Multi Channel Sequence Processing

Samy Bengio [a] Hervé Bourlard [a,b]

IDIAP–RR 05-04

January 2005

submitted for publication

[a] IDIAP Research Institute, Martigny, Switzerland, {bengio,bourlard}@idiap.ch
[b] Swiss Federal Institute of Technology at Lausanne (EPFL), Switzerland

IDIAP Research Institute www.idiap.ch
Rue du Simplon 4 P.O. Box 592 1920 Martigny − Switzerland
Tel: +41 27 721 77 11 Fax: +41 27 721 77 12 Email: info@idiap.ch

IDIAP Research Report 05-04

Multi Channel Sequence Processing

Samy Bengio Hervé Bourlard

January 2005

submitted for publication

Abstract. This paper summarizes some of the current research challenges arising from multi-channel sequence processing. Indeed, multiple real life applications involve simultaneous recording and analysis of multiple information sources, which may be asynchronous, have different frame rates, exhibit different stationarity properties, and carry complementary (or correlated) information. Some of these problems can already be tackled by one of the many statistical approaches towards sequence modeling. However, several challenging research issues are still open, such as taking into account asynchrony and correlation between several feature streams, or handling the underlying growing complexity. In this framework, we discuss here two novel approaches, which recently started to be investigated with success in the context of large multimodal problems. These include the asynchronous HMM, providing a principled approach towards the processing of multiple feature streams, and the layered HMM approach, providing a good formalism for decomposing large and complex (multi-stream) problems into layered architectures. As briefly reported here, combination of these two approaches yielded successful results on several multi-channel tasks, ranging from audio-visual speech recognition to automatic meeting analysis.

1 Introduction

Given the proliferation of electronic recording devices (cameras, microphones, EEGs, etc) with ever cheaper, and ever increasing processing speed, storage, and bandwidth, together with the advances in automatically extracting and managing information recorded from these devices (such as speech recognition, face tracking, etc), it becomes more and more feasible to simultaneously capture a same event (or multiple events) with several devices, generating richer and more robust sets of feature-streams.

Modeling such data coming from multiple channels (thus resulting in multiple observation streams) is the goal of *multi-channel sequence processing*. Examples of practical applications of this field are numerous, such as audio-visual speech recognition, which can be more robust to ambient noise than only using an audio stream. While several statistical models were presented recently in the literature to cope with this growing amount of data accessible in parallel, several open research problems are still to be solved. The purpose of this paper is thus to discuss some of these solutions, and specifically addressing two important issues, i.e., *asynchrony* (when the feature streams are supposed to be piecewise stationary, but with different stationary properties) and *complexity* (when it is furthermore necessary to split the problem into several multi-stream sub-problems).

The outline of the paper is as follows. Section 2 justifies the need for multi-channel sequence processing by discussing some of the numerous applications that require such a framework. Section 3 reviews some of the current models used in the literature. Section 4 shows that despite all these models, there is still room for several improvements. Section 5 proposes a model to handle temporal asynchrony between channels, while Section 6 proposes a principled approach to control the complexity of multi-channel sequence processing through “optimal” hierarchical processing.

2 Some Applications

Several tasks that are currently handled with only one stream of information could in fact benefit from the addition of other parallel streams. Furthermore, like in speech recognition (as well as video processing), it becomes more and more usual to apply different feature extraction techniques to the same signal, resulting in multiple feature streams

For instance, in *audio-visual speech recognition*, the audio signal is typically complemented by the video recording of the face (and thus the lips) of the person. It has already been shown [10, 2] that if the resulting audio and visual feature streams are properly modeled, such a multi-channel approach will significantly help in recognizing the speech utterances under noise conditions. Similar settings have also been used successfully for *audio-visual person authentication* [2]. In fact, even using only one raw source of information can yield better results in a multi-channel setting, e.g., using multiple sampling rates (*multi-rate*) or feature extraction (*multi-stream*) techniques, as already demonstrated for the task of speech recognition [16].

