PREDICTION-BASED GESTURE DETECTION IN LECTURE VIDEOS BY COMBINING VISUAL, SPEECH AND ELECTRONIC SLIDES

Feng Wang
Dept. of Computer Science
Hong Kong University of
Science and Technology

Chong-Wah Ngo
Dept. of Computer Science
City University of
Hong Kong

Ting-Chuen Pong
Dept. of Computer Science
Hong Kong University of
Science and Technology

ABSTRACT

This paper presents an efficient algorithm for gesture detection in lecture videos by combining visual, speech and electronic slides. Besides accuracy, response time is also considered to cope with the efficiency requirements of real-time applications. Candidate gestures are first detected by visual cue. Then we modifity HMM models for complete gestures to predict and recognize incomplete gestures before the whole gestures paths are observed. Gesture recognition is used to verify the results of gesture detection. The relations between visual, speech and slides are analyzed. The correspondence between speech and gesture is employed to improve the accuracy and the responsiveness of gesture detection.

1. INTRODUCTION

Due to the popularity of e-learning, lecture videos are widely available for online access. To support effective browsing and search, these videos need to be properly captured, indexed and edited. Traditionally, this work is mostly operated by expert cameramen and editors. This procedure is costly and requires manual work. Recently, numerous attempts have been made to automate the multimedia authoring of live presentations. These efforts include off-line video editing [1] and real-time broadcast by automatic camera management [2, 3]. One desired goal is to correctly predict what presenters want to highlight, at any moment, and produce videos with appropriate views. This goal requires effective recognition of video content such as the speech, gesture and posture of a presenter.

Deictic gestures are used by almost all lecturers to direct the students' attentions to something that the lecturer is talking about. Gesture is therefore a reliable cue to estimate the focus of the lecture. Gesture detection has been employed in off-line video editing [1] and real-time lecture shooting [2, 3]. These algorithms are mostly based on visual cue such as skin-color and frame difference. A gesture is detected when the lecturer's hand enters the defined regions (an LCD projected screen or a whiteboard).

During a lecture, the lecturer can move freely in front of the class and thus the hand may interact with the slide even when the lecturer does not intend to point to anything in the slide. Here we define a gesture as an intentional interaction between the lecturer's hand and some object instances (*e.g.* a paragragh or a figure) in the slide. A problem for gesture detection is how to distinguish a gesture and non-gesture movement. In [6], we proposed an approach for gesture detection in lecture videos for offline video editing. Skin-color and frame difference are employed to detect candidate gestures, which are then recognized as three kinds of gestures: *lining*, *circling* and *pointing*. By gesture recognition we extract meaningful gestures and eliminate non-gesture movement. A gesture is detected only when it is verified by gesture recognition.

In this paper, we address gesture detection in lecture videos for real-time applications. The main difference between this work and [6] lies in two aspects. Firstly, to cope with the efficiency requirements of real-time applications, besides the accuracy of gesture detection, the response time is considered as another important criterion. For example, in an automatic camera management system for lecture capture, when a gesture is present, it is expected to focus the camera on the region interacting with the gesture as soon as possible so that the interaction can be captured. In [6], for off-line video editing, a gesture is verified when the gesture is finally completed, which is too late for the camera to respond to the gesture. In this paper, we propose an algorithm to detect and predict the gestures before the complete gesture paths are observed so as to verify the gestures earlier. Secondly, we combine different cues including visual, speech and electronic slides to improve the accuracy and reduce the response time.

