AUDIO-VISUAL CONTINUOUS SPEECH RECOGNITION USING MPEG-4 COMPLIANT VISUAL FEATURES

Petar S. Aleksic, Jay J. Williams, Zhilin Wu, and Aggelos K. Katsaggelos

Department of Electrical and Computer Engineering
Northwestern University
2145 North Sheridan Road, Evanston, IL 60208
Email: {apetar, jjw, zlwu, aggk} @ece.nwu.edu

ABSTRACT

In this paper we utilize Facial Animation Parameters (FAPs), supported by the MPEG-4 standard for the visual representation of speech, in order to significantly improve automatic speech recognition (ASR). We describe a robust and automatic algorithm for extraction of FAPs from visual data that requires no hand labeling or extensive training procedures. Multi-stream Hidden Markov Models (HMM) were used to integrate audio and visual information. ASR experiments were performed under both clean and noisy audio conditions using relatively large vocabulary (approximately 1000 words). The proposed system reduces the word error rate (WER) by 20% to 23% relatively to audio-only ASR WERs, at various SNRs with additive white Gaussian noise, and by 19% relatively to audio-only ASR WER under clean audio conditions.

1. INTRODUCTION

The use of visual information in addition to audio, improves speech understanding especially in noisy environments [1]. Improving ASR performance, by exploiting the visual information of the speaker's mouth region is the main objective of audio-visual speech recognition (AVSR) [2]. Many researchers have reported results that demonstrate AVSR performance improvement over audio-only ASR systems [3-5]. Most of these systems performed tests using a small vocabulary, while recently results on a large vocabulary were shown [3, 6].

MPEG-4 is an audiovisual object-based video representation standard supporting facial animation. MPEG-4 facial animation is controlled by the Facial Definition Parameters (FDPs) and FAPs, which describe the face shape, and movement, respectively [7]. The MPEG-4 standard defines 68 FAPs, divided into 10 groups as shown in Figure 1 [7]. Transmission of all FAPs at 30 frames per second requires only around 20 kbps (or just a few kbps, if MPEG-4 FAP interpolation is efficiently used [8]), which is much lower than standard video transmission rates. FAPs contain important visual information that can be used in addition to audio information in ASR. To the best of our knowledge no results have been previously reported on the improvement of AVSR performance when FAPs are used as visual features with a relatively large vocabulary audio-visual database of about 1000 words. Reporting on such results is the main objective of this paper.

AVSR performance depends strongly on the accuracy of the visual feature extraction algorithms. The use of templates, defined as certain geometric shapes, and their fitting to the visual features can, in some cases, be successfully used as a visual feature extraction algorithm [9]. On the other hand, in order to achieve a close fit of the templates to the real visual features the order of the geometric shapes has to be increased. As a result this also increases the computational requirements. The active contour method is a relatively new method for the extraction of visual features, and is very useful in cases where it is difficult to present the shape of an object with a simple template [4]. However, this method is sensitive to random noise, and to certain salient features (i.e. lip reflections, shadows, etc.).

In this paper, we first describe the audio-visual database used (Section 2) and an automatic and robust method for extracting FAPs, by combining active contour and templates algorithms (Section 3). We use the Gradient Vector Field (GVF) snake, since it has a large capture area, and parabolas as templates. Next the audio-visual integration model used and the AVSR system are described (Section 4). Finally, the results of the performance improvement over a wide range of noise levels and under noise-free conditions are reported in Sections 5 and 6.

2. THE AUDIO-VISUAL DATABASE

This work utilizes speechreading material from the Bernstein Lipreading Corpus [10]. This high quality audio-visual database includes a total of 954 sentences, of which 474 were uttered by a single female speaker, and the remaining 480 by a male speaker. For each of the sentences, the database contains a speech waveform, a word-level transcription, and a video sequence time synchronized with the speech waveform. Each utterance began and ended with a period of silence. The vocabulary size is approximately 1,000 words. The average utterance length is approximately 4 seconds. In order to extract visual features from the database, the video was sampled at a rate of 30 frames/sec (fps) with a spatial resolution of 320 x 240 pixels, 24 bits per pixel. The luminance information was used in the algorithms and the experiments. Audio was acquired at a rate of 16 kHz.

