HAND GESTURE SPOTTING USING HIDDEN MARKOV MODELS

Hyeon-Kyu Lee and Jin-Hyung Kim
Department of Computer Science, KAIST, Taejon, Korea

KEY WORDS: Gesture spotting, Hidden Markov Model, threshold model, internal segmentation

ABSTRACT

This paper proposes a new gesture spotting method that extracts gestures from hand motions. The proposed approach is based on the HMM which can solve segmentation problem and can absorb spatio-temporal variance of gestures. To remove non-gesture patterns from input patterns, we introduce the threshold model that helps to qualify an input pattern as a gesture. The proposed approach extracted gestures from hand motions with 94.9% reliability.

1. INTRODUCTION

Gesture is a subspace of human motions expressed by the body, the face, or hands. Among a variety of gestures, hand gestures are the most expressive and the most frequently used. The hand gestures have been studied as an alternative interface between human and computer by several researchers including Quek [Quek, 1994], Freeman [Freman et al., 1995], Starner [Starner et al., 1995], Kjeldsen [Kjeldsen et al., 1995], and Takahashi [Takahashi et al., 1992]. In this paper, we define a gesture to be a motion of the hand to communicate with a computer.

The technique of extracting meaningful segments from unpredictable input signals and recognizing them is called “pattern spotting”. Gesture recognition is an instance of pattern spotting applications as it has to locate the start and the end point of a gesture. Hereinafter, we denote by “gesture spotting” the gesture recognition with the extraction of gesture patterns.

The gesture spotting has two major difficulties: segmentation and spatio-temporal variances. The segmentation problem is to determine when a gesture starts and when it ends from a hand trajectory. As the gesturer switches from one gesture to another, the hand passes through many intermediate positions located between the two gestures. The recognizer may attempt to recognize this unconscious immediate motion as intended. Without segmentation, the recognizer should try to match a gesture with all possible segments of input signals. Another difficulty in gesture recognition is that even the same gesture varies in shape and duration depending on gesturers; it also varies instance by instance even for the same gesturer. Therefore, the recognizer should consider the spatial and temporal variances simultaneously.

We choose the HMM approach for the gesture spotting because it can represent non-gesture patterns that are crucial to hand motions and can reflect spatio-temporal variance very well. It has been the most successful and widely used approach to model events which have spatio-temporal variances [Huang et al., 1990]. Particularly, it has been successfully applied in online hand-writing recognition [Lee et al., 1995] and speech recognition [Wilcox, 1992] areas.

The HMM can suggest multiple candidates because it estimates the similarity of an input

pattern with a reference pattern. Also, the matching process of the HMM does not require additional consideration for reference patterns with spatial and temporal variances because they are internally represented as probabilities of each state and transition. In addition, if the set of unknown patterns is finite, the HMM can represent unknown patterns using a garbage model that can be trained with the unknown patterns.

However, there are some limitations in representing non-gesture patterns using HMM. In pattern spotting, reference patterns are defined by keyword models and unknown patterns are defined by a garbage model. The garbage model is trained using data within a finite set (character set, voiced word set, etc.). In gesture spotting, however, it is not easy to train the garbage model that can best match non-gesture (i.e., unknown) patterns because the set of non-gesture patterns is not finite. To overcome this, we utilize the internal segmentation property of the HMM and introduce the threshold model that consists of states in trained gesture models and helps to qualify the matching results of gesture models.

To evaluate the performance of the threshold model based on the HMM, we construct a test-bed system called the PowerGesture system with which we can browse the slides of PowerPoint™ using gestural commands. In experiments, the proposed approach showed 94.9% reliability, and spotted gestures at a 5.8 frames per second rate.

The remainder of this paper is organized as follows. In Section 2, we describe the details of the threshold model and the end-point detector. Experimental results are provided in Section 3, and concluding remarks are given in Section 4.

2. GESTURE SPOTTING

2.1. The Threshold Model

The internal segmentation property implies that states and transitions in trained HMM represent sub-patterns of a gesture, and a sequential order of sub-patterns implicitly. With this property, we may construct a new model that can match new patterns generated by combining sub-patterns of a gesture in a different order. Furthermore, by constructing a fully connected ergodic model using states in a model, we may construct a model which can match all patterns generated by combining sub-patterns of a gesture in any order.

