

Asynchrony modeling for audio-visual speech recognition

Guillaume Gravier* Gerasimos Potamianos Chalapathy Neti

IBM Thomas J. Watson Research Center
Yorktown Heights, NY 10598, USA
(—, gpotam, cneti)@us.ibm.com

ABSTRACT

We investigate the use of multi-stream HMMs in the automatic recognition of audio-visual speech. Multi-stream HMMs allow the modeling of asynchrony between the audio and visual state sequences at a variety of levels (phone, syllable, word, etc.) and are equivalent to product, or composite, HMMs. In this paper, we consider such models synchronized at the phone boundary level, allowing various degrees of audio and visual state-sequence asynchrony. Furthermore, we investigate joint training of all product HMM parameters, instead of just composing the model from separately trained audio- and visual-only HMMs. We report experiments on a multi-subject connected digit recognition task, as well as on a more complex, speaker-independent large-vocabulary dictation task. Our results demonstrate that in both cases, joint multi-stream HMM training is superior to separate training of single-stream HMMs. In addition, we observe that allowing state-sequence asynchrony between the HMM audio and visual components improves connected digit recognition significantly, however it degrades performance on the dictation task. The resulting multi-stream models dramatically improve speech recognition robustness to noise, by successfully exploiting the visual modality speech information: For example, at 11 dB SNR, they reduce connected digit word error rate from the audio-only 2.3% to 0.77% audio-visual, and, for the large-vocabulary task, from 28.3% to 19.5%. Compared to the audio-only performance at 10 dB SNR, the use of multi-stream HMMs achieves an effective SNR gain of up to 9 dB and 7 dB respectively, for the two recognition tasks considered.

1. INTRODUCTION

We have made significant progress in *automatic speech recognition* (ASR) for well-defined applications like dictation and medium vocabulary transaction processing tasks in relatively controlled environments. However, ASR performance has yet to reach the level required for speech to become a truly *pervasive user interface*. Indeed, even in “clean” acoustic environments, and for a variety of tasks, state of the art ASR system performance lags human speech

perception by up to an order of magnitude [1]. In addition, current systems are quite sensitive to channel, environment, and style of speech variations, as a number of techniques for improving ASR *robustness* have met limited success in severely degraded environments, mismatched to system training [2–4]. Clearly, novel, non-traditional approaches, that use orthogonal sources of information to the acoustic input, are needed to achieve ASR performance closer to the human speech perception level, and robust enough to be deployable in field applications. *Visual speech* constitutes a promising such source, obviously not affected by the acoustic environment and noise.

Both human speech production and perception are bimodal in nature [5, 6]. This fact has recently motivated significant interest in automatic recognition of visual speech, formally known as *automatic lipreading*, or *speechreading*. Work in this field aims at improving ASR by exploring the visual modality of the speaker’s mouth region, in addition to the traditional audio modality, thus giving rise to *audio-visual automatic speech recognition* (AVASR) systems [6–14]. There are two main problems in achieving this goal [6]: First, the design of the visual front end, i.e. how to obtain informative visual features given the video of the speaker’s face, and, second, the combination of such features with the traditional audio features. In this paper, we concentrate on the latter issue.

A number of audio-visual integration strategies appear in the literature that can be grouped into two broad categories (see Fig. 1): *Feature fusion* methods and *decision fusion* techniques. The first category consists of algorithms that combine the single modality features, by projecting them onto an audio-visual feature space, and subsequently use traditional classifiers to model the generation of the joined audio-visual observations [7–9]. In contrast, decision fusion methods combine single-modality (audio-only and visual-only) classifier outputs (decisions), such as the scores (likelihoods) of the classes of interest on basis of single-modality features.

A popular method within the decision fusion framework employs the *multi-stream hidden Markov model* (MSHMM). Such a model, originally proposed for multi-band (audio-only) speech recognition [15], has been successfully used in the AVASR literature to improve speech recognition performance [9–13]. The MSHMM linearly combines the class log-likelihoods based on the audio- and visual-only observations at a number of possible stages, thus allowing a limited *asynchrony* between the state sequences of its audio and visual HMM components. In most cases, the log-likelihoods are combined at the HMM state level, thus forcing a complete *synchrony* between the two streams. However, it is well known that, although the visual activity and the audio signal are correlated, they are not synchronous. As a matter of fact, the visual activity often precedes the audio signal by as much as 120 ms [14]. To take this asynchrony into account, the MSHMM approach can be used

*Dr. Gravier is currently at IRISA/INRIA Rennes, 35042 Rennes Cedex, France. He can be contacted at ggravier@irisa.fr

