Interspeech 2018
2-6 September 2018, Hyderabad

Visual recognition of continuous Cued Speech using a tandem CNN-HMM approach

Li Liu, Thomas Hueber, Gang Feng and Denis Beautemps

Univ. Grenoble Alpes, CNRS, Grenoble INP*, GIPSA-lab, 38000 Grenoble, France
* Institute of Engineering Univ. Grenoble Alpes
firstname.lastname@gipsa-lab.grenoble-inp.fr

Abstract

This study addresses the problem of automatic recognition of Cued Speech (CS), a visual mode of communication for hearing impaired people in which a complete phonetic repertoire is obtained by combining lip movements with hand cues. In the proposed system, the dynamic of visual features extracted from lip and hand images using convolutional neural networks (CNN) are modeled by a set of hidden Markov models (HMM), for each phonetic context (tandem architecture). CNN-based feature extraction is compared to an unsupervised approach based on the principal component analysis. A novel temporal segmentation of hand streams is used to train CNNs efficiently. Different strategies for combining the extracted visual features within the HMM decoder are investigated. Experimental evaluation is carried on an audiovisual dataset (containing only continuous French sentences) recorded specifically for this study. In its best configuration, and without exploiting any dictionary or language model, the proposed tandem CNN-HMM architecture is able to identify correctly more than 73% of the phoneme (62% when considering insertion errors).

Index Terms: Cued speech (CS), visual speech recognition, convolutional neural networks, hidden Markov models, modality fusion, assistive speech technology.

1. Introduction

Cued speech (CS) is a gesture-based communication system proposed by Cornett [1] in 1967. It uses a set of specific hand shapes and positions to complement the lip information and make all phonemes of a given spoken language clearly visible. Its goal is to overcome the limitations of lip-reading [2] and to improve the reading abilities for deaf children. It has been adapted to more than 60 different languages and dialects and is now used all over the world. For French, CS is named *Langue française Parlée Complétée (LPC)* [3], and is based on five hand positions and eight hand shapes encoding respectively the vowels and consonants (see Fig. 1). Automatic recognition of CS has been explored in [5, 6, 7]. In these studies, visual artifices were used to track lips and hand features. As shown in Fig. 2 (a), lips of the CS interpreter were painted in blue and color landmarks were placed on hand. Other studies such as [8, 9] focused only on the classification of static hand position and/or shape in CS. In [8], a color glove was used to make hand segmentation and tracking easier. The first motivation for the present study is to get rid of these artificial visual marks (as shown in Fig. 2 (b)).

As concern the decoding stage, classification of static hand configurations in CS was addressed in [9] using an artificial neural network (ANN). In [5, 7], HMM-GMM was used for CS phonemes recognition. In [5], HMM-GMMs were used to decode a set of isolated phonemes extracted from CS sentences, i.e. the temporal boundaries of each phoneme to recognize in the video was known at test stage. In [7], continuous phoneme recognition was also performed by HMM-GMMs. However, the dataset used in that study was composed only of isolated words repeated several times (not continuous sentences). Therefore, to the best of our knowledge, no study has addressed yet the problem of continuous decoding of CS sentences. Achieving this challenging task is the second main motivation of the present study.

Automatic recognition of CS shares some issues with other fields of multimodal speech processing, such as audiovisual speech recognition [10], visual speech recognition (i.e. automatic lip reading) [11], silent speech interfaces [12, 13], as well as with gesture recognition, including sign-language recognition [14]. Most of these fields have recently benefited from recent advanced of deep learning. When dealing with 2D data, the convolutional neural network (CNN) [15] has shown to be a powerful approach to learn representations directly from the raw data and to extract a set of high-level discriminative features. CNN has recently been used in [16] for automatic lip-reading, in [17] for speech synthesis driven by lips movements, in [13] for silent speech recognition and in [14] for sign language recognition.

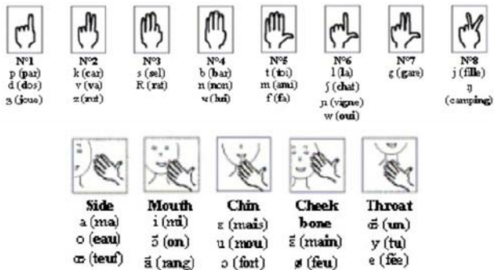

Figure 1: *French Cued Speech (from [4]).*

Figure 2: *(a) Artificial marks placed on the CS interpreter in [5, 7] in order to track lips and hand. (b) Typical images considered in the present study (no artificial marks are used).*

This work is partly supported by the CNRS "PEPS-LGV" grant.

