AUDIO-VISUAL INTENT-TO-SPEAK DETECTION FOR HUMAN-COMPUTER INTERACTION

Philippe de Cuetos

Institut Eurecom
2229, route des Crêtes, BP 193
06904 Sophia-Antipolis Cedex, FRANCE
decuetos@eurecom.fr

Chalapathy Neti, Andrew W. Senior

IBM T.J. Watson Research Center
Yorktown Heights, NY 10598, USA
(cneti,aws)@us.ibm.com

ABSTRACT

This paper introduces a practical system that aims to detect a user's intent to speak to a computer, by considering both audio and visual cues. The whole system is designed to intuitively turn on the microphone for speech recognition without needing to click on a mouse, thus improving the human-like communication between users and computers. The first step is to detect a frontal face through a simple desktop video camera image, by using some well-known image processing techniques for face and facial feature detection on one image. The second step is an audio-visual speech event detection that combines both visual and audio indications of speech. In this paper, we consider visual measures of speech activity as well as audio energy to determine if the previously detected user is actually speaking or not.

1. INTRODUCTION

Speech recognition systems have opened the way towards intuitive and natural Human-Computer Interaction (HCI). Today, people can control their computer by using their voice: they can talk to computers just as they do to other people. While, for the moment, this audio communication is mostly one-sided and limited to simple command actions and word processing, computers are becoming more human-friendly and natural to use because they can emulate an important feature of human communication: the ability to listen and understand what other people say. However, this ability is not the only one to affect human communication.

Indeed, auditory cues are often used in combination with visual cues in human communication processes. Visual cues help, for instance, to determine who is speaking and even to improve speech recognition accuracy in noisy environments by looking at the speaker's mouth (see [11]). In particular, they can also help to determine to whom a person is speaking, as well as his intent to speak. Detecting that someone is speaking by just hearing him does not tell you, in general, if this person is actually talking to you or to someone else. Computers have the same problem: when they detect audio speech activity they do not know if it is intended for them. What you say to someone else on the phone or in the same room could be interpreted as a command by the computer — in particular as some word to type in a word processor. Current HCI systems using speech recognition software allow users to show their intent to speak to the computer by turning a microphone on, using the keyboard or mouse. Of course this is not a natural and convenient thing to do for the user.

Our system assumes the existence of a desktop video camera, mounted above the computer screen, that continuously monitors the user in front of his computer. Having a picture of the user (or the room without the user), the first step is to detect frontal faces in the picture. We will assume that, when a person is frontally looking at the computer screen, appearing as a full-frontal face in the camera image, he gives a first indication that he wants to communicate with the computer. We choose to detect frontal faces of size in a certain range in order to detect users that are close enough to their computer screen. When a frontal face has been detected on the current frame, we turn the microphone on and we look at audio and visual speech activities to detect speech behavior by the user. The mouth shape, through viseme classification, will yield a visual indication of speech, that will be combined with audio energy to yield the final decision of speech activity.

2. FRONTAL FACE DETECTION

The first step of our system is to detect a frontal face in a camera image. This is a 2-class pose detection problem. Our approach is to adapt some more general techniques for face and facial feature detection on a single image.

2.1. Face and Facial Features Detection

The details of our algorithm for face detection are explained in [1]. The first step is a skin-tone segmentation that locates image regions where colors could indicate the presence of a face. Then, the image is sub-sampled and regions are compared to a face template using Fisher Linear Discriminant and Principal Component Analysis as explained in [8]. This yields a first face likelihood score that is combined with another score based on Distance From Face Space (DFFS), which considers the distribution of the image energy over the eigenvectors of the covariance matrix, as explained in [1]. The combination of both scores yields a final face likelihood score that we will call the *face score*. The higher the *face score*, the higher the chance that the considered region is a face. It is thus compared to a threshold to decide whether or not a face candidate is a real face.

When a human face has been detected on the image, a similar method is applied, combined with statistical considerations of position, to detect the features within the face. Note that this face and feature detection scheme was designed to detect strictly frontal faces only, and the templates are intended only to distinguish strictly frontal faces from non-faces: faces in more general pose are not considered at all.

2.2. The Pruning Method

We have noticed that the *face score*, for a given user, varies almost linearly as the user is turning his head — the score being the highest when the face is strictly frontal and the lowest when it is profile. But the absolute score is user dependent making it difficult to find a robust user independent threshold that could allow us to decide on the frontalness of the pose by simple thresholding of the *face score*. So, we tried to rely on some face-specific considerations such as face features and face geometry, in order to get results that are more robust to users' change than the simple face score thresholding.