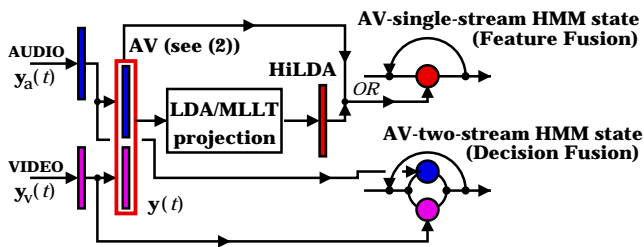


Figure 1: Two feature fusion methods versus multi-stream HMM based decision fusion (see also Figs. 2 and 4).

to combine log-likelihoods at a coarser level than the HMM state, such as the phone, syllable, or word level. Asynchronous multi-stream models are based on the use of *composite* HMMs [16], built as the “*product*” of the audio and visual HMMs (see also Fig. 2). Such product HMMs have already been proposed for AVASR [9–12], however, in these works, they have either been composed by audio-only and visual-only HMMs, which have been *independently* trained for each stream [10–12], or they have been jointly trained without the appropriate stream observation probability tying [9]. As a result, to date, comparisons between state-synchronous and state-asynchronous MSHMMs have been inconclusive and incomplete.

In this paper, we proceed to *jointly* estimate all the product HMM parameters in a single step, using the appropriate stream density tying across states, by employing the standard *expectation-maximization* (EM) HMM parameter estimation algorithm. Such a scheme ensures that the audio-visual state asynchrony within each phone is *modeled at training*, whereas stream tying guarantees that the new model has the *same number* of HMM density parameters as the state-synchronous MSHMM. In addition, in this paper, we investigate whether state-asynchrony modeling is beneficial to AVASR by considering both a small- and a large-vocabulary recognition task, namely, *connected digit* (DIGIT) ASR, and *large-vocabulary continuous speech recognition* (LVCSR) of read, dictation-like, speech. To the extent of the authors’ knowledge, such models have seldom been studied on large-vocabulary tasks.

The rest of the paper is organized as follows: In Section 2, we present in detail the MSHMM approach, discuss its implementation as a product HMM, and consider possible maximum likelihood based estimation techniques of MSHMM parameters. Section 3 describes the DIGIT and LVCSR tasks, the audio-visual feature extraction method employed, and some specifics of the recognition algorithm (decoder) and HMMs used. Experimental results are reported in Section 4, followed by our conclusions in Section 5.

2. THE MULTI-STREAM HMM

In this section, we briefly review the multi-stream hidden Markov model in the framework of audio-visual speech recognition, and we discuss strategies for estimating its parameters.

2.1 Preliminaries

Given an audio-visual utterance, we denote the extracted audio and visual features at time t by $\mathbf{y}_s(t)$, where $s = a, v$ denotes the audio or visual modality (stream), respectively. Such features are obtained as described in Section 3.2, below, and it is assumed that they are extracted at a common rate (frequency), i.e., that $\mathbf{y}_a(t)$ and $\mathbf{y}_v(t)$ are time-synchronous.

Classical single-stream HMMs are used to model sequences of audio- or visual-only features, where the state-conditional density

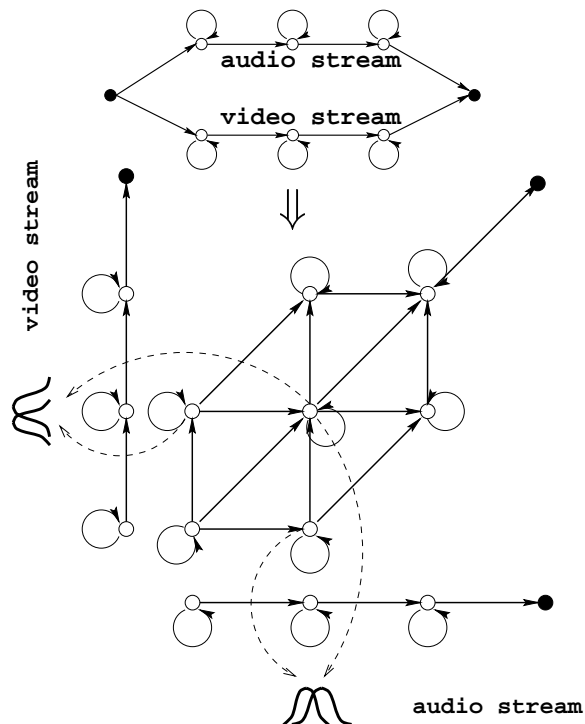


Figure 2: *Upper*: Multi-stream HMM consisting of three audio and three video HMM states. *Lower*: The corresponding product (composite) HMM that limits the possible asynchrony between the audio and visual state sequences to one state. The class-conditional observation probabilities of its composite audio-visual states are given by (4), with their audio and video component densities tied along states lying on the same column, or row, respectively.

function (emission probability) is a Gaussian mixture defined as

$$Pr(\mathbf{y}_s(t) | i_s) = \sum_{k=1}^{K_s i_s} c_{s i_s k} \mathcal{N}_{d_s}(\mathbf{y}_s(t); \mathbf{m}_{s i_s k}, \Sigma_{s i_s k}), \quad (1)$$

where $s = a, v$, respectively. In (1), i_s denotes a class (state) of the single-stream HMM used in modality s , and d_s is the dimensionality of observation vector $\mathbf{y}_s(t)$. Typically, the audio and visual HMMs are considered to have an identical set of states $\{i_a\} = \{i_v\}$.

