A simplified viterbi matching algorithm for word partition in visual speech processing

Foo, Say Wei; Yong, Lian; Liang, Dong

2009

Foo, S. W., Yong, L., & Liang, D. (2009). A simplified viterbi matching algorithm for word partition in visual speech processing. Proceedings of 4th European Workshop on Image Analysis for Multimedia Interactive Services, 355-358.

<https://hdl.handle.net/10356/79871>

https://doi.org/10.1142/9789812704337_0065

A simplified viterbi matching algorithm for word partition in visual speech processing ©
copyright 2009 World Scientific Publishing Co. This conference's website is located at
http://eproceedings.worldscinet.com/9789812704337/9789812704337_0065.html.

Downloaded on 26 Apr 2023 20:24:37 SGT

A SIMPLIFIED VITERBI MATCHING ALGORITHM FOR WORD PARTITION IN VISUAL SPEECH PROCESSING

SAY WEI FOO

*School of EEE., Nanyang Technological University, Singapore, 639798
E-mail: eswfoo@ntu.edu.sg*

YONG LIAN AND LIANG DONG

*ECE Dept., National University of Singapore, Singapore, 119260
E-mail: {eleliany, engp0564}@nus.edu.sg*

In this paper, a novel simplified Viterbi matching algorithm for sequence partition is presented. This method connects the states of different HMMs to model the observed sequence. The method is applied to visual speech processing to partition the word into visual speech elements. Experimental result shows that good accuracy is achieved with the proposed method.

1. Introduction

In automatic lip reading systems using Hidden Markov Models (HMMs), how to partition the word or sentence into individual recognition units is an important topic of research. In traditional method, exhaustive means such as level building is commonly applied to achieve this goal. In this paper, a novel attempt in solving this problem is reported.

In the proposed method, the HMMs are treated as detachable state set. A word is partitioned into visual speech elements by connecting the states of different HMMs. A simplified Viterbi matching is performed while retrieving the state chain. The experimental results indicate that the word can be partitioned with good accuracy if the basic visual speech elements are configured properly.

2. Construction of Viseme Models and State Database

The basic visual speech elements in English are called visemes. A viseme is a time series indicating the movement of the lip that is repeated in different word articulation. In MPEG-4 multimedia standards, the visemes are categorized into 14 groups with one viseme corresponding with 2~5 visually similar phonemes [1]. Our investigation on the dynamical features of visual speech shows that the

Figure 1. a.) The three phases of viseme production
b.) Segmentation of the transition state from the production of /ot/

motion of the lip can be partitioned into three phases during the production of an independent viseme: The first phase is called initial phase, starting from a closed mouth to the beginning of sound production. The next phase is called articulation phase, which is the course when sound is produced. The third phase is the end phase when the mouth restores to the relaxed state. Among the above three phases, the articulation phase is the most important for recognition because the difference between visemes chiefly lies there and it is relatively independent to the context. The initial phase and the end phase, on the other hand, are transitional ones. They may change much under different contexts.

HMMs are used to model the visemes in this paper. Assume $\{S_1, S_2, \dots, S_N\}$ is the state set and $\{O_1, O_2, \dots, O_M\}$ is the symbol alphabet, an N -state- M -symbol HMM is determined by: probabilities of the initial state $\pi = [\pi_i]$, state transition matrix $A = [a_{ij}]$ and symbol emission matrix $B = [b_{ij}]$. In the proposed strategy, three-state left-right HMMs are adopted to model the visemes. If the parameters of the HMMs are properly configured, the states of the HMM can be physically associated with the three phases of viseme production [2]. After performed Baum-Welch estimation, a number of viseme models (HMMs) are obtained. The articulation states of the HMMs are collected as the recognition units.

On the other hand, the transitions between visemes, which are used to connect the recognition units, are also segmented out of two visemes joint together, e.g. /ot/. As illustrated in Fig. 1(b), the approximate transition phase is manually segmented from the image sequence. The statistics of the transitions between different pairs of visemes are obtained by counting and averaging the symbols appeared in the process. These probability functions are collected to build the transition unit set.

3. Simplified Viterbi Matching Algorithm for Word Partition

The purpose of sequence partition is to decode a state chain that the recognition units and transition units appear alternately. Given the state database Θ and T -

Figure 2. The probability trellis that is generated from the forward process

length observation sequence $X = \{x_1, x_2 \dots x_T\}$, the task is to find a state sequence $Q = \{q_1, q_2 \dots q_T\}$ to maximize $P(Q|X, \Theta)$, which is equivalent to maximizing $P(Q, X|\Theta)$. A Viterbi matching is implemented as follows.