10.21437/Interspeech.2018-2434

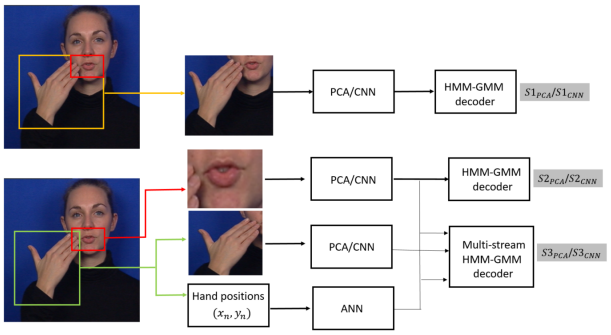

Figure 3: *Schematic representation of the different tandem architectures proposed for decoding automatically continuous CS.* (x_h, y_h) *encodes the coordinates of the center of the hand's ROI.*

In this study, we investigated the use of CNN to extract visual features from raw images of lip and hand in CS. CNNs are combined with an HMM-GMM classifier that models the dynamics of extracted feature trajectories for each phonetic context (in conventional ASR, this combination is often referred to as a *tandem* architecture). The different tandem architectures, as well as a baseline technique based on PCA, are described in Section 2. Experimental protocol and results are presented in Section 3.

2. Methodology

We propose several architectures for the automatic recognition of continuous CS which differs from each other in: 1) the considered region of interest (ROI): one single ROI containing both lips and hand *vs.* two distinct ROIs focusing on lips and hand, respectively, 2) the visual feature extraction technique: an unsupervised and linear technique based on PCA *vs.* a supervised and non-linear technique based on CNN, 3) the way lip and hand features are combined within the HMM-GMM decoder: early *vs.* middle fusion. These architectures are referred to as $S1_{PCA}/S1_{CNN}$, $S2_{PCA}/S2_{CNN}$, $S3_{PCA}/S3_{CNN}$. Fig. 3 shows an overview of all the proposed architectures in this work.

2.1. Feature extraction

2.1.1. Preprocessing

First, an ROI focusing on lips was extracted in each image using the Kanade-Lucas-Tomasi (KLT) feature tracker [18] (with a dezooming process). Then two approaches were investigated. In the first one, a unique bounding box large enough to contain both lips and hand and anchored on the lip ROI was defined (architectures $S1_{PCA/CNN}$). In the second one, two separate ROIs (lips and hand) were used. The hand ROI was extracted using the GMM-based foreground extraction technique described in [19] (architectures $S2_{PCA/CNN}$ and $S3_{PCA/CNN}$). Lips and hand ROIs were then converted to grayscale and resized to 64*64 pixel images using cubic interpolation.

2.1.2. PCA-based approach

This technique, also known as the EigenFaces technique [20], is an unsupervised and linear technique which aims at finding a decomposition basis that best explains the variation of pixel intensity in a set of training frames. At training stage, a PCA is performed on a set of N training frames (the resulting basis vectors are often called *EigenLips* [21] when applying this technique on lip images). At feature extraction stage, each new frame is projected onto the set of these basis vectors. Visual features are defined as the D first coordinates in that decomposition basis. It was set by keeping the eigenvectors that carry 85% of the variance, which led in our case to $D = 40$ for $S1_{PCA/CNN}$ when encoding jointly lips and hand, and $D = 34$ (resp. $D = 45$) when considering the lips (resp. the hand) only in $S2_{PCA}$ and $S3_{PCA}$.

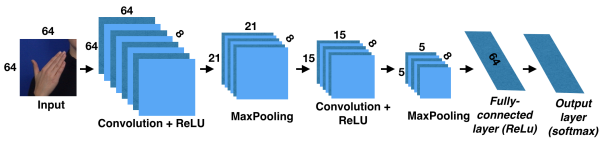

Figure 4: *Illustration of the convolutional neural network used to extract visual features from hand images in* $S2_{CNN}$ *and* $S3_{CNN}$.