Single-stream HMMs are also used in audio-visual feature fusion to model sequences of concatenated audio-visual features

$$\mathbf{y}(t) = [\mathbf{y}_a(t)^\top, \mathbf{y}_v(t)^\top]^\top \in \mathbb{R}^d \quad (2)$$

of dimension $d = d_a + d_v$ (concatenative feature fusion), or of any transformation of (2), such as the hierarchical discriminant feature fusion [8] (see also Fig. 1 and Section 3.2).

2.2 MSHMMs for audio-visual modeling

The principle of the multi-stream HMM is to model independently each stream between two pre-determined synchronization points, using a number (here, two) of single-stream HMMs of form (1). In this study, the synchronization points are taken to be the *phone boundaries*. At every phone boundary, the log-likelihoods of each stream are linearly combined to get the phone posterior. Such a model can be implemented as a product, or composite, HMM, as illustrated in Fig. 2. In this model, the observation log-likelihood conditioned on an audio-visual state i , that has been composed



Figure 3: Eight audio-visual database example subjects.

from the audio and visual stream states i_a and i_v , respectively, is given by

$$\ln \mathcal{L}(\mathbf{y}(t)|i) = \lambda_a \ln Pr(\mathbf{y}_a(t)|i_a) + \lambda_v \ln Pr(\mathbf{y}_v(t)|i_v). \quad (3)$$

Equivalently, the product HMM emission “score” (or, state-conditional likelihood) is given by (see also (1))

$$\mathcal{L}(\mathbf{y}(t)|i) = \prod_{s \in \{a, v\}} \left[\sum_{k=1}^{K_{s i_s}} c_{s i_s k} \mathcal{N}_{d_s}(\mathbf{y}_s(t); \mathbf{m}_{s i_s k}, \Sigma_{s i_s k}) \right]^{\lambda_s}. \quad (4)$$

In (3) and (4), λ_s , where $s = a, v$, denote the stream likelihood combination weights (exponents). These determine the contribution of each modality to the joint audio-visual conditional log-likelihood, and they can be used to model the reliability of each stream in deciding about the hidden class i . In general, such weights can also be a function of class i and the time instant t . In this paper, however, we consider them to be constant within each stream, and, in addition, to satisfy $\lambda_a + \lambda_v = 1$.

It is not hard to observe that the number of states per phone in the composite HMM (3) is equal to the product of the number of audio and visual phone states (assuming left-to-right single-stream HMMs). Typically, such single-stream phone HMMs consist of 3 states, therefore the resulting MSHMM has 9 states per phone, allowing a within-phone maximum asynchrony of 2 states between its audio and visual state sequences. One can choose to reduce the maximum tolerated within-phone asynchrony of the two streams, by restricting the product model states to appropriate subsets of the Cartesian product $\{i_a\} \times \{i_v\}$. A MSHMM with a one-state maximum asynchrony is illustrated in Fig. 2, where two states are missing w.r.t. the full 9-state product model. In the extreme case of zero maximum asynchrony, and assuming that $\{i_a\} = \{i_v\}$, the product model is equivalent to the state-synchronous MSHMM, and $i = i_a = i_v$ holds.

Note that, although the number of states in the state-asynchronous MSHMM increases compared to the state-synchronous MSHMM, the number of emission density parameters *remains the same*, due to the *stream parameter tying*, inherent in (3). This is also true for the HMM *transition probabilities*, if they are set to the product of the transition probabilities of the single-stream HMMs, for example (transition probability tying).

2.3 Parameter estimation

With the exception of exponents λ_s , maximum-likelihood estimates of all other HMM parameters in (4) can be obtained using the EM algorithm. This can be achieved either separately per-stream (*independent stream training*), or all at once, using *joint training*. The first strategy consists in estimating the stream HMM parameters independently for each stream before composing the product HMM. In this case, the synchrony, or asynchrony, constraints between the two stream models are not considered during training, but only used at recognition, thus creating a mismatch between the training and testing assumptions. In contrast, the second strategy utilizes the product HMM representation of the MSHMM to di-

Table 1: The training and test sets of the two audio-visual databases used in this paper. The number of utterances, words, duration (in hours), and number of subjects are depicted for each set. Two recognition tasks are considered: Connected digits (DIGIT) and continuous, read speech (LVCSR).