3. FAP EXTRACTION

Figure 2 illustrates the FAP extraction system we have implemented [11]. In order to extract the mouth area from the Bernstein Lipreading corpus, a neutral facial expression image was chosen among the sampled video images (Figure 3a). A 17 x 44 image of the nostrils (Figure 3b) was extracted from the neutral facial expression image to serve as a template for the template matching algorithm. The nostrils were chosen since they did not deform significantly during articulation. The template matching algorithm, applied on the first frame of each sequence, locates the nostrils by searching a 10x10 pixel area centered at the neutral face nose location, for the best match. In the subsequent frames, the search area is constrained to a 3x3

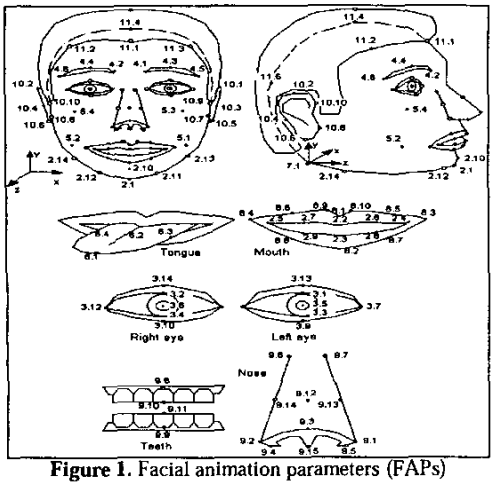


Figure 1. Facial animation parameters (FAPs)

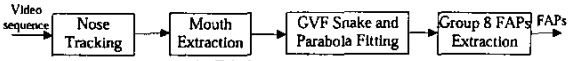


Figure 2. FAP extraction system

(a) (b) (c)

Figure 3. (a) Neutral facial expression image; (b) Extracted nose template; (c) Extracted mouth image

(a) (b) (c)

Figure 4. (a) Mouth image; (b) GVF, and (c) The final snake

pixel area centered at the nose location in the previous frame. Once the nose location has been identified, a rectangular 90 x 68 pixel region is extracted enclosing the mouth (Figure 3c).

In this work, only group 8 FAPs, which describe the outer lip movement, are considered (Figure 1).

3.1. GVF snake

A snake is an elastic curve defined by a set of control points [12], and is used for finding visual features, such as lines, edges, or contours. The snake parametric representation is given by

$$\boldsymbol{x}(s) = [x(s), y(s)], s \in [0,1], \quad (1)$$

where $x(s)$ and $y(s)$ are vertical and horizontal coordinates and s the normalized independent parameter. Snake deformation is controlled by internal and external snake forces, $E_{int}(\boldsymbol{x}(s))$ and $E_{ext}(\boldsymbol{x}(s))$, respectively. The snake moves through the image minimizing the functional

$$E = \int_0^1 (E_{\text{int}}(\boldsymbol{x}(s)) + E_{ext}(\boldsymbol{x}(s)))\, ds . \quad (2)$$

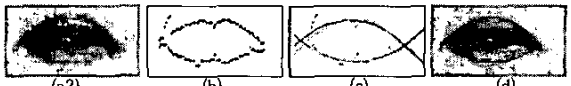
Internal snake force, $E_{int}(\boldsymbol{x}(s))$, is designed to hold the curve together by controlling its tension and rigidity. External force $E_{ext}(\boldsymbol{x}(s))$, is derived from the image data. The GVF [13], defined as a vector field, $\boldsymbol{v}(x,y)=(u(x,y),v(x,y))$, can be used as an external force. It is computed by minimizing the functional