2.1.3. CNN-based approach

In its canonical form, a CNN contains a given number of *convolutional layers*, each being divided into convolutional filtering, non-linearity, and pooling, stacked with a set of fully connected layers, and with an output layer giving the posterior probability of each class to decode. Because of the limited size of our dataset (see Section 3.1), instead of using the well-known CNN architectures (e.g. AlexNet), we only investigated a few architectures based on one or two convolutional/pooling layers, one fully connected layer, and one output (i.e *softmax*) layer. Cross-validation was used to optimize some hyper-parameters for each layer (i.e. the number of filters, the kernel size for the 2D convolutions, the down-sampling factor for the pooling layer, and the number of neurons in the fully connected layer). In all tested models, the activation function of the convolutional and fully-connected layer was *ReLu* whereas the *softmax* function was used for the output layer. For all architectures, two convolution layers with 8 filters, a kernel size of 7x7 pixels, a down-sampling factor of 3 (in both vertical and horizontal directions), and 64 hidden neurons in the fully connected layer were used. The structure of the CNN used to extract visual features from hand images in $S2_{CNN}$ and $S3_{CNN}$ is shown in Fig. 4.

At training stage, a mini-batch gradient descent algorithm based on the *RMSprop* adaptive learning rate method (with a learning rate equal to 0.001) and a batch size of 2048 frames, was used to estimate the CNN parameters. The categorical cross-entropy was used as loss function. Over-fitting was controlled using i) an early stopping strategy, i.e. 20% of the training set was used as a validation set and the training was stopped when the error on this dataset stopped decreasing during 10 epochs, and ii) a dropout mechanism (with a dropout probability of 0.25). All models were implemented using the *Keras* Python library [22] and were trained using GPU acceleration. After training, a vector of visual features was extracted from the CNN by taking the output of the fully-connected layer, just before the output layer (the dimension of the extracted feature is set as 64).

In the proposed CNN-HMM architectures, CNNs act as discriminative feature extractors. In $S1_{CNN}$, the single CNN models jointly lips and hand (position and shape). CNN is trained in a supervised manner with 34 phonetic classes as targets. In $S2_{CNN}$ and $S3_{CNN}$, each CNN focuses on lips or hand separately and was trained with either a set of 8 *visemes* as targets for lips or a discrete set of 5 positions and 8 shapes for the hand. For lips, the temporal segmentation of the lip image sequences into

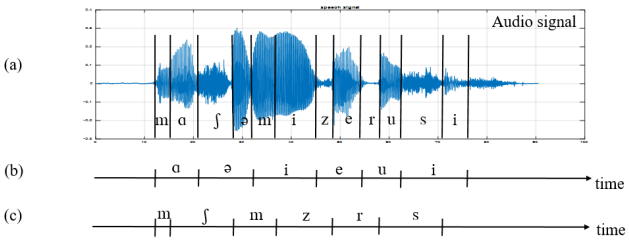

Figure 5: *Illustration of the procedure proposed to derive the temporal segmentation of the hand movements for the sentence "Ma chemise est roussie". (a) audio signal segmented at phonetic level, (b): segmentation of the hand position (encoding vowels in CS), (c): segmentation of the hand shape (encoding consonants in CS).*

visemes was derived directly from the audio signal (the asynchrony between lips and audio was here neglected). However, in CS, the hand generally precedes the lips. [23] reported a temporal advance of approximately one syllable (i.e. between 171 to 256ms). To take this phenomenon into account, a simple procedure was therefore used to derive the temporal segmentation of the hand stream from the temporal segmentation of the audio stream. As illustrated in Fig. 5, the left boundary of each phoneme (except the first phoneme) was extended to the beginning of the previous phoneme. The boundary of the first phoneme in each sentence keeps it as the audio based segmentation. This procedure allowed us to train the CNNs efficiently. Note that we recently proposed in [24] a method for predicting an optimal temporal segmentation of hand movements from the audio speech signal for each phonetic context. However, since this method did not bring significant improvements here, we kept the simpler procedure described above.

2.1.4. Hand position

In both $S2_{PCA/CNN}$ and $S3_{PCA/CNN}$, the coordinates of the center of the hand ROI were used as explicit additional features (we recall that in CS the position of the hand encodes the vowel). These values are then processed by a simple feed-forward neural network (ANN) with one single fully connected layer (with *ReLU* activation function) and one output *softmax* layer, trained with the very same procedure than the one used for CNNs.