Recog. task	Training set				Test set			
	Utter.	Wds.	Dur.	Sub.	Utter.	Wds.	Dur.	Sub.
DIGIT	5490	46638	8:01	50	529	4513	0:46	50
LVCSR	17111	219470	34:55	239	207	3176	0:30	26

rectly estimate all its parameters using the standard EM procedure. In this scenario, (a)synchrony constraints between the two streams are used for both parameter estimation and decoding. The difference between these two approaches lies on the E-step of the estimation algorithm.

In the AVASR literature, both independent [9, 11], and joint training schemes [9, 13] have been considered in the case of state-synchronous MSHMMs. However, for state-asynchronous MSHMMs, only independent training has been considered [9–12], whereas in the joint training algorithm of [9], stream tying has not been employed, thus resulting to trained HMMs with significantly more density parameters than the MSHMM of (4). Proper joint training of the product HMM parameters is employed for the first time in this paper.

Note that exponent values λ_s cannot be obtained using maximum-likelihood approaches. Instead, they can be estimated using discriminative training [12, 13], or by minimizing the *word error rate* (WER) on a held-out set [9, 11], as in this paper.

3. EXPERIMENTAL FRAMEWORK

In this section, we describe the two audio-visual recognition tasks (databases) considered, the audio-visual feature extraction algorithm employed, and specifics on MSHMM training and decoding.

3.1 Audio-visual databases

As already mentioned, in this paper we report AVASR experiments on two recognition tasks, namely connected digits recognition and LVCSR. For this purpose, two suitable audio-visual databases have been recently collected at the IBM T.J. Watson Research Center, under similar recording conditions. The data consist of full-face frontal video and audio of a large number of subjects (see also Fig. 3), uttering connected digit strings (7- or 10-tuples) with an 11-word vocabulary, or ViaVoiceTM scripts (continuous read speech with mostly verbalized punctuation) with a 10,400 word vocabulary [9]. In both cases, the database video is of size 704×480 pixels, interlaced, captured in color at a rate of 30 Hz (60 fields per second are available at a resolution of 240 lines), and is MPEG2 encoded at the relatively high compression ratio of about 50:1. High quality wideband audio is synchronously collected with the video at a rate of 16 kHz in an office environment at a 20 dB *signal-to-noise ratio* (SNR).

The duration and other specifics of the two datasets are depicted in Table 1. Each is partitioned into a training, test, and held-out set. The latter is only used for estimating MSHMM exponents in (4), and it is not shown in Table 1. Notice that the DIGIT task corresponds to a multi-speaker training/testing scenario, as the same 50 subjects are used for both training and testing. On the contrary, the LVCSR task is speaker-independent, as the 26 test subjects are disjoint to the 239 training set ones. It is worth mentioning that the second database is the largest audio-visual corpus collected to date, and the only one suitable for LVCSR [6, 7].

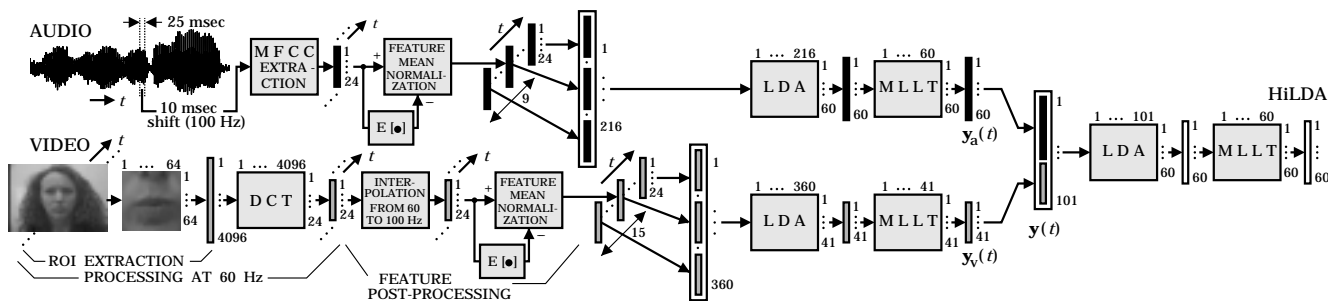


Figure 4: Feature extraction for audio-visual ASR [8] (see also Fig. 1).

To investigate the benefit of the visual modality to ASR noise-robustness, *speech “babble”* (i.e., non-stationary) noise is artificially added to the audio data at various SNRs, ranging from 20 dB (corpus original audio) to approximately -3 dB. In the experiments reported in Section 4, a matched to the noise training/testing scenario is considered, i.e., HMMs are trained at each noise level and tested on the data at the same noise condition.