(a) (b) (c) (d)

Figure 5. (a) Extracted mouth image; (b) Edge map image; (c) Upper and lower lip boundaries; and (d) Fitted parabolas

(a1) (b) (c) (d)

Figure 6. (a) Mouth image; (b) The edge map and the scan area; (c) and (d) The edge map after the step (i), and after the step (ii)



(a2) (b) (c) (d)

Figure 7. (a) Mouth image; (b) Edge map; (c) First step (dark) and second step (light) fitted parabolas, and (d) The final result.

$$\varepsilon = \iint \mu (u_x^2 + u_y^2 + v_x^2 + v_y^2) + |\nabla f|^2 |v - \nabla f|^2 \, dxdy \ , \quad (3)$$

where f is an edge map derived from the image using the gradient operator. The parameter μ is a weighting factor which is determined based on the noise level in the image. The important property of the GVF is that when used as an external force, it increases the capture range of the snake algorithm. Figure 4 depicts an example of the GVF, and snake results.

3.2. Parabola templates

Based on our investigations we concluded that the snake algorithm, which uses only the GVF as an external force, is sensitive to random image noise and salient features around the lips (i.e. lip reflections). In order to improve the lip-tracking performance two parabolas are fit along the upper and lower lip.

Edge detection is performed on every extracted mouth image using the Canny detector to obtain an edge map image (Figure 5). In order to obtain sets of points on the upper and lower lip, the edge map image is scanned column-wise, keeping only the first and the last encountered nonzero pixels (Figure 5c). Parabolas are fitted through each of the obtained sets of points (Figure 5d).

The noise present in the mouth image and the texture of the area around the mouth in some cases may cause inaccurate fitting of the parabolas to the outer lips. We resolved these cases by taking the following steps:

(i) We constrained the scan area by two half ellipses to eliminate the pixels in the corners of the mouth image (Figure 6). The fact that most of the edges in the edge map appear inside the mouth area is used to calculate in two steps the medians of the horizontal and vertical coordinates of the edge map pixels, and position the scan area. In the first step, medians were calculated based on all edges in the edge map image, while in the second step medians were obtained after outliers that contribute to the variance much more than average were removed (Figure 6).

(ii) In order to resolve the cases when there are unwanted edges inside the scan area (Figure 6d), we calculated in two steps the median (m) and variance (σ) of the vertical coordinate of the nonzero pixels, inside the scan area, and kept the ones inside the area of $(m-l*\sigma)$, and $(m+n*\sigma)$ (l and n are experimentally determined to be 4.75 and 4.5, respectively).

(iii) In order to eliminate unwanted edges, which are

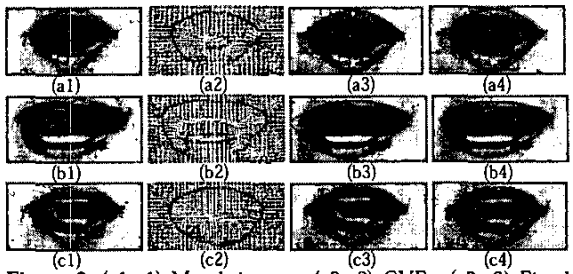


Figure 8. (a1-c1) Mouth images; (a2-c2) GVFs; (a3-c3) Fitted parabolas; and (a4-c4) Snake results, when GVF (a), or the parabola templates (b) do not give good results when applied individually, and when both methods give good results (c).