2. GESTURE DETECTION BY VISUAL CUE

2.1. HMM for Complete Gesture Recognition

HMM has been proven to be useful in sign gesture recognition [4]. A detailed tutorial on HMM can be found in [7]. In [6], we employ HMM to recognize the deictic gestures in lecture videos. Based on gesture tracking by employing frame difference and skin-color detection, three HMM models are trained to recognize the three defined gestures. Given an observation $O = (o_1, o_2, \ldots, o_T)$, where each $o_i (i = 1, 2, \ldots, T)$ is a sampled point on the gesture path, gesture recognition prob-

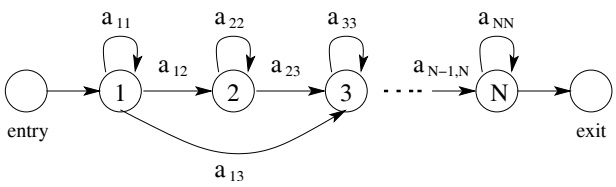

Fig. 1. The HMM Model for gesture recognition

lem can be regarded as that of computing

$$\arg\max_i\{P(g_i|O)\} \tag{1}$$

where g_i is the *i*-th gesture.

In HMM based gesture recognition, it is assumed that the observation sequence corresponding to each gesture is generated by a Markov model as shown in Fig 1. A Markov model is a finite state machine which changes state once for every observation point. In [7], by using Bayes' Rule and approximation, Equation 1 is deduced to calculating the likelihood

$$\tilde{P}(O|M) = \max_X\{\prod_{t=1}^{T} b_{x_t}(o_t)a_{x_t x_{t+1}}\} \tag{2}$$

where $X = x_1, x_2, \ldots, x_T$ is the state sequence that O moves through the model M.

In HMM models, a_{ij} and $b_i(.)$ are estimated by Baum-Welch algorithm in the training phase. For gesture recognition, Equation 2 is calculated by employing the Viterbi algorithm. The details of the two algorithms can be found in [7].

2.2. Modified HMM for Incomplete Gesture Recognition

The HMM models introduced in Section 2.1 are used for gesture recognition in offline systems. Each gesture is recognized when the complete path is observed. In this paper, we address gesture recognition for online applications, *e.g.* automatic camera control for lecture capture. For real-time processing, besides accuracy, the response time is an important criterion. To respond to the lecturer's gestures as soon as possible, gesture recognition and verification are expected to be carried out before the gesture is finally completed. We modify the HMM models in Section 2.1 to predict incomplete gestures.

Different from complete gestures, an incomplete gesture usually cannot move through the HMM model to the state *exit* in Figure 1, but just reaches one of the intermediate states $1, 2, \ldots, N$. In the HMM model topology shown in Figure 1, the final non-emitting state *exit* can be reached only through the last state N, which means a complete gesture must reach the state N before it is completed. To recognize incomplete gestures that may stop at any intermediate state, we modify the HMM models by adding a forward state transition from each intermediate state to the state *exit* in Figure 2. The joint

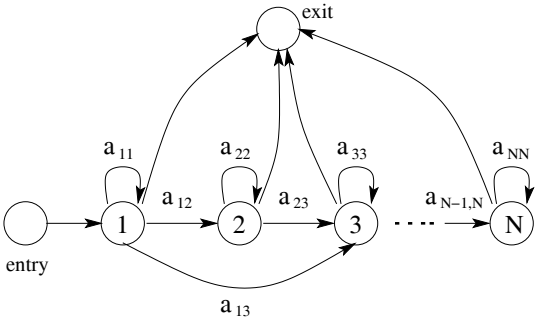

Fig. 2. The HMM Model for incomplete gesture recognition

probability that O moves through an HMM model M and stops at an intermediate state S can be approximated by

$$\tilde{P}(O|M,S) = \max_X\{\prod_{t=1}^{S} b_{x_t}(o_t)a_{x_t x_{t+1}}\} \tag{3}$$

where x_{S+1} is the *exit* state.