We constructed gesture models in the left-right model, and re-estimated the parameters of each model with the Baum-Welch algorithm. Then, a new ergodic model was constructed by removing all outgoing transitions of states in all gesture models and fully connecting the states. In the new model, each state can reach all other states in a single transition. Probabilities of each state and its self-transition in the new model remain the same as in gesture models, and probabilities of outgoing transitions are equally assigned using the fact that the sum of all transition probabilities is 1.0 in a state. **Figure 1** shows the threshold model that includes two null states, ST and FT, that consume no observations.

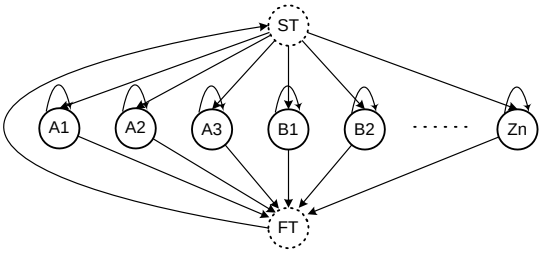


Figure 1: The threshold model.

Maintaining the probabilities of states and their self-transitions makes the new model represent all sub-patterns of reference patterns, and

constructing an ergodic model makes it match well with all patterns generated by combining sub-patterns of reference patterns in any order. Nevertheless, a gesture can best match a gesture model because the outgoing transition probability of the new model is smaller than that of the gesture model. Therefore, the output of the new model can be used as an adaptive threshold for that of a gesture model. For this reason, we call the model a "threshold model".

After training gesture models and creating the threshold model, we constructed a gesture spotting network (GSN) for spotting gestures from continuous hand motions, as shown in **Figure 2**. In the figure, S is the null start state.

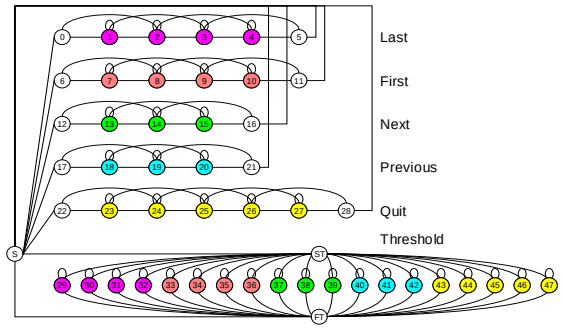
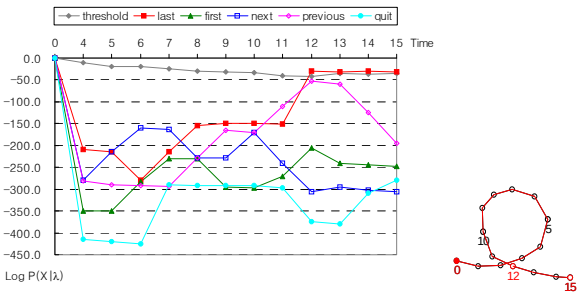


Figure 2: Gesture Spotting Network.

When the likelihood of a gesture model is greater than that of the threshold model, that point is considered as a candidate end point. The start point can be easily found by backtracking the Viterbi path. This is because that the final state can only be reached through the start state in the left-right HMM.

2.2. End-point detector

According to observing the likelihood of individual models, the threshold model is usually the best matched model. However, as the forward path gets close to the end of a gesture, the corresponding gesture model becomes the most likely, as shown in **Figure 3** (time=12). In the figure, the likelihood of the "last" model is lower than that of the threshold model before the time reaches to 12. However, the likelihood becomes greater than that of the threshold model from time=12. All points between time=12 and time=15 are candidate end points of the "last" gesture. The start point can be found by backtracking Viterbi path of each candidate.



(a) (b)

Figure 3: (a) Likelihood graph, (b) input pattern.

