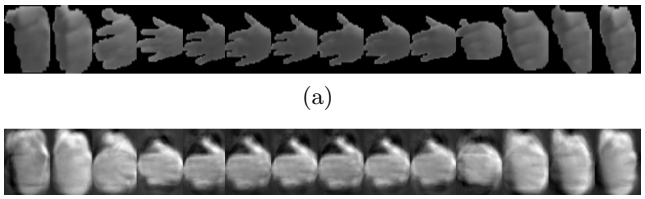
4.4 Spatial representation of gesture

Spatial features of a hand gesture represent the changing of hand shapes, non synchronization in phase and difference of length.

4.4.1 Hand shape representation

PCA is a popular technique for dimension reduction of feature space. After segmenting hand region, the image of hand region will be converted to a gray image and resized to the same size X $(64 \times 64$ pixels) as (5) and normalized to 1 standard deviation that is presented as (6):

$$X = \begin{bmatrix} x_{1,1} & x_{1,2} & ... & x_{1,64} \\ x_{2,1} & x_{2,2} & ... & x_{2,64} \\ . & . & ... & . \\ . & . & ... & . \\ . & . & ... & . \\ x_{64,1} & x_{64,2} & ... & x_{64,64} \end{bmatrix} \tag{5}$$



(a)

(b)

Figure 9: Go left hand gesture before and after projects on PCA space. (a) Gray hand images 64 x 64 pixels; (b)Hand images project on PCA space.

$$X^* = \{x^*_{i,j}\}; x^*_{i,j} = \frac{x_{i,j} - g_j}{\sqrt{n}\sigma_j}; g^*_j = \frac{\sum x_{i,j}}{n} \tag{6}$$

σ_j is standard deviation in column j of X matrix, n is rows number of X matrix, g_j is average value of column j of X matrix. Hand images X^* as (7) is reshaped into matrix one row and 64×64 columns (Y) matrix as (8):

$$X^* = \begin{bmatrix} x^*_{1,1} & x^*_{1,2} & ... & x^*_{1,64} \\ x^*_{2,1} & x^*_{2,2} & ... & x^*_{2,64} \\ . & . & ... & . \\ . & . & ... & . \\ . & . & ... & . \\ x^*_{64,1} & x^*_{64,2} & ... & x^*_{64,64} \end{bmatrix} \tag{7}$$

$$=> Y = \begin{bmatrix} q^1 & q^2 & ... & q^{4096} \end{bmatrix} \tag{8}$$

Using PCA to reduce correlation data between component from Y helps to reduce computational workload and still enough information that will be implement in training phase and testing phase. A training hand gesture includes M hand postures: $G_i = [Y_{i1}, Y_{i2}, ..., Y_{iM}]^T$. A training hand gesture set includes N hand gestures: $G = [G_1, G_2, ..., G_N]^T$. This set is input for the PCA algorithm and create feature results. In this research, PCA space is set to 20 dimensions. Feature space created from PCA includes: μ covariance matrix of Y_i vector in G_j, eigenvalues λ, eigenvectors e and H^* is projection of H. Given a testing gesture G_k consisting of n postures $Y_{ki}(i = [1, n])$, as shown in Fig. 9(a), G_k is transformed into the PCA space, as illustrated in Fig. 9(b):

4.4.2 Time normalization

As described in Sec.3, each subject could perform one gesture in different length/duration. That leads to different number of postures/frames of gestures in one class. In addition, states of gestures often are non-synchronization phase (e.g., hand closing and hand opening states). As consequent, phase synchronization between two dynamic hand gestures is necessary. In this work, we use Dynamic Time Warping (DTW) technique due to its efficiency and simplicity to deploy in real-time. DTW technique aims to match two time series of different length with some constraints. Two gesture sequences are stretched non-linear along the time axis in order to determine the similarity between them. The simplest version of DTW can be implemented as follows: Let G_1 and G_2 two hand gestures $G_1\{X_{11}, X_{12}, ..., X_{1n}\}$, $G_2\{X_{21}, X_{22}, ..., X_{2m}\}$, m and n are their length respectively. DTW will indicate that each X_{1i} of G_1 is matched with X_{2j} of G_2. This synchronization gives results as shown in Fig. 12(a). The DTW results will be utilized as an input