The field of *multimedia analysis*, which includes analysis of news, sports, home videos, meetings, etc, is very rich and these events are often recorded with at least two streams of information (audio and video) and sometimes more (as for the meeting scenario described later in this paper), and may contain complex human human interactions [15]. These multimedia documents also give rise to other applications such as *multimodal tracking of objects/humans* [12]. Furthermore, as the quantity of such archived documents grows, it becomes important to develop *multimedia document retrieval* systems [21, 24] to find relevant documents based not only on their textual content but also on their joint visual and audio content.

Finally, numerous multi-channel sequence processing processing also appear in the context *wearable computers* [14], aiming at assisting people in various everyday activities (e.g., life saving, security, health monitoring, mobile web services) by using small devices such as cameras, microphones (e.g., recording all what you see and all what you hear), and multiple extra sensors (e.g., recording diverse physiological signals), etc.

In all the above applications, multi-channel processing presents several challenges. As already mentioned earlier, we first have to develop new sequence recognition strategies accommodating multiple frame rates, asynchrony, correlation between stream, etc. One solution to this problem, referred to as "Asynchronous HMM" (AHMM) will be discussed in the paper (Section 5). Furthermore, multi-channel processing may also impact differently the different levels of information that we aim at extracting from the observation streams. While AHMM can be well suited to classify sequential patterns into "low level" classes, they may not be appropriate, or easily tractable (because of training data and complexity issues), when one aims at extracting higher level information, such as semantic classes. In this case, it may be necessary to use a "hierarchical HMM" approach, where each "HMM layer" will use different types of multiple observation streams (possibly resulting of the previous HMM layer). This layered approach will be discussed in Section 6.

3 Notation and Models

Several models have already been proposed in the literature to handle multi-channel applications. We briefly discuss here some of the most successful approaches, using a unified notation. Let us denote an observation sequence $\mathbf{O}$ of T feature vectors as

$$\mathbf{O} = (\mathbf{o}_1, \mathbf{o}_2, \ldots, \mathbf{o}_T), \tag{1}$$

where $\mathbf{o}_t$ is the vector of all multimodal features available at time t. In general, such a set of features can be broken down into multiple streams (associated with channels, modalities, or different preprocessing) m. We thus further define the feature vector

$$\mathbf{o}_t^m \in \mathbb{R}^{N_m}, \tag{2}$$

where N_m is the number of features for stream m, with $1 \leq m \leq M$ (the total number of observation streams). Each observation sequence is typically associated with a corresponding sequence of high level classes or "events". For instance, in speech or handwriting recognition, this would correspond to a sequence of words. The most successful types of model used to handle observation sequences are all based on a statistical framework. In this context, the general idea is to estimate, for each type of high level event $\mathbf{v}_j \in V$, the parameters θ_j of a distribution over corresponding observation sequences $p(\mathbf{O}|\theta_j)$, where $\mathbf{O}$ would correspond to the event $\mathbf{v}_j$. The most well-known solution to efficiently model such distributions is to use Hidden Markov Models (HMMs).

HMMs have been used with success for numerous sequence recognition tasks, including speech recognition [20], video segmentation [5], sports event recognition [25], and broadcast news segmentation [11]. HMMs introduce a state variable q_t and factor the joint distribution of the observation sequence and the underlying (unobserved) HMM state sequence into two simpler distributions, namely emission distributions $p(\mathbf{o}_t|q_t)$ and transition distributions $p(q_t|q_{t-1})$. Such factorization assumes an underlying piece-wise stationary process (each stationary segment being associated with a specific HMM state), and yields efficient training algorithms such as the Expectation-Maximization (EM) algorithm [9] which can be used to select the set of parameters θ_j^* of the model corresponding to event $\mathbf{v}_j$ in order to maximize the likelihood of L observation sequences:

$$\theta_j^* = \arg\max_{\theta_j} \prod_{l=1}^{L} p(\mathbf{O}_l|\theta_j). \tag{3}$$

The success of HMMs applied to sequences of events is based on a careful design of sub-models (topologies and distributions) corresponding to lexical units (phonemes, words, letters, events), and possibly semantic units (like the meeting group actions discussed in Section 6.1). Given a training set of observation sequences for which we know the corresponding labeling in terms of high level events (but not necessarily the precise alignment), we create a new HMM for each sequence as the concatenation of sub-model HMMs corresponding to the sequence of high level events. This HMM can then be trained using EM, thus adapting each sub-model HMM accordingly.