2.2. HMM-GMM phonetic decoding

Sequences of visual features extracted using either PCA or CNN (and ANN for hand position in S2 and S3) were modeled, together with their first derivatives, by a set of context-dependent triphone HMM-GMM (i.e. phone model with left and right context). A standard topology was used with three emitting states (with no connection between initial and final state). HMM-GMMs were trained with *HTK 3.4* [25]. The number of components of each GMM emission probability was iteratively increased from 1 to 4. In S1, a single stream HMM-GMM was used to model visual features extracted from the joint observation of lips and hand. In S2, lips and hand were processed separately, and the three stream features were concatenated in a single feature vector (i.e. early fusion). In S3, lips and hand information were combined at the state level using a 3-stream HMM-GMM (middle fusion). The stream weights were optimized empirically (i.e. only a few combinations were tested) using cross-validation, and the optimal weights were found to be 0.4 for lips, 0.4 for hand shapes and 0.2 for hand positions. At decoding stage, the most likely sequence of phonemes was estimated by decoding the HMM-GMM state posterior probabilities using the Viterbi algorithm. For the HMM-GMM phonetic decoding, the model insertion penalty was optimized on the training set. Importantly, neither pronunciation dictionary nor language model was used in this study. In fact, we aimed at evaluating only the ability of the system to extract the phonetic information from raw data without any prior linguistic knowledge (indeed, the global performance should be significantly higher when using such information).

Table 1: *Monomodal decoding experiments. Two CNN-HMM for lips and hand shape, and a ANN-HMM for hand position were evaluated on each modality considered independently.*

Acc(%)	8 lips visemes	8 hand shapes	5 hand positions
CNNs-HMM	58.5	68.5	—
ANN-HMM	—	—	64.8

3. Experiments

3.1. Database

A database was recorded for the present study. A professional interpreter of CS (with no hearing impairment) was asked to utter and encode simultaneously a set of 238 French sentences (extracted from [26]). Each sentence was repeated twice resulting in a set of 476 sentences (about 11770 phonemes totally). Color video images of the interpreter's upper body were recorded at 50 fps, with a spatial resolution of 720x576 pixels RGB images. Data acquisition was done in the sound-proof room of GIPSA-lab, France. The French language was described with a set of 34 phonetic classes (14 vowels and 20 consonants). The French CS was described with 8 lips visemes, 8 different hand shapes, and 5 different hand positions (as described in [6]). The phonetic transcription was extracted automatically and manually post-checked to adapt it to the pronunciation of the CS interpreter. Importantly, the dataset is made publicly available (see Section 6).

3.2. Protocol and metrics

In our experiments, 80% of the sentences were randomly chosen to build the training set (with 20% used for validation), the remaining 20% for test (since the recorded dataset contains repetitions of the same sentences, we removed automatically from the training set each sentence that was selected for testing). All experiments were repeated 10 times, with each time, a different random partitioning of the data. Two metrics were used to assess the decoder performances: 1) the correctness of the HMM-GMM decoder defined as $Corr = (N - D - S)/N$ with N the number of phones in the test set, D the number of deletion error, and S the number of substitution errors, and 2) the accuracy of the HMM-GMM decoder Acc which takes into account the insertion errors (I) and which is defined as $Acc = (N-D-S-I)/N$. The statistical significance of these measurements was assessed by calculating the Binomial proportion confidence interval $\Delta_{95\%}$ (using the Wilson formula).

In addition to evaluating the performance of S1, S2 and S3 architectures, we run an additional series of experiments, referred to as the *monomodal decoding*. Two tandem CNN-HMMs, one for lips, the other one for hand shape and one ANN-HMM for hand position, were trained to classify each stream

Table 2: *Performance of the proposed architectures for automatic recognition of continuous CS, concerning correctness and accuracy. The temporal segmentation of hand streams mentioned in section 2.1.3 was used for CNN training. Confidence interval $\Delta_{95\%}$ is about 4%.*

	$S1_{PCA}$	$S1_{CNN}$	$S2_{PCA}$	$S2_{CNN}$	$S3_{PCA}$	$S3_{CNN}$
$Corr(\%)$	45.2	55.0	50.9	68.3	51.0	73.3
$Acc(\%)$	32.3	38.2	36.0	58.4	36.5	61.5

considered independently. Here, the HMM-GMM decoders were respectively trained with visemes, hand shape classes and hand position classes as targets. These experiments give some hints on the discriminative power of extracted visual features.

3.3. Results and discussion

First, we report in Table 1 the results of the *monomodal decoding* experiments. We observe a relative homogeneity between the different modalities, even one can have expected a better performance for the lip decoding task (however, these results remain comparable with [27]). In order to further understand the behavior of the CNN-based feature extractor, we show in Fig. 6 the output of the CNN processing hand images in $S2_{CNN}$ and $S3_{CNN}$ for one test sentence (i.e. the sequence of posterior probabilities for all possible hand shape classes). As expected, the posterior probabilities evolve smoothly between consecutive hand targets (with maximum value mostly achieved when the target is reached). This motivates an explicit modeling of the dynamic of the extracted features, as performed by the HMM-GMM decoder. Interestingly, the CNN seems to be robust to small intra-class variation, e.g. between configurations 3 (/d/) and 5 (/ʒ/) which were both correctly classified as [p, d, ʒ] while the hand shape is quite different (due to gestures co-articulation).

