Lip Detection in Video using AdaBoost and Kalman Filtering

Bac Le Hoai, Viet To Hoai, Thao Nguyen Ngoc

Faculty of Information Technology, University of Science, Ho Chi Minh City
Email: {lhbac, thviet, nnthao}@fit.hcmus.edu.vn

Abstract: Lip reading is an active field that receive much attention from computer scientists. Its applications take part not only in science, such as a speech recognition system, but also in social activities, such as teaching pronunciation for deaf children in order to recover their speaking ability. In this paper, we aim to solve a narrower problem, the lip tracking, which is an essential step to provide visual lip data for the lip-reading system. Inspired by the idea of AVCSR, which has combined visual features with audio features to increase the accuracy in noisy environments, we use AdaBoost algorithm and Kalman filter for the face and lip detectors. Our result shows that the system can detect and track mouth motion in real time. It is a critical point that encourages more researches in the visual tracking and voice recognition fields.

Keywords: face dection, lip detection, Kalman filter, Adaboost

I. INTRODUCTION

Children who are deaf from birth usually tend to be unable to speak because they cannot receive sound signal to imitate when they were babies. However, their speaking ability still exists. To recover this ability, deaf children are taught to pronounce with the hope that they can speak as normal people. Lip reading technique from the computer science field can help us to fulfill this hope. Lip reading is the technique to recognize what a person is saying by visually interpreting the movements of the lips, face, and tongue with information provided by the context, language, and any residual hearing. Applications applied this technique can help the deaf to communicate easily and we can use these applications for education purposes, such as to teach them how to pronounce correctly. This paper aims to solve a narrower problem, the lip tracking, which is an essential step to provide visual lips data for the lip reading system. The purpose of lip tracking is to locate mouth on a human face. There are two main steps: detect the face and locate the mouth on that face. For many years, face detection has been developed by many scientists and there are many approaches available [1]. We can divide these approaches into different groups, (a) *Methods based on knowledge* about parts of face, (b) *Methods based on invariant features* of a face, (c) *Pattern matching methods*, (d) *Methods based on appearance*. Among these groups, methods based on appearance with the help of machine learning is the most prominent because they require less effort of human and can be applied in general cases. From static images to video, detecting methods have to face a problem about performance in real time. In this case, *Cascade of Boosted Classifiers*, introduced by Viola and Jones [2], which has high appreciation in both accuracy and time consuming, is an appropriate choice. Combining with Kalman filter, AVCSR system [3] give us promising results in tracking face and lip appeared in a video. Some other tracking approaches can be found in [6][7][8]. These models are suggested to describe lip shape border and then are applied to detect and track lip motion in a sequence of images. In [6], Yuille *et al*. used deformable template [4][5] to locate and tracke the border of lip. However, because of some constraints about initial polygon describing lip shape, we are prevented from modeling various borders with higher details. The snake method [7] can overcome the problem of details, but we still have a trade-off between flexibility and detailed analysis. Two approaches mentioned match the model with image edges with the assumption that strong edges lie along lip border. This assumption usually cannot be satisfied completely because lip edge varies due to speaker, light condition, the appearance of teeth, and how wide the mouth opens. Besides, these models also depend on some thresholds, weights, etc., which are determined by heuristic.

In contrast to those methods above, [8] uses Active Shape Models to model, locate, and track the lip border. This is a dynamic model for describing border or other import parts of a given object with a set of labeled points. All parameters of this model are not determined manually but computed automatically based on statistics using a training set. In the best case with optimal condition, the result of this model may reach to 81% of accuracy.

Inspired by the idea of AVCSR and excellent success of the OpenCV community in the face recognition problem, we decide to choose AdaBoost algorithm and Haar-like feature as the core for our face and lip detectors.