3.2 Audio-visual feature extraction

We now briefly describe the audio and visual feature extraction algorithms (front ends) employed in our AVASR system, as well as a baseline technique, proposed in [8], for automatic recognition of audio-visual speech, based on feature fusion. The whole AVASR system is schematically depicted in Fig. 4.

From the audio signal, 24 mel-frequency cepstral coefficients are retained as “static” audio features, after feature mean normalization. At every frame, 9 consecutive audio features are concatenated, subsequently projected to a lower dimensional space using linear discriminant analysis (LDA), and rotated by a maximum likelihood linear transform (MLLT). The final audio features $y_a(t)$ have a dimension of $d_a = 60$, and they are extracted at a 100 Hz rate.

Given the video of the speaker’s face, sampled at 60 Hz, a normalized mouth region-of-interest (ROI) is first extracted, and subsequently compressed using a discrete cosine transform (DCT). The 24 highest energy DCT coefficients are retained, and after linear interpolation (up-sampling from 60 to 100 Hz) and mean normalization, they result to “static” visual features. Similarly to the audio features, dynamic information is obtained by concatenating consecutive “static” features over 15 frames, and by applying the LDA and MLLT transforms. The final visual feature $y_v(t)$ dimension is $d_v = 41$.

Notice that the audio and visual features are time-synchronous, extracted at 100 Hz. Therefore, a joint audio-visual feature vector $y(t)$, of dimension $d = 101$, can easily be obtained by concatenating them, as in (2). Sequences of the resulting vectors can then be modeled using the MSHMM framework of Section 2.2. For completeness, the performance of these MSHMMs will also be gauged against an alternative method for AVASR, namely a feature fusion technique that applies a second stage of LDA and MLLT data transforms on the audio-visual vector $y(t)$, and uses a single-stream HMM to model the resulting 60-dimensional features (see also Figs. 1 and 4). This feature fusion algorithm is known as *hierarchical LDA* (HiLDA), and it has been demonstrated to perform better than feature fusion using single-stream HMMs to directly model $y(t)$ [8].

3.3 HMM training and decoding

For both recognition tasks considered (DIGIT and LVCSR), we use 3-state, left-to-right phone HMMs, with context-dependent sub-phonetic HMM classes (states). These classes are obtained by means

of decision trees that cluster phonetic contexts spanning up to 5 phones to each side of the current phone, in order to better model co-articulation and improve ASR performance. The DIGIT and LVCSR decision trees are estimated using the clean audio of the corresponding database training sets, by bootstrapping on a previously developed audio HMM (and its corresponding front end) that provides data class labels by forced alignment. Subsequently, K-means clustering is used to estimate the single-stream audio HMMs, that correspond to the newly developed trees. It is by bootstrapping on these models, that the parameters of all HMMs considered in this paper are estimated (on their required front ends). The total number of the resulting context-dependent HMM states are 159 for the DIGIT task and approximately 2.8k for LVCSR, whereas all single-stream HMMs considered in the experiments (i.e., for audio-only, visual-only, or audio-visual HiLDA feature modeling) have identical number of Gaussian mixture components, namely about 3.2k and 47k for the DIGIT and LVCSR tasks, respectively.

Once decision trees and initial DIGIT and LVCSR audio HMMs are developed, we proceed to estimate the parameters of single-stream HMMs that model visual-only, as well as audio-only and audio-visual HiLDA feature sequences at the required SNR levels. We use three EM algorithm iterations for this task, with the E-step of the first iteration employing the initial audio HMM (for bootstrapping). Subsequently, we proceed to estimate MSHMMs using the audio-only and visual-only single-stream HMMs, as outlined in Section 2.3. In particular the following schemes are used:

(i): In the first method, we consider independent MSHMM training, by composing the required MSHMMs from the two single-stream HMMs as in (4) (note that $\{i_a\} = \{i_v\}$). We denote this scheme by $n=3$, to indicate the three EM iterations used to separately train each single-stream HMM. We then choose exponents λ_a and λ_v that minimize the WER using the resulting MSHMM on the held-out set.

(ii): The second scheme jointly estimates MSHMM parameters. It first composes the required MSHMM from the audio- and visual-only single-stream HMMs, which have been independently obtained after the first iteration of the EM algorithm (note that the same model has been used at the E-step for bootstrapping both). Subsequently, two more EM iterations are performed using the MSHMM with exponents set to the ones estimated in the previous scheme. We denote this scheme by $n=12$.

(iii): Finally, a third scheme considers independent training of the MSHMM parameters, followed by their joint re-estimation. It is similar to the previous scheme, with the main difference being the use of audio- and visual-only single-stream HMMs obtained at the third (instead of the first) EM iteration, in order to compose the initial MSHMM (identical to the one of the first scheme). Two additional joint MSHMM estimation iterations of the EM algorithm are used. The method is denoted by $n=32$.