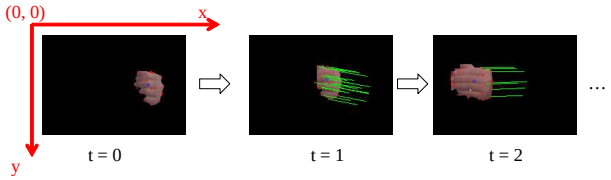


Figure 10: Optical flow of the go-left hand gesture.

for the temporal hand gesture step.

4.5 Temporal representation of gesture

4.5.1 *Hand gesture trajectory*

Many proposed methods for action recognition in the last years reply on temporal features. In our work, hand movement trajectory is extracted based on KLT (Kanade-Lucas-Tomasi) technique. This technique combines the optical flow method of Lucas-Kanade[16] and the good feature points segmentation method of Shi-Tomasi[11]. This algorithm determines optical flow of Lucas-Kanade that based on three assumptions: the invariance of light intensity, the movement of hands in two consequence frames is small and Cohesion of space (the neighboring points on the same surface of the hand is the same motion). KLT helps to describe trajectory of feature points of hand or calculates optical flow of hand between two sequence postures as shown in Fig.10. At the first frame of hand gesture, feature points of hand posture will be detected and tracked in the next frame until the end posture of gesture. The connected feature points through hand postures creates a trajectory. Our research selects the 20 most significant trajectories to represent a hand gesture that result is illustrated in (10). Each trajectory is composed by K points $\{P_1, P_2, ..., P_K\}$. Each point P_i has coordinates (x_i, y_i). Taking average of all points gives a average trajectory $T = [\overline{p^1_{i,j}}, \overline{p^2_{i,j}}, ..., \overline{p^K_{i,j}}]$. This average trajectory represents hand directions of a gesture. Fig .11. illustrates trajectories of 20 feature points and the average trajectory of the Next command in spatial-temporal coordinate. Red circles present feature points coordinates P_i at frame i ($i = [0, n-1]$). Blue square represents the average $\overline{P_i}$.

4.5.2 *Similarity between two hand gestures*

Each hand gesture is represented by an average trajectory T. As the coordinates of key-points (x, y) on image are different, we normalize into T^*:

$$T = [\overline{p^1_{i,j}}, \overline{p^2_{i,j}}, ..., \overline{p^n_{i,j}}] \quad (9)$$

$$T^* = [\overline{p^1_{i,j}} - (\overline{x}, \overline{y}), \overline{p^2_{i,j}} - (\overline{x}, \overline{y}), ..., \overline{p^n_{i,j}} - (\overline{x}, \overline{y})] \quad (10)$$

In hand gesture recognition, a correlation between training set is P and a test set is T is estimated by RMSE (Root-Mean Square Error) distance which calculates an error at $p_{i,j}$ of P and $q_{i,j}$ of T, as defined by following (11) equation:

$$RMSE = \sqrt{\frac{\sum^{i=1} (p_i(x,y) - q_i(x,y))^2}{n}} \quad (11)$$

In practice, length of T and P are not the same therefore directly calculating RMSE is not trivial. In other words,

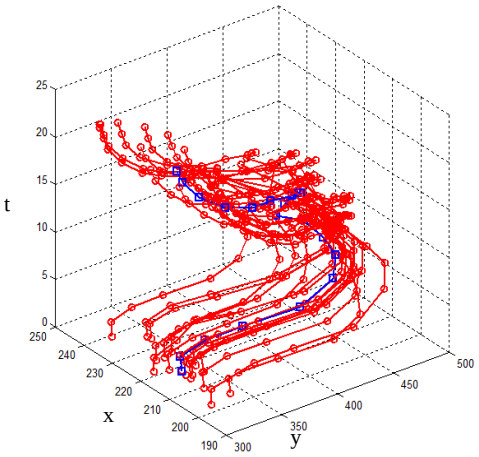


Figure 11: Trajectory of the go-right hand gesture.