Figure 3 shows the trajectory of a circling gesture. (c) is the complete gesture, and (a), (b) are two incomplete gestures stopping at different intermediate states. By comparing (a) and (b), we are more confident the current observation will compose a gesture if it moves further through the HMM model or stops at a state nearer to the state *N*. Based on this finding, we take into account the stopping state for the probability calculation and modify equation 3 to be

$$\tilde{P}^{'}(O|M,S) = \max_X\{(\prod_{t=1}^{S} b_{x_t}(o_t)a_{x_t x_{t+1}}) \cdot e^{\frac{S}{N}}\} \tag{4}$$

Let $a^{'}_{ij} = a_{ij}e^{\frac{j-i}{N}}$, and Equation 4 can be written as

$$\tilde{P}^{'}(O|M,S) = \max_X\{\prod_{t=1}^{S} b_{x_t}(o_t)a^{'}_{x_t x_{t+1}}\} \tag{5}$$

Thus, the Viterbi algorithm can still be used for the calculation of Equation 5. The probability that O is an incomplete gesture modeled by M is

$$\tilde{P}(O|M) = \max_S \tilde{P}^{'}(O|M,S) = \max_{X,S}\{\prod_{t=1}^{S} b_{x_t}(o_t)a^{'}_{x_t x_{t+1}}\} \tag{6}$$

Equation 1 can be solved by Bayes' Rule and assuming that

$$P(O|g_i) = \tilde{P}(O|M_i) \tag{7}$$

2.3. Gesture Verification by Recognition

Given an observation sequence O, for the three defined gestures, three confidence values are calculated that indicate how likely O will be a gesture g_i

$$C_i = P(g_i, O) \tag{8}$$

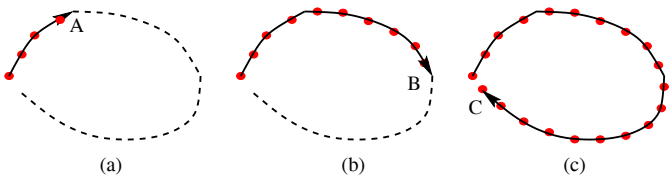

Fig. 3. (a)(b): incomplete circling gestures; (c) complete circling gesture

C_i values are used to verify whether O is an intentional gesture or not. A confidence value C_{visual} on the presence of a gesture is calculated by visual cue as

$$C_{visual} = \frac{C_{max}}{\sum_i C_i} \cdot C_{max} \tag{9}$$

where $C_{max} = \max_i C_i$. Given a threshold C_{thres}, a gesture is verified if $C_{visual} > C_{thres}$. C_{visual} is calculated for each sampled point. If $C_{visual} > C_{thres}$ is satisfied for an intermediate point on the gesture path, the gesture is verified before it is completed. In general, the further O moves through the corresponding HMM model, the more confident there is a gesture; however, the longer response time is required.

3. SPEECH IN GESTURE DETECTION

3.1. Relations Between Visual, Speech and Slides

In Figure 4, when a gesture is pointing to a paragraph in the slide, a keyword "*model*" is found in speech, which lies in the paragraph interacting with the gesture. We employ the correspondence between gesture and speech to improve the performance of gesture detection and reduce the response time.

As illustrated in Figure 4, electronic slides are used to find out the relationship between speech and visual. In [5], we synchronize videos with electronic slides by text analysis. The spatial correspondence between them is discovered by computing the homography [6]. After registration, the relationship between videos and the slides can be easily realized. In this work, the transcripts of speech are generated by an ASR (Automatic Speech Recognition) engine and matched with texts extracted from the slides. The video texts interacting with the gesture are obtained by the spatial correspondence between the video and the slide. Then gesture-speech correspondence can be detected by video text and speech matching.

3.2. Gesture-Speech Correspondence

As shown in Figure 4, if a candidate gesture is detected and a gesture-speech correspondence is found, there is more likely an intentional gesture present. A confidence value is calculated based on this correspondence.