The end-point detector finds the best candidate end point from candidates. The detection criteria of the end-point detector are defined by a heuristic that uses the pattern immediately following the candidate point. The heuristic is as follows:

(1) When the immediately following pattern is not a gesture, the detector chooses the last candidate as the end point.
(2) When the immediately following pattern is a gesture, the detector has two alternatives:
  (a) When the start point of the next gesture precedes the first candidate end point, the detector regards current gesture as part of a large next gesture that extends beyond the end point of the current gesture and ignores all the candidate end points.
  (b) When the next gesture starts between the first and the last candidate end points of the

current gesture, the detector chooses the last candidate as the end point.

In addition, the end-point detector has to catch the gesturer's intention to complete a gesture for an immediate response. The gesturer's intention for gesture completion may be represented by moving the hand beyond the camera range or by not moving the hand for a while. Once the end-point detector catches the intention, it stops the spotting process and retrieves a gesture, if one exists.

3. EXPERIMENTAL RESULTS

To evaluate the performance of the threshold model based on the HMM, we constructed the PowerGesture system with which we could browse the slides of PowerPoint™ using gestural commands, as shown in **Figure 4**.

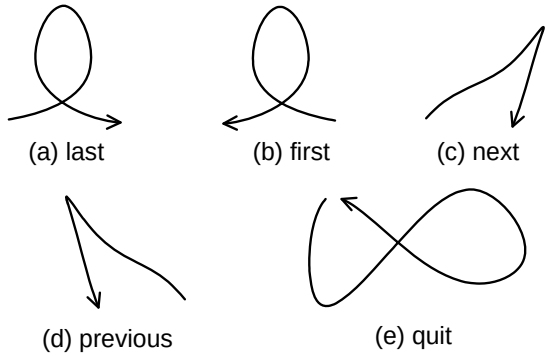


Figure 4: Gestures used.

The current gesture spotting system is integrated into the hypermedia presentation system - PowerGesture - with a gestural interface that captures image frames of hand motion from a camera, interprets them using the proposed spotting method and controls the browsing of slides. **Figure 5** shows the block diagram of the PowerGesture; it is built on a Pentium Pro PC with a Windows 95 operating system.

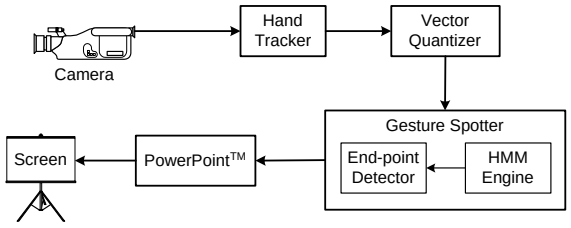


Figure 5: Block diagram of the PowerGesture.

The hand tracker converts an RGB color image captured from a camera to a YIQ color image because the I-component in YIQ color space is sensitive to skin color. Then, it thresholds the I-component image to produce a binary image, and extracts objects using one-pass labeling algorithm [Ko et al., 1996]. For simple image processing, we adopted a uniform background and restricted hand motions with only the right hand.

3.1. Threshold Model Test

We collected 1250 isolated gestures and trained gesture spotter using the data set in **Table 1**. The success of our gesture spotter greatly depended on the discrimination power of the gesture models and the threshold model. For this, we carried out an isolated gesture recognition task. The majority of misses were caused by the disqualification effect of the threshold model, which rejected some gestures due to the lower likelihood of the target gesture model than that of the threshold model.

	Train Data	Test data	correct	delete	recognition (%)
last	196	54	54	0	100.0
first	195	55	55	0	100.0
next	198	52	51	1	98.1
prev	195	55	54	1	98.2
quit	202	48	45	3	93.8
Total	**986**	**264**	**259**	**5**	**98.1**

Table 1: Training data and test results.

3.2. Gesture Spotting Test

The second test concerns evaluating the spotting

capability of the gesture spotter using gesture and threshold models of the previous experiment. We collected 30 test data set for this experiment. Each test sample is a sequence of 200 image frames that contains more than one gesture.