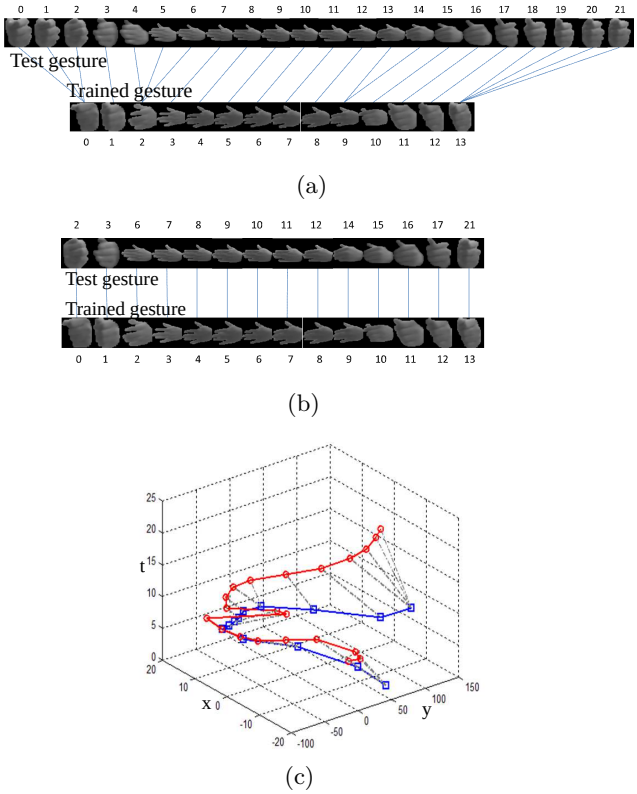


Figure 12: All trajectory link of two hand gesture (T, P) (a)DTW results,(b)DTW results removed repetition links, (c)RMSE results.

mapping frames between T and P can be one-to-many in common cases. Thanks to the link between postures (T, P) that comes from DTW results, we can eliminate hand shapes (or frames) which appear several times in P and only one time in T. For example, as shown in Fig .12(a) they are some frames at initial and ending states of P. After removing those frames, as shown Fig .12(c), correspondences frames between T and P is mapping one-to-one. This expression can be shown through links of two trajectories T and P in Fig .12(b), in which the blue path is trajectory of T template gesture, the red path is trajectory of P testing gesture, the gray line are connected-link between a state at T and the corresponding sate at P. Therefore, estimating RMSE will be trackable (Fig .12(b)). The experimental results in Section 5 will show that the using RMSE is simple and obtaining a high accuracy of the recognition rate. If a RMSE values is small, two gestures (T, P) is more similar. Based on RMSE distance, a K-NN classifier is utilized to vote K nearest distances from template gestures. A label is assigned to a testing gesture based on maximal number of a label from K.

5. EXPERIMENTAL RESULTS

The proposed framework is warped in a C++ program on a PC Core i5 3.10GHz CPU, 4GB RAM. A MS Kinect sensor [1] is mounted on a tripod at fixed position. The Kinect sensor captures data at 20 fps. Fine defined gestures are captured by five volunteers (3 males, 2 females) who are asked to implement the evaluations at different scenarios, lighting conditions and the complex background. Each person implements one command in five times. Our dataset saved and labeled by a template below (12):

$$[P + order\_number]\_[L + times]\_[G + class\_label] \quad (12)$$

$[P+order\_number]$ is $\underline{P}$erson $\underline{O}$rder $\underline{N}$umber. $[L+times]$ is times number. $[G + class\_label]$ is $\underline{G}$ gesture labels. For example, a gesture $P_1\_L_1\_G_4$ means that it is the first person implemented at the first time with increase command (gesture label is 4). Consequently, in our dataset, each command/gesture are collected in thirty times. We present two evaluations: (a) detection and segmentation of the dynamic hand gesture recognition, and (b) hand gesture recognition. Experimental results are shown in following section.