A group of researchers in University of Science, Ho Chi Minh City, Vietnam, has built educational software for deaf children, which supports them in pronouncing and developing thought. The lip tracking method took part in a module that teaches children how to open their mouth to pronounce correctly and helps them to practice the lip motions themselves through the webcam. The result got from this paper is not only useful in scientific community but also a good contribution for our society.

II. AUDIO-VISUAL CONTINOUS SPEECH RECOGNITION - AVCSR

Audio-Visual Continuous Speech Recognition (AVCSR) is a technology that combines sound features and visual features to improve the accuracy of voice recognition system in such environments with noise. This is also the name of a research project administrated by Intel[1]. AVCSR was built on the base of OpenCV, a famous open source library of Intel for digital image processing.

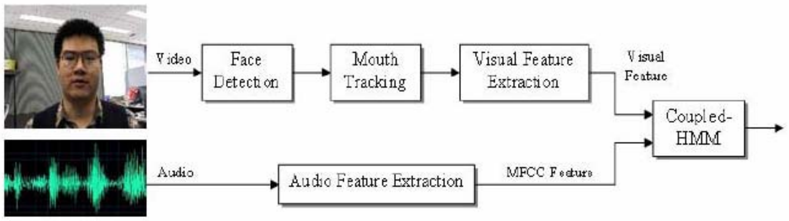


Figure 1: The AVCSR system

There are two parallel main steps (see Fig. 1): visual processing and audio processing. In the visual step, first, face and lips of the speaker are detected and tracked in a sequence of images. Then, a set of visual features are extracted from the lip area. In audio step, features extracted from the audio channel including Mel Frequency Cepstral Coefficients (MFCC). These features, visual and audio features, are modeled together using a Coupled Hidden Markov Model.

Here, we only focus on the former of two main steps in AVCSR, the lip tracking step.

III. THE TRACKING LIPS METHOD

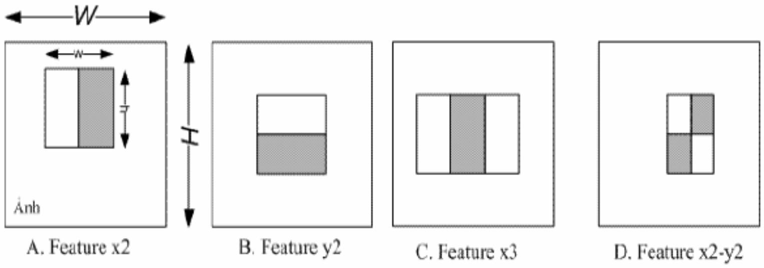
In order to track the lip on video data, we have solved two small problems: face detection and lip tracking.

1. Haar-like features

Haar-like features are often used in object recognition in digital images. They are equal rectangles used for calculating the different between pixels in adjacent regions. Not like single pixel, Haar-like features can describe the connection between parts of an object.

In Fig. 2a and 2b, features of an image are given by the difference of pixels in dark and light rectangles.

In Fig. 2c, it is the result of subtracting pixels in the middle rectangles from sum of pixels in two other rectangles. In Figure 2d, we calculate the feature by subtracting pixel in the dark rectangles from pixels in two light rectangles.