Initialization: Assume the sequence starts from a transition state, e.g. from the mouth is closed, the first column of the probability trellis is obtained from (1),

$$\delta_i(i) = \begin{cases} 0 & 1 \leq i \leq K \\ \pi_i b_i(x_1) & K < i \leq K+L \end{cases} \quad \text{and} \quad \psi_i(i) = 0 \quad 1 < i \leq K+L \quad (1)$$

where π_i is the initial probability of the i -th state. Because a word may start from any transition state, a uniform distribution $\pi_i = 1/L$ ($K < i \leq K+L$) is applied to the transition units.

Forward Process: At each moment, a state will either repeat itself or transit to a unit in another set. If the $(t-1)^{\text{th}}$ state is a recognition unit, the t^{th} state can be the same recognition unit as the $(t-1)^{\text{th}}$ state or a transition unit. Two probabilities are then computed.

$$\delta_i^1(i) = \delta_{i-1}(i) a_{ii} \quad \text{and} \quad \delta_i^2(i) = \max_{K < i \leq K+L} [\delta_{i-1}(i)(1 - a_{ii})] \quad 1 \leq i \leq K \quad (2)$$

and the t^{th} column of the probability trellis is,

$$\delta_i(i) = \max[\delta_i^1(i), \delta_i^2(i)] b_i(x_t) \quad \text{and} \quad \psi_i(i) = \arg \max_i [\delta_i^1(i), \delta_i^2(i)] b_i(x_t) \quad (3)$$

If the $t-1$ -th state is a transition state, the following probabilities are computed.

$$\delta_i^1(i) = \lambda \delta_{i-1}(i) \quad \text{and} \quad \delta_i^2(i) = \max_{1 \leq i \leq K} [\delta_{i-1}(i)(1 - \lambda)] \quad K < i \leq K+L \quad (4)$$

where λ has the same meaning as state reiteration probability a_{ii} . It controls the duration of the transition state. Also, we have,

$$\delta_i(i) = \max[\delta_i^1(i), \delta_i^2(i)] b_i(x_t) \quad \text{and} \quad \psi_i(i) = \arg \max_i [\delta_i^1(i), \delta_i^2(i)] b_i(x_t) \quad (5)$$

Termination: The last column of the probability trellis is

$$\delta_{end} = \max[\delta_T(i)] \quad \text{and} \quad q_T = \arg \max_{1 \leq i \leq K+L} [\delta_T(i)] \quad 1 \leq i \leq K+L \quad (6)$$

The optimal state sequence is obtained by backtracking the nodes using (7).

$$q_t = \psi_{i+1}(q_{t+1}) \quad t = T-1, T-2, \dots, 1 \quad (7)$$

In (4), the entry λ is introduced to build a state chain because the transition unit is merely a probability distribution function. In applications, λ can be set to different values to better fit the process investigated.

4. Experiments

Some experiments are conducted to assess the performance of the proposed method for word partition. As listed in Table 1, some words are decoded into viseme combinations with greatest probabilities are listed in Table 1 and the correct partition is marked with a “√”. It can be observed that the correct partitions always appear in the best matches.

Table 1. Examples of word partition with the proposed strategy

Word	Viseme combinations	Probability
lip	/l/ + trans. + /p/	43%
	/l/ + trans. + /l/ + trans. + /p/	32% √
	/ch/ + trans. + /p/	11%
zoo	/U/	72%
	/z/ + trans. + /U/	17% √
	/sh/ + trans.	7%
with	/w/ + trans. + /l/ + trans. + /tz/	35% √
	/U/ + trans. + /tz/	30%
	/f/ + trans. + /s/	9%
black	/b/+trans.+/l/+trans.+/A/+trans.+/k/	27% √
	/b/+trans.+/l/+trans.+/A/+trans.+/t/	24%
	/b/ + trans.+ /A/ + trans. + /k/	13%

5. Conclusion

In this paper, the simplified Viterbi algorithm is applied to lip reading as an example. This method can also be adopted in sequence partition of the following conditions: 1.) the states of the HMM of the basic recognition units can be classified as variable states and relatively constant states. 2.) It is difficult to decode the target sequence into an HMM chain of the basic recognition units.

References

1. M. Tekalp and J. Ostermann, “Face and 2-D mesh animation in MPEG-4”, *Image Communication J.* (1999)
2. Say Wei Foo and Liang Dong, “Recognition of Visual Speech Elements Using Hidden Markov Models”, *Proc. PCM 2002*, **LNCS2532**, (2002)
3. L. R. Rabiner and B. H. Juang, “Fundamentals of speech recognition”, *Prentice Hall Signal Processing Series*, (1993)
4. Liang Dong and Say Wei Foo, “Boosting of Hidden Markov Models and the Application to Visual Speech Analysis”, *Proc. IWSSIP*, **358-364**, (2002)