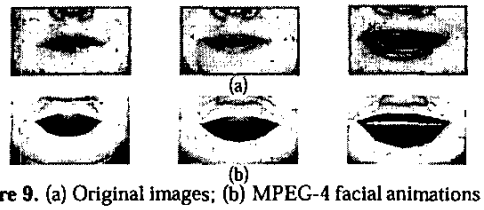


Figure 9. (a) Original images; (b) MPEG-4 facial animations [14]

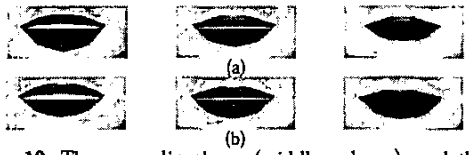


Figure 10. The mean lip shape (middle column), and the lip shapes obtained by the variation of the first (a), and second (b) eigenvector weights by 2 st. dev. (left and right column) [14].

close to the lip contour, we performed the fitting of the parabolas in two steps by eliminating the points that are the farthest from the parabolas obtained in the first step (Figure 7).

Afterwords, the image consisting of the two final parabolas was blurred and the parabola external force, $v_{parabola}$, was obtained using the gradient operator. $v_{parabola}$ was added to the GVF external force, v, to obtain the final external force, v_{final}, by appropriately weighting the two external forces, that is,

$$v_{final} = v + w \cdot v_{parabola}. \quad (4)$$

The value of w=1.5 proved to provide consistently better results. The final external force, v_{final}, was used in the snake algorithm. Shown in Figure 8 are the snake results for cases of bad quality GVF (Figure 8a), badly fitted parabolas (Figure 8b), and when both methods give sufficient information (Figure 8c).

The FAPs from group 8 are represented with the use of two FAP units, mouth-width separation and mouth-nose separation. Each of these two distances is normalized to 1024. The first frame face in each sequence was used as a neutral face for that sequence. The resulting lip contour of each frame, at time t, and the resulting lip contour of the neutral frame were compared in order to generate 10 group 8 FAPs, f_t.

Through visual evaluation of the FAP extraction results we observed that the extracted parameters produced a natural movement of the MPEG-4 decoder [14] that synchronized well with the audio (Figure 9). Therefore, we concluded that the developed algorithm performed very well, without any previous use of hand labeling or extensive training. However, the ultimate

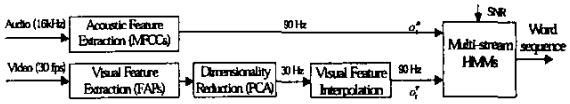


Figure 11. Audio-visual system for ASR

success in extracting the FAPs needs to be evaluated in terms of the increase in performance of the AVSR system.

3.3. Visual features dimensionality reduction

In order to decrease the dimensionality of the visual feature vector, Principal Component Analysis (PCA) was performed on the FAP vectors. After 10x1 mean FAP vector $\bar{f}$ and the 10x10 covariance matrix were obtained, the FAPs, f_t, were projected onto the eigenspace defined by the first K eigenvectors.

$$f_t = \bar{f} + E \cdot o_t^f, \quad (5)$$

where, $E=[e_1\ e_2 \ldots e_K]$ is the matrix of K eigenvectors, which correspond to the K largest eigenvalues, and o_t^f the Kx1 vector of corresponding projection weights. The first six, two and one eigenvectors represent 99.6%, 93%, and 81% of the total statistical variance, respectively. By varying the projection weights by ±2 standard deviations (st. dev.), we concluded that the first and second eigenvector mostly describe the position of the lower and upper lip, respectively (Figure 10).

4. AUDIO-VISUAL INTEGRATION

In this work, we utilized multi-stream HMMs and a late integration approach [5] for the combination of audio and visual speech information (Figure 11). The audio feature vector (o_t^a) consisted of 12 MFCC, an energy term, and the first and second order derivatives, while the visual feature vector (o_t^v) consisted of, projections weights (o_t^f), and the first and second order derivatives. The audio-visual feature vector, o_t, is obtained by concatenation of audio and visual feature vectors. Since MFCCs were obtained at a rate of 90Hz, while FAPs at a rate of 30Hz, visual features were interpolated in order to obtain synchronized data.