A. Feature x2 B. Feature y2 C. Feature x3 D. Feature x2-y2

Figure 2: The four basic Haar-like features

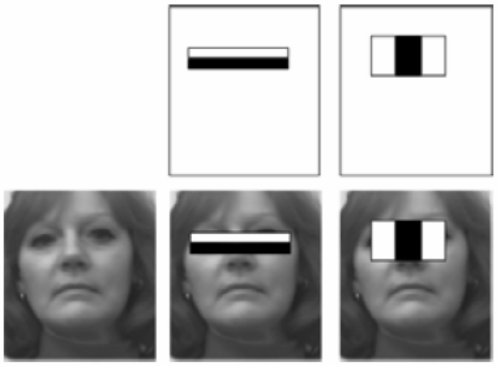
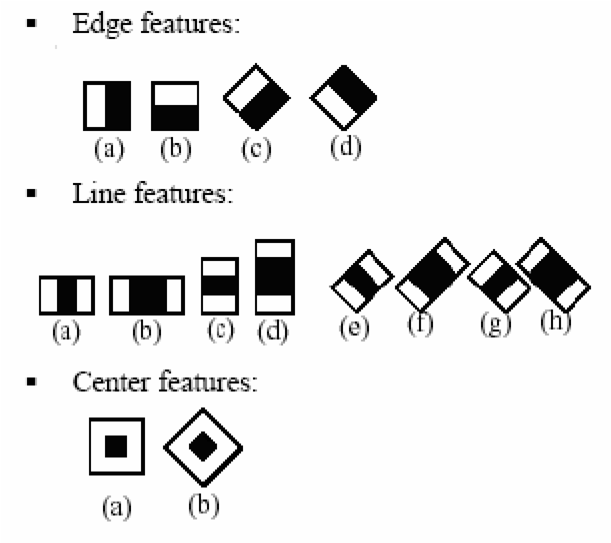


Figure 3: Application of Haar-like feature for detecting face

Features used for detecting face are extension of the basic Haar-like set. There are three important features sets which can be applied in face detecting listed below.



2. The AdaBoost approach

AdaBoost (Adaptive Boost) is a boosting approach introduced by Freund and Schapire [10]. Its principle is linear combination of weak classifiers to build a stronger classifier.

AdaBoost uses weights to identify samples that are difficult to recognize. While training, the algorithm updates weights of each classifier in order to prepare for constructing the next weak classifier: increase the weights of incorrectly recognized samples and decrease weights of correctly recognized samples by the weak classifier that has just been built. In that way, successive classifiers can concentrate on samples which former classifiers didn’t recognize well. After all, these weak classifiers will be combined together to create a strong classifier depending on their effect.

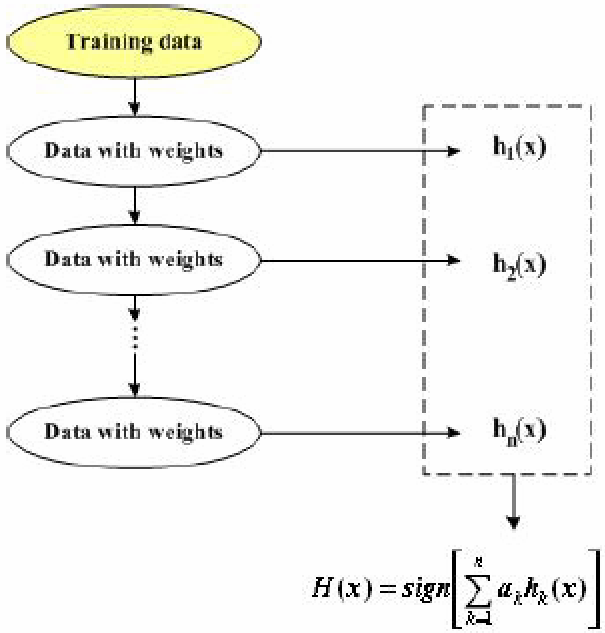


Figure 4: The strong classifier H(x) constructed with AdaBoost

For better understanding, we can simply imagine as follows. For knowing if an image is about hand or not, we ask T people (equivalent to T weak classifiers constructed from T loops of boosting). Evaluations of each person only need to be a little better than random. After that, we weights each evaluation by using α factor, the person who evaluates well difficult samples will have more important role in the final result than those only evaluate well on easy samples. We update samples' weights after each boosting loop so that we can determine the degree of difficulty for each sample. A difficult sample is the one that many people evaluate incorrectly.

Weak classifiers $h_k(x)$ is represented by the formula as below:

$$h_k(x) = \begin{cases} 1, p_k f_k < p_k \theta_k \\ 0, \textit{otherwise} \end{cases}$$

- $x = (x_1, x_2..., x_n)$: a feature vector of the sample
- θ: threshold
- f_k: an evaluation function for feature vector of the sample
- p_k: a factor that determines the direction of inequality.