The method consists of combining face and feature detection to prune non-frontal faces. We first detect faces candidates in the current frame using a low *face score* threshold. This low threshold will allow us to detect even faces that are far from being strictly frontal, so that we do not miss any more general frontal face. Of course this will also yield some profile faces and even non-faces to be detected. Then, in each candidate, we estimate the location of the facial features.

The false candidates (non-frontal faces and non faces) should be pruned according to the following computations:

- The sum of all the facial feature scores.
- The number of main facial features that are well recognized. These features are the nose, the mouth and the eyes. These are the most characteristic and visible features of a human face and they differ a lot between frontal and non-frontal faces.
- The ratio of the distance between each eye and the center of the nose.
- The ratio of the distance between each eye and the side of the face region (each face is delimited by a square for template matching as explained in [1]). these last ratios, for 2D projection reasons, will differ from 1 the more as the face is non-frontal.

These 4 measures are independently thresholded to yield an indication of non frontality. Such an indication has the effect of removing the considered frontal face candidate from the candidate stack. After this pruning, if one or more face candidates remain in the candidates stack, we will consider that a frontal face has been detected in the current frame.

Finally, we use the temporal constraints of our interactive system in order to smooth results. As the video camera is expected to take pictures from the user at a high rate (typically 15 frames per second), we can use the results of the former x frames (x is called the *burst* parameter) to predict the results in the current frame, assuming that human movements are slow compared to the frame rate. For instance, if a frontal face has been found in the current frame, we expect that this face will remain frontal in the next x frames. This mechanism avoids some false negative results that can sometimes happen for a limited number of frames.

2.3. Results

This section presents results for the pruning method and compares them with results for the simple thresholding method.

The Receiver Operating Curves (ROCs) shown in figure 1 were made from a video of 6 different users taken by an inexpensive desktop camera. Each user turns his head from far right to far left over 2 seconds, yielding around 30 frames per user. The *burst* value is set to 1 and the *face score* threshold goes from -0.1 to 0.2 — for the pruning method, this is the minimum value of a candidate *face score* before pruning. False positive detections are frames where a frontal face has been detected when there is actually none, and false negative detections, frames where no frontal face has been detected when there was at least one.

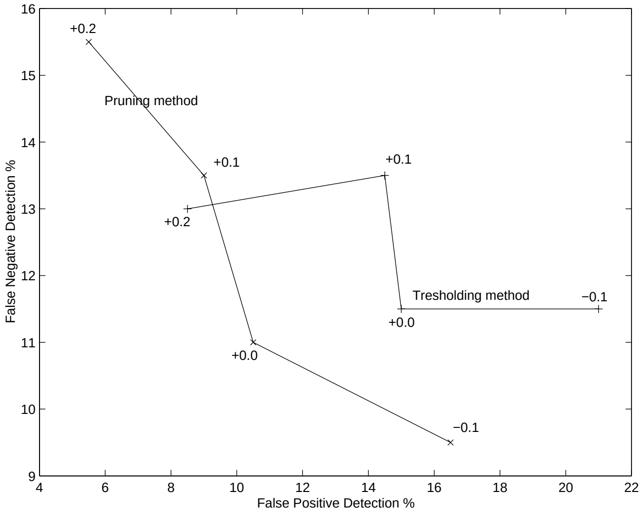

Figure 1: ROCs from the results of the pruning and thresholding methods on a video of 6 different users turning their head in front of the camera.

We see clearly, in figure 1, that for a given face threshold (except for 0.2), we obtain the best results in terms of both false positives and negatives with the pruning method. The closer the ROC to the bottom-left corner of the plot, the better the performance of the method. Thus, pruning method is better than the simple thresholding method. A good candidate threshold or operating point is easy to choose: it would be undoubtedly 0.0 for the pruning method.

This experiment yields high false detection rates because it corresponds to a test situation (6 different users turning their head rapidly one after the other), who should be different from the behavior of users in real situation in front of their computer screen. Table 1 shows the results obtained from a 2000 frame video of a user in real situation. The face threshold is set to 0.0 and the *burst* value varies from 1 to 7. We obtain a total correct classification rate of 97% for the pruning method.

Burst parameter x	1	2	4	7
Pruning method	97	97	96	96
Thresholding method	96.5	96	96	95

Table 1: Correct response rates according to the *burst* value and the method used.

3. AUDIO-VISUAL SPEECH DETECTION

We now assume that at least one frontal face has been detected by the pruning method on the current image. The next step in detecting the user's intent to speak is now to consider more specifically his mouth activity.

3.1. Visual indication of speech activity

Having detected a frontal face in the image frame, as well as some facial features, we now extract a bounding rectangle of the detected mouth region and work on this new image. An example of an extracted mouth region is shown in figure 2.