Audio and visual stream log-likelihoods are combined using the weights that capture their reliability. The combination of log-likelihoods can be performed at different levels of integration, such as state (used in this work), phone, word, and utterance [3]. The audio-visual features were used to train a multi-stream HMM, with state emission probabilities given by

$$b_j(o_t) = \prod_{s \in \{a,v\}} \left[\sum_{m=1}^{M_s} c_{jsm}\, N(o_t^s; \mu_{jsm}, \Sigma_{jsm}) \right]^{\gamma_s} \quad (6)$$

where subscript j denotes an HMM state, M_S denotes the number of mixtures in a stream, c_{jsm} denotes the weight of the m'th mixture of the stream s, and N is a multivariate Gaussian with mean vector μ_{jsm} and diagonal covariance matrix Σ_{jsm}. The non-negative stream weights are denoted by γ_s, and they depend on the modality s. We assumed the stream weights satisfy $\gamma_a+\gamma_v=1$.

5. SPEECH RECOGNITION EXPERIMENTS

The baseline ASR system was developed using the HTK toolkit version 2.2. The experiments used the portion of the Bernstein database with the female speaker. Context dependent phoneme models, triphones, were used as speech units. Iterative mixture

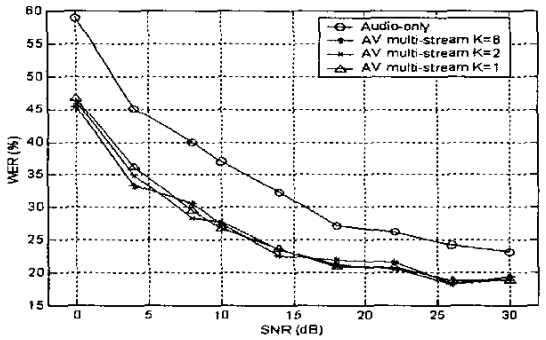


Figure 12. Audio-only, and audio-visual system WERs vs. SNR

splitting was performed to obtain the final 9-mixture triphone HMMs. Approximately 80% of the data was used for training, 18% for testing, and 2% as a development set for obtaining roughly optimized stream weights, word insertion penalty and the grammar scale factor. The bi-gram language model, used for decoding, was created based on the transcriptions of the training data set, and its perplexity was approximately 40. The same training and testing procedures were used for both audio-only and audio-visual experiments. To test the algorithm over a wide range of SNRs (0-30 dB), white Gaussian noise was added to the audio signals. The experiments with the clean speech were also performed.

All results were obtained using HMMs trained in matched conditions, by corrupting the training data with the same level of noise, as used for corrupting the testing data. This approach was used in order to accurately measure the influence of visual data on ASR performance. Audio-only ASR results are summarized in Figure 12. It can be observed that the ASR performance is severely affected by additive noise.

5.1. Audio-visual speech recognition experiments

In all AVSR experiments the stream weights were estimated experimentally by minimizing the WER on the development data set. The stream weights obtained, were roughly linearly related to the SNR, according to

$$\gamma_a = a \cdot SNR(dB) + b \qquad (7)$$

with parameters a (0.0059, 0.0064, 0.0062) and b (0.51, 0.5, 0.52) dependent on the visual feature dimensionality K (6, 2, 1).

The AVSR results obtained for different dimensionalities K are shown in Figure 12. As can be clearly seen, the AVSR system performs considerably better than the audio-only ASR system, for all K and SNR values. The relative reduction of WER, compared to the audio-only WER, ranges from 20% for SNR of 30 dB to 23% for SNR of 0 dB. It is important to point out that considerable ASR performance improvement was achieved, even in the case when only one-dimensional visual features were used (K=1). Clearly the fact that the system has similar performance for different dimension of visual features (K=1, 2, 6), is due to the trade-off between the number of HMM parameters that have to be estimated, and the amount of the speechreading information contained in the visual features. Results may be further improved by better adjusting the audio and visual stream weights [6].