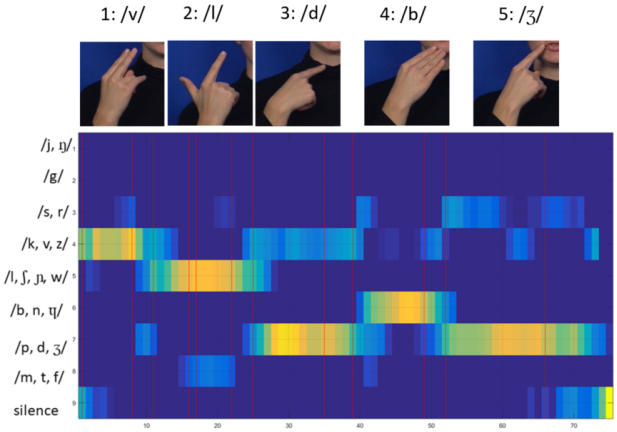

Figure 6: *Top: Sequence of target hand shapes (i.e. key frames) for the sentence "voila des bougies". Bottom: Corresponding posterior probabilities for each group of hand-shape available at the output of the CNN (blue corresponds to 0, yellow corresponds to 1).*

Then, we report in table 2 the global performance of the different proposed architectures S1, S2, and S3. First, CNN clearly outperformed PCA for all proposed architectures (e.g. $Acc = 36.5\%$ *vs.* $Acc = 61.5\%$ in S3). This tends to validate the gain of a non-linear and discriminative feature extraction technique for this particular task. Second, the S1 architecture gives much lower performance than S2 and S3. One possible explanation for this unexpected result may be related to the difference of spatial resolution between lips and hand when considering only one ROI (i.e. the hand occupies much more space than the lips). Extracting a dedicated ROI may help the CNN to balance better the information carried by lips and hand. Third, middle fusion of lips and hand modalities within the HMM-GMM decoder outperforms the early fusion strategy (e.g. with $Acc = 58.4\%$ in $S2_{CNN}$ *vs.* 61.5% in $S3_{CNN}$). Again, this tends to validate the gain of processing lips and hand independently. Fourth, the performance in terms or Acc is about 10% lower than the one regarding $Corr$. Despite the fact that the model insertion penalty is optimized, too many insertion errors remain. Indeed, this issue should be alleviated when using a language model and a pronunciation dictionary. Finally, and even if a fair comparison remains difficult since both studies are not based on the same corpus, the proposed tandem CNN-HMM gives a comparable score to the $Corr = 74.0\%$ obtained in [5]. However, let recall that [5] was based on isolated word and made use of visual artifices to help lip and hand tracking while the present study deals with continuous CS recognition from raw images.

4. Conclusions

This study investigated different CNN-HMM tandem architectures for automatic cued-speech recognition. Importantly, we focused on continuous cued-speech (i.e. connected words) and did not use any artifices used to facilitate the extraction of visual features. In its best configuration (i.e. $S3_{CNN}$), and without exploiting any dictionary or language model, the accuracy at phonetic level is 62% (with the correctness of 73%). This makes the proposed tandem architecture a good candidate for practical use. Future work will mainly focus 1) on the validation of the proposed approach on more speakers, 2) on the use of a language model to decrease the number of insertion errors, and 3) on the design of an end-to-end trainable model combining CNN-based feature extractors with recurrent neural networks.

5. Acknowledgements

The authors would like to thank Christophe Savariaux as well as Laura Machart (professional Cued-speech coder) for their help in the recording of the corpus.

6. Supplementary material

The dataset recorded for this particular study is being made publicly available on Zenodo (https://doi.org/10.5281/zenodo.1206001).

7. References

[1] R. O. Cornett, "Cued speech," *American annals of the deaf*, vol. 112, no. 1, pp. 3–13, 1967.

[2] A. A. Montgomery and P. L. Jackson, "Physical characteristics of the lips underlying vowel lipreading performance," *The Journal of the Acoustical Society of America*, vol. 73, no. 6, pp. 2134–2144, 1983.

[3] C. J. LaSasso, K. L. Crain, and J. Leybaert, *Cued Speech and Cued Language Development for Deaf and Hard of Hearing Children*. Plural Publishing, 2010.