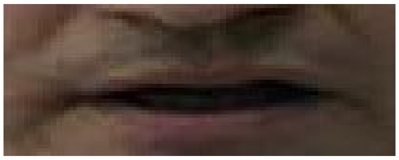

Figure 2: Example of an extracted mouth region

Having extracted the mouth region, we use the visual phonetic classification introduced in [10] to distinguish two classes: visual silence and visual speech. From the 27 visemes that we use for lip reading, 2 visemes of our model represent visual silence and the others represent visual speech. The visual speech features used are the first 100 modes of variations of a Principal Component (PCA) projection of a 45x30 lip image.

Here are the results of our visemic classification. We used the ViaVoice Audio-Visual Database described in [10], which consists of about 700 sentences spoken by 6 different speakers. We extract the 45x30 mouth region images and use project to 100-dimensional PCA vectors. We represent each class distribution by a Gaussian mixture. The detection rates for training and testing data, as a function of the number of Gaussian-mixture components used to model each class distribution, are shown in table 2.

Note that these are only detection rates on one isolated frame. We are currently working on considering a succession of frames in order to take into account the continuity of human visual speech.

3.2. Audio indication of speech

Having found a way to yield a visual indication of speech, we have to find some appropriate audio parameters that can indicate the auditory existence of speech. One of these is intuitively the audio energy, which is higher during speech activity than during silence. We

	m	Silence	Speech	Mean
Training	10	73.94%	79.11%	77%
	20	74.56%	80.93%	78.31%
	30	69.97%	86.22%	79.75%
Testing	10	77.37%	54.72%	63.94%
	20	75.72%	62.32%	67.78%
	30	70.19%	70.32%	70.27%

Table 2: Visual speech classification results. m represents the number of Gaussian mixture components for each of the two classes

use c_0, the zeroth cepstral coefficient, which is a measure of speech energy, to yield an auditory indication of speech activity. Using around 250000 speech samples and 1000 silence samples from the HUB4 database, we build single Gaussian models of speech and silence classes. In table 3, are presented the results on the same number of test samples as of training samples.

	Silence	Speech	Mean
Training	69.33%	84.53%	84.47%
Testing	92.99%	87.80%	87.82%

Table 3: Audio speech activity detection rates based on audio energy

These results show that audio frames can be effectively classified into silence or speech just by using c_0. But we must also add, as in the pruning method, a continuity constraint so that the microphone does not turn off between each word as the user is uttering a whole sentence. This simple classification of speech and silence based on c_0 suggests that visual speech classification results require further improvements.

4. CONCLUSION AND FUTURE WORK

We have shown, in this paper, that we could efficiently detect a user's intent to speak by first detecting a frontal face on a camera image, and presented our initial experiments on detecting speech using visual indicators based on the mouth shape and audio indicators based on audio energy. We are now experimenting with methods to improve the visual classification performance and methods to combine visual and audio cues by considering early or late integration fusion techniques as explained in [11, 10].

5. REFERENCES

[1] Andrew W. Senior, "Face and feature finding for a face recognition system", Proc. Int. Conf. on Audio- and Video-based Biometric Person Authentication, March 1999, pp 154–159.

[2] Andrew W. Senior, "Recognizing faces in broadcast video." In proceedings IEEE Workshop on Recognition, analysis and tracking of faces and gestures in real-time systems, Sept 1999.

[3] Richard O. Duda and Peter E. Hart, "Pattern Classification and Scene Analysis", Wiley-Interscience 1973.

[4] Andrew Gee and Roberto Cipolla, "Estimating Gaze from a Single View of a Face", Tech. Rep. CUED/F-INFENG/TR174, March 1994.

[5] A. Jonathan Howell and Hilary Buxton, "Towards Visually Mediated Interaction using Appearance-Based Models", CSRP 490, June 1998.

[6] N. Krüger, M. Pötzch, C. von der Malsburg, "Determination of face position and pose with a learned representation based on labeled graphs", Image and Vision Computing 15, 1997, pp 665–673.

[7] R. Brunelli, "Estimation of pose and illuminant direction for face processing", Image and Vision Computing 15, 1997, pp 741–748.

[8] Peter N. Belhumeur, Joao P. Hespanha, and David J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 7, July 1997.

[9] David Beymer and Tomaso Poggio, "Face Recognition From One Example View", Massachusetts Institute of Technology, September 1995.

[10] S. Basu, C. Neti, N. Rajput, A. Senior, L. Subramaniam, A. Verma, "Audio-Visual Large Vocabulary Continuous Speech Recognition in the Broadcast Domain", Workshop on Multimedia Signal Processing, September 1999.

[11] Tsuhan Chen and Ram R. Rao, "Audio-Visual Integration in Multimodal Communication", Proceedings of the IEEE, Vol 86, No. 5, May 1998.