The results obtained in experiments using clean speech are shown in Table 1. Considerable improvement in ASR performance was achieved for all values of K. The maximum relative reduction of WER (19%) over the audio-only ASR system WER was achieved when visual features of dimension 2 (K=2) were used with the audio stream weight equal to 0.7.

ASR system		WER [%]	Audio stream weight
Audio-only system		22.19	
Audio-visual system	K=1	18.21	0.75
	K=2	18.07	0.7
	K=6	18.16	0.85

Table 1. ASR performance under clean audio conditions

6. CONCLUSIONS

We described a robust and automatic FAP extraction system that does not require any previous use of hand labeling or computationally extensive training. We evaluated the system on a relatively large audio-visual database, for different values of the dimensionality of the visual feature vectors, over a wide range of noise levels and in clean audio conditions. Although we used only K-dimensional (K=1, 2, 6) visual feature vectors and no information utilizing mouth area intensities, we still obtained considerable improvement in ASR performance, for all noise levels tested, and under clean audio conditions. The improvement in ASR performance that can be obtained by exploiting the visual speech information contained in group 8 FAPs was determined.

As a second step we plan to extract all the remaining FAPs that contain important speechreading information (FAPs describing inner lip shape and tongue position) and determine how much they can improve the AVSR performance. We also plan to perform experiments on multi-speaker large vocabulary databases, such as [3], when they become available.

7. REFERENCES

[1] R. Lippman, "Speech recognition by machines and humans," *Speech Communication*, vol. 22(1), pp. 1-15, July 1997.

[2] D. G. Stork and M. E. Hennecke, editors, Speechreading by Man and Machine, Springer-Verlag New York Inc., 1996.

[3] C. Neti et al., "Audio-visual speech recognition," Tech. Rep., Johns Hopkins University, Baltimore, 2000.

[4] C. Bregler and Y. Conig, "Eigenlips' for robust speech recognition," In *IEEE Proc. ICASSP*, pp. 669-672, Adelaide, 1994.

[5] S. Dupont, J. Luettin, "Audio-visual speech modeling for continuous speech recognition, "*IEEE Trans. on Mult.*, vol. 2(3), pp. 141-151, 2000.

[6] H. Glotin et al., "Weighting schemes for audio-visual fusion in speech recognition," In *IEEE Proc. ICASSP*, vol. 1, pp. 165 -168, 2001.

[7] Text for ISO/IEC FDIS 14496-2 Visual, ISO/IEC JTC1/SC29/WG11 N2502, Nov. 1998.

[8] F. Lavagetto and R. Pockaj, "An efficient use of MPEG-4 FAP interpolation for facial animation at 70 bits/frame," *IEEE Trans. on Cir. and Sys. for Video Tech.*, vol. 11(10), pp.1085-1097, October 2001.

[9] A. L. Yuille, P. W. Hallinan, and D. S. Cohen, "Feature extraction from faces using deformable templates," *Int. Journal of Computer Vision*, vol. 8(2), pp. 99-111, 1992.

[10] L. Bernstein and S. Eberhardt, "Johns Hopkins lipreading corpus I-II," tech. Rep., Johns Hopkins U., Baltimore, MD, 1986.

[11] P. S. Aleksic, J. J. Williams, Z. Wu, A. K. Katsaggelos, "Audio-Visual Speech Recognition Using MPEG-4 Compliant Visual Features", to appear, EURASIP Journal on Applied Signal Processing 2002.

[12] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes:Active contour models.", *Int. Journal of Comp. Vis.*, vol. 1(4), pp. 321-331, 1988.

[13] C. Xu and J. L. Prince, "Gradient vector flow: A new external force for snakes," *Int. IEEE Proc. Conf. on CVPR*, pp. 66-71, June 1997.

[14] G. A. Abrantes, FACE-Facial Animation System, version 3.3.1, Instituto Superior Tecnico, (c) 1997-98.