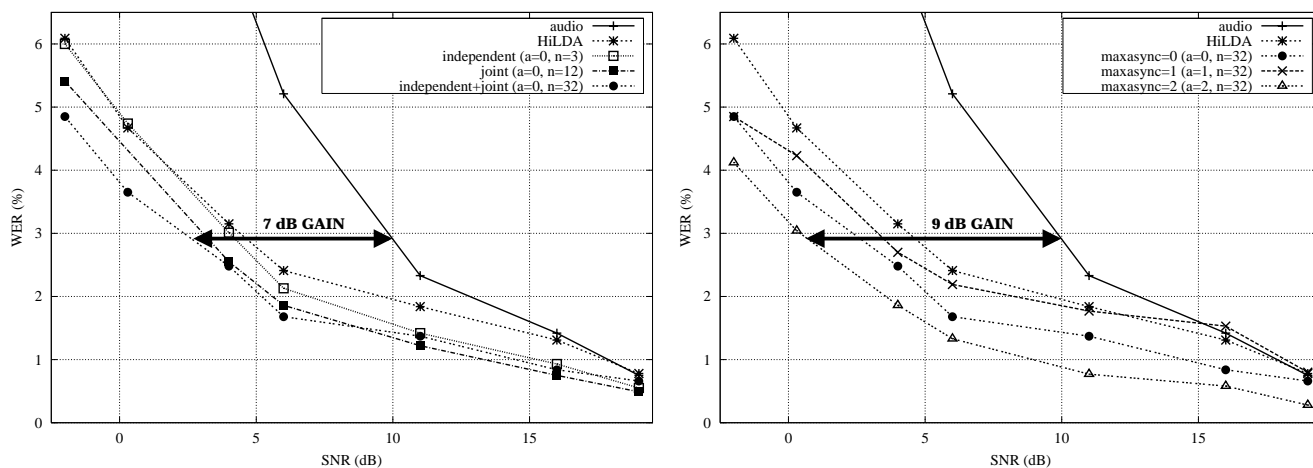


Figure 5: Results on the DIGIT recognition task. Test set audio-only and audio-visual WERs using HiLDA feature fusion and MSHMM based decision fusion are depicted against the audio channel SNR. *Left*: WER for different training strategies with state-synchronous MSHMMs. *Right*: WER for different levels of asynchrony for jointly trained MSHMMs.

All three schemes are used for various levels of maximum allowed within-phone state asynchrony between the audio and visual state sequences of the MSHMM. Since the single-stream HMMs are 3-state phone models, there can be a maximum allowed asynchrony (denoted by “a”) of 0 (state-synchronous MSHMM), 1, or 2 states. Notice that in the state-asynchronous MSHMM cases $a=1,2$, the transition probabilities in the product model can be tied to the transition probabilities in the stream models in the same way to the state conditional densities are tied. However, unless otherwise stated, transition probability tying is not used in the experiments reported in this paper.

Finally, recognition of matched test data (same SNR as in training) is carried out. For the DIGIT task, decoding is based on a simple digit-word loop grammar (with unknown string length), whereas for LVCSR, a trigram language model is used. In both cases, a two-stage stack decoding algorithm is employed, that uses a fast match followed by a detailed match [17]. Preliminary experiments on state-asynchronous MSHMMs have shown that the best results are obtained with a fast-match selection of candidate words based on state-synchronous MSHMMs rather than on asynchronous product models. The latter are solely used in the subsequent detailed match. Apart from the fact that different models are used, the fast and detailed matches are carried out as described in [17].

4. EXPERIMENTAL RESULTS

We now present a number of ASR results that employ the experimental framework discussed of Section 3. We first describe ASR on the DIGIT task, followed by LVCSR.

4.1 Results on the DIGIT task

A first experiment aims at comparing the independent and joint parameter estimation strategies discussed in Sections 2.3 and 3.3. WERs as a function of the SNR are reported in Fig. 5 (left graph) for state-synchronous MSHMMs using the three different parameter estimation schemes of Section 3.3: Independent ($n=3$), joint ($n=12$) and independent followed by a joint re-estimation ($n=32$). Results for the feature fusion (HiLDA) and for audio-only HMMs are also depicted. Not surprisingly, AVASR exhibits dramatically improved noise-robustness w.r.t. audio-only ASR. In addition, state-synchronous MSHMMs outperform HiLDA based feature fusion.