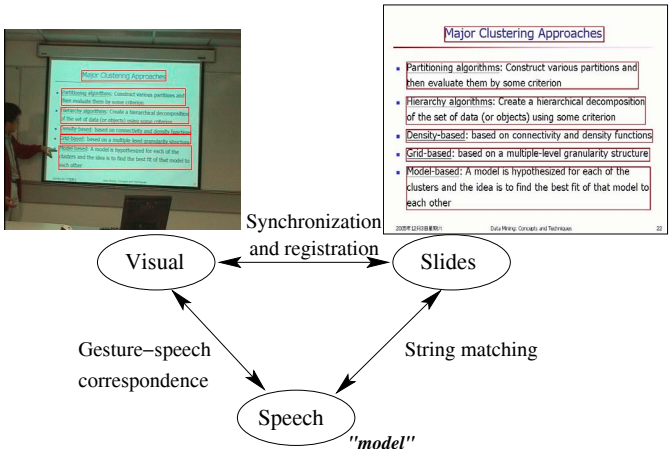

Fig. 4. Relations between different cues

With a PowerPoint slide as example in Figure 4, the paragraphs are semantically grouped into separate object instances. By constructing the one-to-one mapping between the instances in slides and vieos through homography projection, we can easily organize and structure the layout of videos. In Figure 4, the slide region in videos is partitioned into different ROIs (Region Of Interest). Each ROI is a unit that may be pointed by a gesture. The transcript of video texts in the ROIs can be obtained by the registered slides. Methods of stemming and stop word removal are first employed to the texts in each slide. By treating each ROI as a document, TF (Term Frequency) and IDF (Inverse Document Frequency) are computed for the remaining keywords to distinguish different ROIs. When such a gesture-speech correspondence is found, a confidence value C_{speech} is calculated based on the the TF and IDF values of the keywords as

$$C_{speech} = \sum_{w \in W} \frac{\log(1 + TF_w \cdot IDF_w)}{1 + \Delta t_w} \tag{10}$$

where W is the set of matched keywords, and Δt_w is the time interval between the presence of the gesture and of the keyword w in speech (If w is found during a gesture, $\Delta t_w = 0$).

3.3. Gesture Detection by Combining Speech and Visual

When a candidate gesture is detected in the slide region, we keep tracking it and calculating C_{visual}. At the same time, the transcripts of speech are generated by ASR. We search the keywords in the transcripts that match with the texts in the ROI interacting with the candidate gesture. Once a correspondence between speech and visual is detected, C_{speech} is then calculated. The confidence value C on a gesture's presence is calculated as $C = \lambda_1 C_{visual} + \lambda_2 C_{speech}$, where λ_1 and λ_2 are the weights of the two cues. A threshold for confidence value C_{thres} is defined to verify the presence of a gesture. If $C > C_{thres}$ is satisfied, a gesture is then detected.

4. EXPERIMENTS

We conduct experiments on 5-hour videos consisting of 15 presentations given by 10 lecturers and tutors. The presenters include 5 males and 5 females. The presentations are given in the classrooms and seminar rooms of different sizes, layouts and lighting designs. A stationary camera is used to capture the projected slide and the lecturer.

We employ MicroSoft Speech SDK 5.1 for speech recognition. The overall recognition rate is no more than 20% due to the environmental noise in audio track. However, an interesting point we found from experiments is that when a presenter tries to highlight something by a gesture, the corresponding keywords are usually also highlighted in speech. This results in better performance for keyword recognition in speech to analyze the gesture-speech correspondence.

For experiments, we set different response time as constraints. Given a gesture, within the required time after the gesture starts, if the confidence value $C > C_{thres}$, then the gesture is detected. In Table 1, two methods for gesture detection are compared. The first one just employs visual cue and verify gestures by modified HMM models described in Section 2.2. The second one combines visual, speech and electronic slides. For the parameter settings, λ_1, λ_2 represent the importance of visual and speech in gesture detection. Their values depend on the reliablity of the two cues. Considering the limitation of speech recognition, we set $\lambda_1 = 0.7$ and $\lambda_2 = 0.3$. Since we collected gesture paths of different lecturers in the HMM training phase, the calculated C_{visual} varies very little for different presenting styles. C_{speech} just depends on the performance of ASR, not the lecturers. Finally, a range of $0.4 - 0.6$ is acceptable for C_{thres}. A larger C_{thres} usually means higher *Recall* and *Precision* vaues, but longer response time. In our experiment, we set $C_{thres} = 0.45$.