[4] N. Aboutabit, D. Beautemps, O. Mathieu, and L. Besacier, "Feature adaptation of hearing-impaired lip shapes: the vowel case in the cued speech context." in *9th Annual Conference of the International Speech Communication Association (Interspeech 2008)*, 2008.

[5] P. Heracleous, D. Beautemps, and N. Aboutabit, "Cued speech automatic recognition in normal-hearing and deaf subjects," *Speech Communication*, vol. 52, no. 6, pp. 504–512, 2010.

[6] N. Aboutabit, "Reconnaissance de la langue française parlée complété (lpc): décodage phonétique des gestes main-lèvres." Ph.D. dissertation, Institut National Polytechnique de Grenoble-INPG, 2007.

[7] P. Heracleous, D. Beautemps, and N. Hagita, "Continuous phoneme recognition in cued speech for french," in *Proc. IEEE-EUSIPCO*, 2012, pp. 2090–2093.

[8] T. Burger, A. Caplier, and S. Mancini, "Cued speech hand gestures recognition tool," in *Proc. EUSIPCO*, 2005, pp. 1–4.

[9] A. Caplier, L. Bonnaud, S. Malassiotis, and M. G. Strintzis, "Comparison of 2D and 3D analysis for automated cued speech gesture recognition," in *9th Conference on Speech and Computer (ISCA)*, 2004.

[10] G. Potamianos, C. Neti, G. Gravier, A. Garg, and A. W. Senior, "Recent advances in the automatic recognition of audiovisual speech," *Proceedings of the IEEE*, vol. 91, no. 9, pp. 1306–1326, 2003.

[11] M. Wand, J. Koutník, and J. Schmidhuber, "Lipreading with long short-term memory," in *Proc. IEEE-ICASSP*, 2016, pp. 6115–6119.

[12] T. Hueber and G. Bailly, "Statistical conversion of silent articulation into audible speech using full-covariance HMM," *Computer Speech & Language*, vol. 36, pp. 274–293, 2016.

[13] E. Tatulli and T. Hueber, "Feature extraction using multimodal convolutional neural networks for visual speech recognition," in *Proc. IEEE-ICASSP*, 2017, pp. 2971–2975.

[14] L. Pigou, S. Dieleman, P.-J. Kindermans, and B. Schrauwen, "Sign language recognition using convolutional neural networks," in *Proc. ECCV*. Springer, 2014, pp. 572–578.

[15] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[16] K. Noda, Y. Yamaguchi, K. Nakadai, H. G. Okuno, and T. Ogata, "Lipreading using convolutional neural network." in *Proc. Interspeech*, 2014, pp. 1149–1153.

[17] A. Ephrat and S. Peleg, "Vid2speech: Speech reconstruction from silent video," in *Proc. IEEE-ICASSP*, 2017, pp. 5095–5099.

[18] J. Shi and C. Tomasi, "Good features to track," in *Proc. IEEE-CVPR*, 1994, pp. 593–600.

[19] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. IEEE-CVPR*, vol. 2, 1999, pp. 246–252.

[20] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of cognitive neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.

[21] C. Bregler and Y. Konig, "Eigenlips for robust speech recognition," in *Proc. IEEE-ICASSP*, vol. 2, 1994, pp. 669–672.

[22] F. Chollet *et al.*, "Keras," https://github.com/fchollet/keras, 2015.

[23] V. Attina, D. Beautemps, M.-A. Cathiard, and M. Odisio, "A pilot study of temporal organization in cued speech production of french syllables: rules for a cued speech synthesizer," *Speech Communication*, vol. 44, no. 1, pp. 197–214, 2004.

[24] L. Liu, G. Feng, and D. Beautemps, "Automatic temporal segmentation of hand movements for hand positions recognition in french cued speech," in *Proc. IEEE-ICASSP*, 2018, p. to appear.

[25] S. J. Young and S. Young, *The HTK hidden Markov model toolkit: Design and philosophy*. University of Cambridge, Department of Engineering, 1993.

[26] G. Gibert, G. Bailly, D. Beautemps, F. Elisei, and R. Brun, "Analysis and synthesis of the three-dimensional movements of the head, face, and hand of a speaker using cued speech," *The Journal of the Acoustical Society of America*, vol. 118, no. 2, pp. 1144–1153, 2005.

[27] L. Cappelletta and N. Harte, "Viseme definitions comparison for visual-only speech recognition," in *Proc. EUSIPCO*, 2011, pp. 2109–2113.