Notice that, below 7 dB SNR, the best results are obtained by scheme $n=32$, i.e., independent parameter estimation followed by a joint re-estimation. The resulting MSHMM provides a 7 dB *effective SNR gain* in ASR, compared to the audio-only performance at 10 dB SNR (see Fig. 5). For less degraded audio (above 7 dB SNR), all three parameter estimation schemes perform similarly well.

Next, we study the influence of allowing MSHMM within-phone asynchrony. Results are depicted in Fig. 5 (right graph) for maximum allowed asynchrony of $a=0,1,2$. In view of our previous results, the MSHMMs are obtained using the combined parameter estimation scheme, $n=32$. Results clearly demonstrate that increasing the allowed asynchrony to $a=2$ improves AVASR performance. It is worth noticing, that even at 20 dB (original database clean audio), the state-asynchronous MSHMM ($a=2$) reduces WER from the audio-only 0.75% to an audio-visual 0.28% (if the $n=3$, independent training scheme is used, the WER becomes 0.40%). Overall, the state-asynchronous MSHMM provides a 9 dB effective SNR gain (see Fig. 5).

4.2 Results on the LVCSR task

The experiments described in the previous section are repeated on the LVCSR task and results are reported in Fig. 6. As for the DIGIT task, state-synchronous MSHMMs demonstrate increased robustness to noise over the audio-only and HiLDA based feature fusion, yielding a 7 dB effective SNR gain, compared to the audio-only performance at 10 dB SNR. Also, similarly to Section 4.1, joint estimation ($n=12$) of MSHMM parameters outperforms their independent estimation ($n=3$). In contrast to the DIGIT task however, joint re-estimation following independent training ($n=32$) significantly hurts performance. Similar observations hold for the state-asynchronous MSHMMs, as well. A possible explanation for this discrepancy between the DIGIT and LVCSR tasks is that for the latter, when the audio and visual HMMs are independently trained, they may converge to very different state alignments of the data, due to the much higher confusability present in the visual stream, compared to the DIGIT task. Forcing the alignments to be synchronous at the phone boundary level when jointly re-estimating the parameters may yield a sub-optimal solution because the initial (stream) models are not suited for this resynchronization. We have not verified this hypothesis so far.

A second observation is that for LVCSR, allowing MSHMM state-asynchrony degrades audio-visual speech recognition. This is

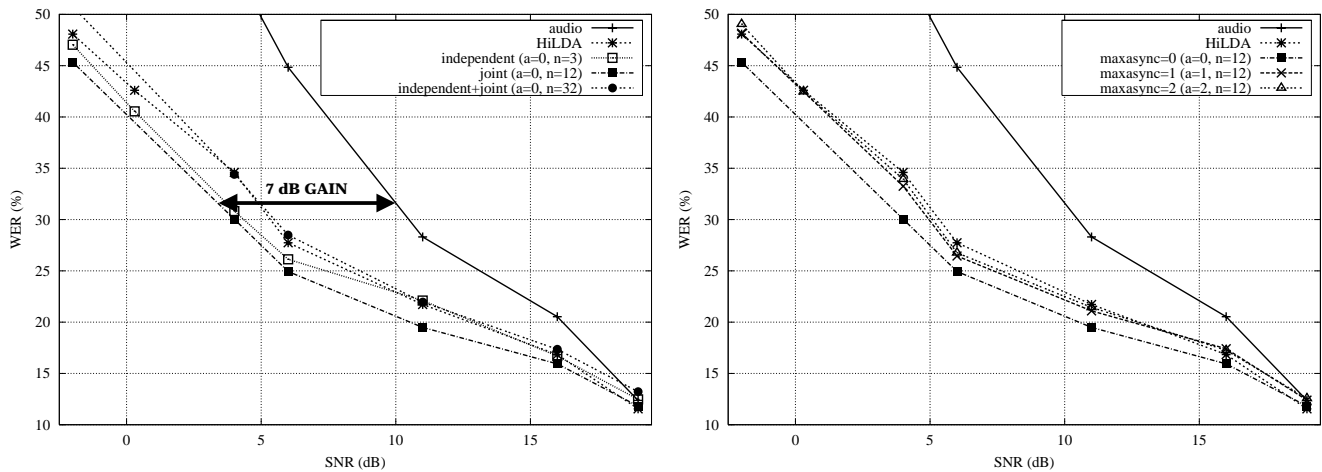


Figure 6: Results on the LVCSR task. *Left*: WER for different training strategies with state-synchronous MSHMMs. *Right*: WER for different levels of asynchrony for jointly trained MSHMMs.

clearly demonstrated in Fig. 6 (right graph), where, in view of the previous remarks, all MSHMMs depicted (for $a=0,1,2$) are jointly trained ($n=12$). Together with our earlier observations, this performance degradation indicates that the phone boundary may not be an appropriate resynchronization point for a visually confusable task such as LVCSR.