As seen in Table 1, when visual cue is used, for a longer response time, more gestures can be correctly detected. By combining speech and slide texts with visual cue, within a given response time, the second method detects more gestures than the first one. In other words, it takes less time to detect more gestures and thus the average response time is reduced. Overall, more than 80% of all gestures can be correctly detected at the middle of the gesture paths.

As the efficieny of the algorithm is concerned, the gesture tracking and feature point sampling are carried out every 3 frames. This sampling rate is enough to depict the trajectories of most gestures. Gesture detection and recognition can process about 13 sampled frames per second, which is efficient enough for real-time applications given the sampling rate.

5. CONCLUSION

We have presented our algorithm for gesture detection in lecture videos. To cope with the efficiency requirements in real-time applications, response time is also considered besides the accuracy of detection. By the modified HMM models, we can recognize and verify intentional gestures before they are completed. Gesture-speech correspondence is employed to improve the accuracy and responsiveness of gesture detection. The experiments show that the algorithm can cope with the requirements of real-time applications.

Table 1. Results of gesture detection. (N_g: Total number of gestures; N_d: Number of gestures detected; N_f: Number of gestures falsely detected; N_m: Number of gestures missed; N_c: Number of gestures correctly detected. $Recall = \frac{N_c}{N_g}, Precision = \frac{N_c}{N_d}$)

N_g			2060			
Gesture duration	Average		4.62 sec			
	Standard Deviation		1.83 sec			
Method	Delay (sec)	0.6	1.2	1.8	2.5	4.0
Visual	N_d	449	736	1255	1783	1966
	N_f	217	320	412	435	308
	N_m	1828	1644	1217	712	402
	N_c	232	416	843	1348	1658
	Recall	0.11	0.20	0.41	0.65	0.80
	Precision	0.52	0.57	0.67	0.75	0.84
Visual + Speech + Slides	N_d	980	1391	2275	2252	2170
	N_f	376	501	533	447	319
	N_m	1456	1170	318	255	209
	N_c	604	890	1742	1805	1851
	Recall	0.29	0.43	0.85	0.88	0.90
	Precision	0.62	0.64	0.77	0.80	0.85

Acknowledgement

The work described in this paper was supported by the grants HIA01/02.EG04, SSRI99/00.EG11 and DAG01/02.EG16.

6. REFERENCES

[1] M. Gleicher and J. Masanz, "Towards Virtual Videography", *ACM Multimedia Conf.*, 2000.

[2] M. Onishi, K. Fukunaga, "Shooting the Lecture Scene Using Computer-controlled Cameras Based on Situation Understanding and Evaluation of Video Images," *Int. Conf. on Pattern Recognition*, 2004.

[3] Y. Rui *et. al.*,"Videography for Telepresentations", *Int. Conf. on Human Factors in Computing Systems*, 2003.

[4] T. Starner *et. al.*, "Real-time American Sign Language Recognition Using Desk and wearable computer-based video", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Dec 1998.

[5] F. Wang, C. W. Ngo & T. C. Pong, "Synchronization of Lecture Videos and Electronic Slides by Video Text Analysis," *ACM Multimedia Conf.*, 2003.

[6] F. Wang, C. W. Ngo & T. C. Pong, "Gesture Tracking and Recognition for Lecture Video Editing," *Int. Conf. on Pattern Recognition*, 2004.

[7] S. Young, "HTK: Hidden Markov Model Toolkit", Cambridge Univ. Eng. Dept. Speech Group and Entropic Research Lab. Inc., Washington DC, 1993.