The above formula can be expressed as follows. If value of a sample feature vector computed by evaluation function of a classifier exceeds a given threshold, this sample is an object (a target to recognize), otherwise it is a background (not a target).

Applying to the problem of lip motion tracking, although AdaBoost with Haar-like features has an acceptable result but its accuracy is still not absolute. In a sequence of lip motion, there are many moments that the classifier cannot detect mouth region, except for the former and latter moment. To connect them as a continuous sequence, we need to use additional Kalman filter.

3. Kalman filter

Kalman filter [9] is a kind of regression filter that can estimate effectively states in the past, present and future of a dynamic system from incomplete and noisy conditions. It solves a set of mathematics equations to do two main phases: prediction and update. Prediction phases use results of state estimation in previous steps to estimate for the next step. In update phase, information measured in a current step will be used to adjust the next prediction with a hope that the state estimation will be more accurate.

In this paper, Kalman filter takes part in the detecting process and tracking lips with two roles: first, it supports lip detector by estimating the center of mouth region in the next state; second, it performs final prediction in the post-processing step, based on state estimated previously. In the first role, we also applied simpler trick when estimating in real-time task (for example, tracking lip in frames capture from a webcam). When the Kalman filter failed to predict (may occur because we do not have the future information – the next incoming frames – and we just use one previous frame due to the limitation of time), we just simply use the center of the mouth in previous frames for the current frame. In post-processing step, Kalman infers the mouth location for frames that were failed to detect. Kalman filter is really a useful assistant for AdaBoost classifier to estimate the lip motion more exactly and continuously.

4. Detecting lips and tracking their motion

The lip tracking process is a combination of detecting lips and tracking the lips motion. In Figure 5, the core of the system is a finite-state-machine including two state, detecting and tracking. We also apply Adaboost approach with the same Haar features described in Section III.2 for lip detecting. In the training phase, two other classifiers, one for mouth with beard and one for mouth without beard, are trained using the same manner as in the face detecting classifier.

In the detection phase, the system first uses cascaded classifiers to detect face at different scales using Haar-like features. After that, two mouth classifiers will locate the mouth region in the lower part of face. If detecting successfully, the finite-state-machine will move to the tracking step.

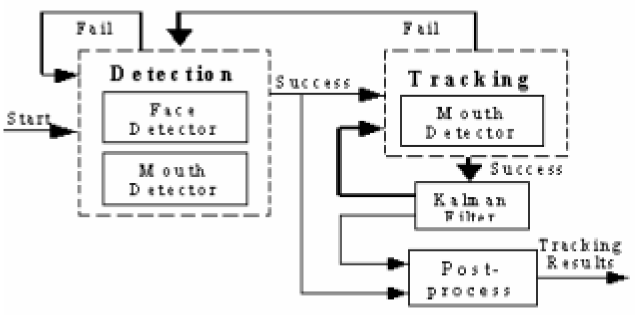


Figure 5: The process of detecting and tracking lips

While detecting, we apply lip detection algorithm into small regions around the position that contains lips in previous frames. Center of searching region is estimated by linear Kalman filter. The mouth region is smoothed and outliers are eliminated by a post-processing procedure including three steps. First, use linear interpolation to fill in blanks of the motion sequence. These blanks appeared because of failed detection. Then, a median filter will exclude incorrectly detection. Finally, Gaussian filter is used to remove noise.

IV. EXPERIMENTS

1. Data preparation

We do experiments mainly on two datasets as follows:

a. Dataset 1: includes 38 video files about pronouncing the Vietnamese alphabet and numbers (17 consonant, 12 vowels and number 1-9).