Finally, we briefly visit the issue of tying transition probabilities when jointly training ($n=12$) MSHMMs with $a=1$, or 2. Such tying affects performance only slightly: For $a=2$, and at 20 dB SNR, transition probability tying degrades WER from 12.6% (when no tying is present) to 12.9%, and from 33.0% to 33.9% at 4 dB SNR.

5. SUMMARY

In this paper, we considered audio-visual speech recognition by means of multi-stream hidden Markov models. We investigated various degrees of allowed audio-visual state asynchrony of these models, as well as a number of schemes for estimating their parameters. We reported experiments on both a small-vocabulary (connected digits) recognition task and on a more complex, large-vocabulary continuous speech recognition task. On both tasks, we demonstrated a significant advantage of the state-synchronous multi-stream approach over a baseline feature fusion approach, and observed that joint training of multi-stream HMM parameters outperforms their independent estimation. Allowing state asynchrony further improved connected digit recognition, however it degraded performance on the large-vocabulary continuous speech recognition task. For this more complex task with a highly confusable vocabulary, it is possible that the audio and visual alignments differ in more than a phone length, illustrating the fact that phone boundaries may not constitute a proper synchronization point.

6. REFERENCES

- [1] R.P. Lippmann, "Speech recognition by machines and humans," *Speech Comm.*, 22(1):1-15, 1997.
- [2] H. Hermansky and N. Morgan, "RASTA processing of speech," *IEEE Trans. Speech Audio Process.*, 2(4):578-589, 1994.
- [3] C.J. Leggetter and P.C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models," *Comp. Speech Lang.*, 9:171-185, 1995.
- [4] R. Stern, A. Acero, F.-H. Liu, and Y. Ohshima, "Signal processing for robust speech recognition," In C.-H. Lee, F.K. Soong, and K.K. Paliwal, Eds., *Automatic Speech and Speaker Recognition. Advanced Topics*. Kluwer Academic Pub., pp. 357-384, 1997.
- [5] R. Campbell, B. Dodd, and D. Burnham, Eds., *Hearing by Eye II*. Psychology Press Ltd. Pub., 1998.
- [6] D.G. Stork and M.E. Hennecke, Eds., *Speechreading by Humans and Machines*. Springer, 1996.
- [7] T. Chen, "Audiovisual speech processing. Lip reading and lip synchronization," *IEEE Sig. Process. Mag.*, 18:9-21, 2001.
- [8] G. Potamianos, J. Luetttin, and C. Neti, "Hierarchical discriminant features for audio-visual LVCSR," *Proc. Int. Conf. Acous. Speech Sig. Process.*, pp. 165-168, 2001.
- [9] C. Neti, G. Potamianos, J. Luetttin, I. Matthews, H. Glotin, D. Vergyri, J. Sison, A. Mashari, and J. Zhou, "Audio-visual speech recognition," *Final Workshop 2000 Report*, Center for Language and Speech Processing, The Johns Hopkins University, 2000.
- [10] M.J. Tomlinson, M.J. Russell, and N.M. Brooke, "Integrating audio and visual information to provide highly robust speech recognition," *Proc. IEEE Int. Conf. Acous. Speech Sig. Process.*, pp. 821-824, 1996.
- [11] S. Dupont and J. Luetttin, "Audio-visual speech modeling for continuous speech recognition," *IEEE Trans. Multimedia*, 2:141-151, 2000.
- [12] S. Nakamura, "Fusion of audio-visual information for integrated speech processing," In J. Bigun and F. Smeraldi, Eds., *Audio and Video-Based Biometric Person Authentication*. Springer-Verlag, pp. 127-143, 2001.
- [13] G. Gravier, S. Axelrod, G. Potamianos, and C. Neti, "Maximum entropy and MCE based HMM stream weight estimation for audio-visual ASR," *Proc. Int. Conf. Acous. Speech Sig. Process.*, pp. 853-856, 2002.
- [14] C. Bregler and Y. Konig, "'Eigenlips' for robust speech recognition," *Proc. IEEE Int. Conf. Acous. Speech Sig. Process.*, pp. 669-672, 1994.
- [15] H. Bourlard and S. Dupont, "A new ASR approach based on independent processing and recombination of partial frequency bands," *Proc. Int. Conf. Spoken Lang. Process.*, pp. 426-429, 1996.
- [16] P. Varga and R.K. Moore, "Hidden Markov model decomposition of speech and noise," *Proc. Int. Conf. Acous. Speech Sig. Process.*, pp. 845-848, 1990.
- [17] L. Bahl, S. De Gennaro, P. Gopalakrishnan, and R. Mercer, "A fast approximate acoustic match for large vocabulary speech recognition," *IEEE Trans. Speech Audio Process.*, 1(1):59-67, 1993.