5.1 Gesture spotting results

We conduct the experiments with video data of six subjects. We then evaluate results in term of true and false alarm rate of the gesture spotting from videos. Each video has 25 gestures as defined and artifacts (true negative) because a video could include both the meaningless gestures and the meaning ones. We then calculate TP (True Positive), TN (True Negative), FP (False Positive), FN (False Negative) by comparing the detected gestures and the ground-truth data, as presented in Table 1 below:

Averagely, Table 1 infers accuracy rate $\frac{TP}{TP+FN}$ of the true dynamic hand gesture detected at 96.1 ± 3.1 (%) and false alarm rate $\frac{FP}{TP+FP}$ at 5.7 ± 5.0 (%).

5.2 Gesture recognition results

We evaluate performance of the gesture recognition for five dynamic hand gestures. For preparing training and testing data, we follow "$Leave - p - out - cross - validation$"

Table 1: Spotting gesture result

Subject \ Attribute	TP	FP	FN	TN
1	25	1	0	2
2	25	0	0	1
3	23	1	2	2
4	26	2	2	1
5	24	1	1	3
6	23	4	1	0

method, in which p equals to 5. Then evaluation results are given in table 2 that presents confusion matrix of then recognized gestures. Averagely, the accuracy rate obtains 95.2 ± 2.04 %. Recognizing dynamic hand gestures required a computational time of 167 ± 16 ms .

Table 2: Confusion matrix gesture recognitions

Testing \ Training	G1	G2	G3	G4	G5
G1	47	0	0	1	2
G2	2	48	0	0	0
G3	2	0	48	0	0
G4	1	0	0	49	0
G5	2	1	1	0	46

6. CONCLUSIONS

This paper described a vision-based hand gesture recognition system. Our work was motivated by deploying a feasible technique into HCI applications, e.g., to control devices/alliances in a smart home. We proposed a new type of hand gestures which map common commands to hand gestures. The proposed gestures help to conveniently detect and separate user's command from a video stream. Regarding the recognition issue, we attempted both spatial-temporal characteristics of a gesture. The experimental results confirmed that accuracy of recognition rate approximates 95% with computational time approximately 167 ms/frame. Therefore, it is feasible to implement the proposed system to control home alliances.

Our proposed system has a few limitations. First, current results could not confirm that new type of gestures are convenient for user performing, particularly, to end-users who need a natural and easy way to control devices. Second, Kinect-based hand gestures recognition techniques recently archive significant results. However, such works have not been listed and compared with the proposed technique. Third, we need to evaluate the proposed technique in other scenarios to confirm its performance. For example, the evaluations in future should include the clustered background; adjustable distances from Kinect to end-user; or changing direction of end-user to device. Fourth, our database is small and we will evaluate with our lager database in the next time. Consequently, these limitations suggest us directions to future works.

7. ACKNOWLEDMENT

The research leading to this paper was supported by HUST projects under grant number T2015-095 and T2015-96. We would like to thank the projects and people involved in these projects.

8. REFERENCES

[1] http://www.microsoft.com/en-us/kinectforwindows.

[2] I. Bayer and T. Silbermann. A multi modal approach to gesture recognition from audio and video data. *Proceedings of the 15th ACM on International conference on multimodal interaction*, pages 461 – 466, 2013.

[3] Q. Chen, C. Joslin, and Georganas.N.D. A Dynamic Gesture Interface for Virtual Environments Based on Hidden Markov Models. In *Proceedings of IEEE International Workshop on Haptic Audio Visual Environments and their Applications*, 2005.

[4] X. Chen and M. Koskela. Online rgb-d gesture recognition with extreme learning machines. *ACM on International conference on multimodal interaction*, pages 467 – 474, 2013.

[5] P. Company. PointGrab brings gesture control to home appliances, 2013.

[6] X. Deyou. A Network Approach for Hand Gesture Recognition in Virtual Reality Driving Training System of SPG. *International Conference on Pattern Recognitioin*, pages 519–522, 2006.

[7] M. Elmezain, A. Al-Hamadi, and C. Michaelis. Real-Time Capable System for Hand Gesture Recognition Using Hidden Markov Models in Stereo Color Image Sequences. *Journal of WSCG*, 16:65–72, 2008.