The video is recorded with webcam quality, speed of 25 frames per second (fps) and in AVI format. The average length for each file is 2-3 seconds, equivalent to 50-70 frames.

In the video, there is only one subject. This actress sits in front of the camera and looks straight at it. She pronounces in succession each character in the Vietnamese alphabet and each number.

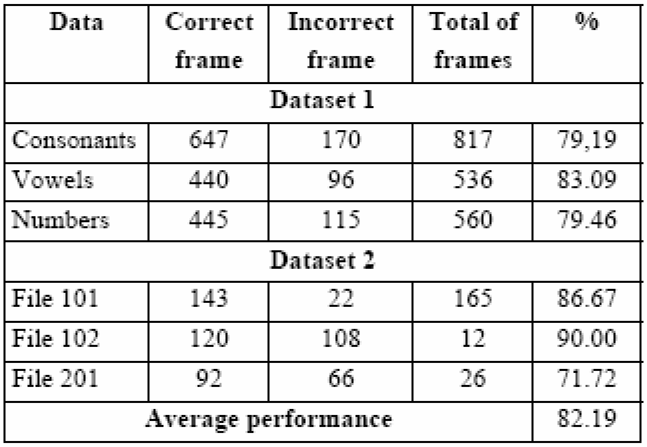
This is the dataset we constructed ourselves in order to instruct deaf children how to pronounce correctly through sample lips motion. At present, it is used for the lip tracking module in software Listen to Me version 2.0.

b. Dataset 2: includes three video files about pronouncing English numbers from 0 to 9.

The video file is recorded with professional camera, speed of 25 fps and has the AVI format. The average length for each file is 4-6 seconds.

There are two subjects in the dataset. The actors look straight at the camera and pronounce the number from 0 to 9. This is a standard dataset used as sample for testing the performance of AVCSR application developed by Intel.

Other data are news video on television. However, we cannot get data directly from a television station. We just collected them from the Internet. Hence, the quality of video decreases significantly. That is why we cannot use them in our experiments, although they are a valuable and meaningful data.



Data	Correct frame	Incorrect frame	Total of frames	%
Dataset 1				
Consonants	647	170	817	79,19
Vowels	440	96	536	83.09
Numbers	445	115	560	79.46
Dataset 2				
File 101	143	22	165	86.67
File 102	120	108	12	90.00
File 201	92	66	26	71.72
Average performance				82.19

Table 1: Results of the tracking lip method without Kalman filter

2. Evaluation method

In our lip detection model, a frame is successfully detected when it can detect and track the both face and lip. For evaluating the accuracy of the model, we count the frames that contain faces and also frames in which face is detected successfully.

The effectiveness of our model on given data is evaluated as follows.

$$Performance(\%) = \frac{Frames tracked successfully}{Total of\ frames contain face} \times 100$$

3. Results

We do two different experiments, one uses Kalman filter to fill in failed-detected frame, and one without Kalman filter.

Experiment 1: without Kalman filter, failed-detected frame are still not filled.

Experiment 2: combine Kalman filter to fill in frame that failed to detect before. The result shows that empty frames in Experiment 1 are track correctly with additional use of Kalman filter.

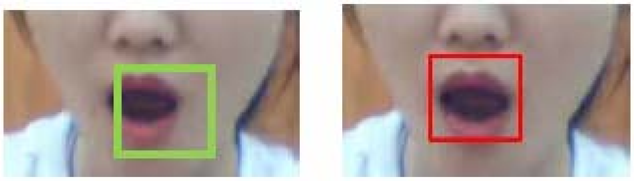


Figure 6: Result of tracking on the same frame without and with Kalman filter. The rectangle in the left image is the region detected in the previous frame. The rectangle in the right image shows the region predicted by Kalman filter.