In gesture spotting task, there are three types of errors: an insertion error, which occurs when the spotter misclassifies a non-gesture as a gesture; a deletion error, which occurs when the spotter misses a gesture; and a substitution error, which occurs when the spotter misclassifies a gesture as another gesture. The insertion error is not considered in calculating the detection ratio of the gesture spotter. However, the insertion error can cause deletion or substitution errors because it seems to force the end-point detector to remove some or all of the true gesture from observation. We use the reliability measure considering the insertion error as **Equation 1** shown below.

$$\text{Reliability} = \frac{\text{Correct Recognition}}{(\text{Gestures} + \text{Insertion Error})} \qquad (1)$$

We count errors by varying the model transition probability towards the threshold model ($p(TM)$) as shown in **Figure 6**. In the figure, as $p(TM)$ decreases between 1.0 and 0.1, the deletion error sharply decreases. However, as $p(TM)$ passes 0.1, the deletion error keeps slowly increasing. We suspect that this is because the increase of the insertion error causes the deletion error to increase. The deletion error directly affects the detection ratio, while the insertion error does not. However, it should be noted that many insertion errors are not totally independent of the detection ratio because some insertion errors cause deletion or substitution errors. **Table 2** shows the result of experiment where the model transition probability towards the threshold model is 0.1.

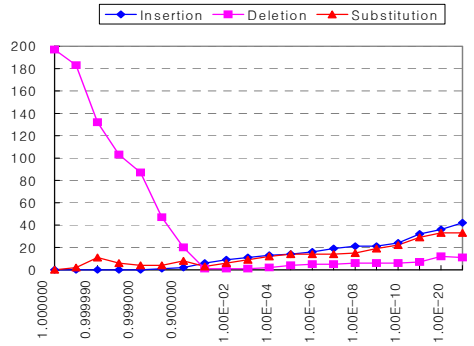


Figure 6: Number of errors according to $p(TM)$.

	# of gesture	Results				
		insert error	delete error	substitute error	correct	reliability
last	38	2	0	0	38	95.0
first	47	3	0	1	46	97.9
next	56	1	0	1	55	98.2
prev	36	0	1	1	34	94.4
quit	21	0	0	0	21	100.0
total	**198**	**6**	**1**	**3**	**194**	**94.9**

Table 2: Spotting result ($p(TM) = 0.1$).

4. CONCLUDING REMARKS

In this paper, we proposed the threshold model. With the model, we could process 5.8 frames/second, and spotted gestures with 94.9% reliability. Experimental results demonstrated that the threshold model was simple but highly effective in qualifying the input pattern as a target gesture.

REFERENCES

- Freeman, W.T., Weissman, C.D. (1995): Television control by hand gestures, *Proc. of 1st IWAFGR*, 179-183, 1995.
- Huang, X.D., Ariki, Y. and Jack, M.A. (1990): Hidden Markov Models for Speech Recognition, *Edinburgh Univ. Press*, 1990.
- Kjeldsen, R., Kender, J. (1995): Visual hand Gesture Recognition for Window System

Control, *Proc. of 1st IWAFGR*, 184-188, 1995.

- Ko, I. and Choi, H. (1996): Hand Region Detection by Region Extraction and Tracking, *Proc. of 23rd Korea Information Science Society Fall Conference*, 239-242, 1996.
- Lee, S., Lee, H. and Kim J. (1995): On-Line Cursive Script Recognition Using an Island-Driven Search Technique, *Proc. of ICDAR*, 886-889, 1995.
- Quek, F. (1994): Toward a Vision-based Hand Gesture Interface, *Proc. of VRST*, 17-31, 1994.
- Starner, T. and Pentland, A. (1995): Real-Time American Sign Language Recognition from Video Using Hidden Markov Models, *TR-375*, MIT Media Lab, 1995.
- Takahashi, K., Seki, S. and Oka, R. (1992): Spotting Recognition of Human Gestures from Motion Images, Technical Report IE92-134, *The Institute of Electronics, Information and Communication Engineers(Japan)*, 9-16, 1992.
- Wilcox, L.D. and Bush, M.A. (1992): Training and Search Algorithms for an Interactive Wordspotting System, *Proc. of ICASSP*, Vol II, 97-100, 1992.