[8] S. Escalera, J. Gonz'alez, X. BarÂt'o, M. Reyes, O. Lopes, I. Guyon, V. Athitsos, and H. Escalante. Multi-modal gesture recognition challenge 2013: Dataset and results. *Proceedings of the 15th ACM on International conference on multimodal interaction*, pages 445 – 452, 2013.

[9] C. L. Huang and S.-h. Jeng. A model-based hand gesture recognition system. *Machine vision and applications*, pages 243–258, 2001.

[10] D. Huong-Giang, V. Hai, T. Thanh-Hai, and C. Eric. Improvements of rgb-d hand posture recognition using an user-guide scheme. In *Proceeding(s) of the 7th IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and the 7th IEEE International Conference on Robotics, Automation and Mechatronics (RAM)*, 2015.

[11] J.Shi and C.Tomasi. Good features to track. In *Proc. International Joint Conference on Artificial Intelligence*, pages 593–600, 1994.

[12] D. Kim, J. Song, and D. Kim. Simultaneous Gesture Segmentation and Recognition Based on Forward Spotting Accumlative HMMs. *Journal of Pattern Recognition Society*, 40:1–4, 2007.

[13] T.-K. Kim and R. Cipolla. Canonical Correlation Analysis of Video Volume Tensors for Action Categorization and Detection. In *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, pages 1415–1428, 2009.

[14] A. Kurakin, Z. Zhang, and Z. Liu. A real time system for dynamic hand gesture recognition with a depth. In *20th European Signal Processing Conference-EUSIPCO*, pages 27–31, August 2012.

[15] Y.-t. Li and J. P. Wachs. Hierarchical Elastic Graph Matching for Hand Gesture Recognition. *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications. Lecture Notes in Computer Science*, 7441:308–315, 2012.

[16] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proc. International Joint Conference on Artificial Intelligence*, page 674ï£¡679, 1981.

[17] A. I. Maqueda, c. del Blanco, and N. Jaureguizar, F García. Human-computer interaction based on visual recognition using volumegrams of local binary patterns. In *IEEE International Conference on Consumer Electronics*, pages 583–584, Jan 2015.

[18] S. Marcel, O. Bernier, J.-E. Viallet, and D. Collobert. Hand gesture recognition using input-output hidden Markov models. In *Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580)*, number Figure 1, pages 456–461, 2000.

[19] C. Oz and C. Leu. American sign language word recognition with a sensory glove using artificial neural networks. *Engineering Applications of Artificial Intelligence*, 24(7):1204–1213, 2011.

[20] P. Qifan, S. Gupta, S. Gollakota, and S. Patel. Whole-Home Gesture Recognition Using Wireless Signals. In *Proceedings of the 19th Annual International Conference on Mobile Computing and Networking*, 2013.

[21] S. Rautaray and A. Agrawal. Vision based hand gesture recognition for human computer interaction: a survey. *Artificial Intelligence Review*, Nov. 2012.

[22] Z. Ren, J. Yuan, and Z. Zhang. Robust Hand Gesture Recognition Based on Finger-Earth MoverâĂŹs Distance with a Commodity Depth Camera. In *Proceedings of the 19th ACM international conference on Multimedia*, 2011.

[23] X. Shen, G. Hua, L. Williams, and Y. Wu. Dynamic hand gesture recognition: An exemplar-based approach from motion divergence fields. *Image and Vision Computing*, 30(3):227–235, Mar. 2012.

[24] C. Stauffer and W. Grimson. Adaptive background mixture models for real-time tracking. In *Proceedings of Computer Vision and Pattern Recognition*, 1999.

[25] K. Takahashi, S. Sexi, and R. Oka. Spotting Recognition of Human Gestures From Motion Images. *In Technical Report IE92-134*, pages 9–16, 1992.

[26] H. Yang, S. Scharoff, and S. Lee. Sign Language Spotting with a Threshold Model Based on Conditional Random Fields. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 31:1264–1277, 2009.

[27] X. Zabulis, H. Baltzakis, and A. Argyros. *Vision-based Hand Gesture Recognition for Human Computer Interaction*. Lawrence Erlbaum Associates, 2009.