4. Discussion

In the experiments, the datasets had the different quality. Dataset 1 is the video captured from a computer webcam with low resolution (640×480 pixels). Furthermore, to enable real-time tracking experiment for this task (tracking while capturing), we reduced its resolution to a half size. Dataset 2 is the testing video released by research group at Intel's lab. This dataset had a high quality with the frame rate at 30 fps and the resolution 1024×768. All these datasets were recorded in the office with normal lighting condition. The experimental results shown in previous sections are fulfilled the detecting performance requirement. With low quality dataset like wild data, the system cannot detect and track successfully. Because Adaboost is used as the detector, the detecting performance depends on the efficiency of Adaboost on detecting object in single images. Kalman filtering then can help to find the regions for missed frames when we have an initial successfully detected by Adaboost.

V. SUMMARY

These results in this paper prove that AdaBoost is not only suitable for static images but also works well in real time with video data. You have witnessed the effect of the practicing pronunciation module in the software for deaf children version 2.0.

We believe that continuing improve this method of lip tracking by AdaBoost, combining with some image enhancement methods, will lead to better result. Hence, the visual feature processing step of AVCSR is improved and the ability of voice recognition will also improve by using both visual and voice features.

REFERENCES

[1] M. H. Yang, David J. Kriegman, and Narendra Ahuja, *Detecting Faces in Images: A Survey*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 24, No. 1, January 2002.

[2] P. Viola and M. J. Jones, *Robust real-time face detection*, International Journal of Computer Vision, 57(2):137--154, May 2004.

[3] *Open Source Audio-Visual Continuous Speech Recognition Documentation*, Intel Corporation, Software and Solutions Group.

[4] M. E. Hennecke, K. V. Prasad and D. G. Stork, *Using Deformable Templates to Infer Visual Speech Dynamics*, 28th Annual Asilomar Conference on Signals, Systems and Computers, 1994.

[5] R. R. Rao and R. M. Mersereau, *Lip Modeling for Visual Speech Recognition*, 28th Annual Asilomar Conference on Signals, Systems and Computers, 1994.

[6] A. L. Yuille, P. Hallinan and D. S. Cohen, *Features extraction from faces using deformable templates*, Int. J. Computer Vision, Vol. 8, pp. 99-112, 1992.

[7] M. Kass, A. Witkin and D. Terzopoulos, *Snakes: active contour models*, Int. J. Computer Vision, pp. 321-331, 1988.

[8] J. Luettin, N. A. Thacker and S. W. Beet, *Active Shape Models for Visual Speech Feature Extraction*, D. G. Storck (Editor), Speechreading by Man and Machine: Models, Systems and Applications (NATO Advanced Study Institute), Springer Verlag, 1996.

[9] R.E. Kalman, *A new approach to linear filtering and prediction problems*, Journal of Basic Engineering 82 (1): 35-45.

[10] Y. Freund and R. Schapire, *A decision-theoretic generalization of on-line learning and an application to boosting*, Journal of Computer and System Sciences, 55(1):119–139, 1997.

AUTHORS' BIOGRAPHIES

Bac Le Hoai (1963) received the BSc degree, in 1984, the MSc degree, in 1990, and the PhD degree in Computer Science, in 1999. He is an Associate Professor, Vice Dean of Faculty of Information Technology, Head of Department of Computer Science, University of Science, Ho Chi Minh City. His research interests are in Artificial Intelligent, Soft Computing, and Knowledge Discovery and Data Mining.

Viet To Hoai (1982) received the BSc degree in computer science from the University of Science, HCM City, in 2002, and the MSc degree in computer science from the same university in 2009. He is a lecturer of Department of Computer Science, Faculty of Information Technology, University of Science, Ho Chi Minh City. His research interests are Artificial Intelligent and Ontology Matching.

Thao Nguyen Ngoc (1984) received the BSc degree in computer science from the University of Science, HCM City, in 2002. She is a lecturer of Department of Computer Science, Faculty of Information Technology, University of Science, Ho Chi Minh City. Her research interests are in Computer Vision and Knowledge Discovery and Data Mining.