During testing, when observing a new observation sequence, the objective is simply to find the optimal sequence of sub-model HMMs (representing high level events) that could have generated the given observation sequence. Multiple algorithms have been developed to efficiently solve this problem, even in large search spaces, including stack decoders [13], or different approximations based on the well-known Viterbi algorithm [23].

While HMMs can be used to model various kinds of observation sequences, several extensions have been proposed to handle simultaneously multiple streams of observations, all corresponding to the same sequence of events [16, 10, 17]. The first and simplest solution is to *merge* all observations related to all streams into a single stream (frame by frame), and to model it using a single HMM as explained above. This solution is often called *early integration*. Note that in some cases, when the streams represent information collected at different frame rates (such as audio and video streams for instance), up-sampling or down-sampling of the streams is first necessary in order to align the streams to a common frame rate.

A better solution may be to use the *multi-stream* approach [7]. In this case, each stream is modeled separately using its own HMM. For instance, if we consider the modalities as separate streams, we would create one model $\theta^*_{m,j}$ for each event $\mathbf{v}_j$ and stream m such that

$$\theta^*_{m,j} = \arg\max_{\theta_{m,j}} \prod_{l=1}^{L} p(\mathbf{O}^m_l|\theta_{m,j}), \tag{4}$$

where $\mathbf{O}^m_l$ is the l^{th} observation sequence of stream m. When a new sequence of events needs to be analyzed, a special HMM is then created, recombining all the single stream HMM likelihoods at various specific temporal ("anchor") points automatically determined during training and decoding. Depending on these recombination points, various solutions appear. When the models are recombined after each state, the underlying system is equivalent to making the hypothesis that all streams are state-synchronous and independent of each other given a specific HMM state. This solution can be implemented efficiently and has shown robustness to various stream-dependent noises. The emission probability of the combined observations of M streams in a given state of the model corresponding to event $\mathbf{v}_j$ at time t is estimated as:

$$p(\mathbf{o}_t|q_t) = \prod_{m=1}^{M} p(\mathbf{o}^m_t|q_t, \theta_{m,j}). \tag{5}$$

One can see this solution as searching the best path into an HMM where each state i would be a combination of all states i of the single stream HMMs[1]. A more powerful recombination strategy enables some form of asynchrony between the states of each stream: one could consider an HMM in which states would include all possible combinations of the single stream HMM states. Unfortunately, the total number of states of this model would be exponential in the number of streams, hence quickly intractable. An intermediate solution, which we call *composite HMM*, considers all combinations of states in the same event only [19]. Hence, in this model, each event HMM j now contains all possible combinations of states of the corresponding event $\mathbf{v}_{m,j}$ of each stream HMM m. The total number of states remains exponential but is more tractable, when the number of states of each stream remains low as well as the number of streams. The underlying hypothesis of this intermediate solution is that all streams are now event-synchronous instead of state-synchronous.

Several other approaches to combine multiple streams of information have been proposed in the literature, but generally suffer from an underlying training or decoding algorithm complexity which is exponential in the number of streams. For instance, *Coupled Hidden Markov Models* (CHMMs) [8] can model two concurrent streams (such as one audio and one video stream) with two concurrent

[1] Note that this solution forces the topology of each single stream to be the same.

HMMs where the transition probability distribution of the state variable of each stream depends also on the value of the state variable of the other stream at the previous time step. More formally, let q and r be respectively the state variables of both streams, then CHMMs model transitions according to $p(q_t{=}i|q_{t-1}{=}j, r_{t-1}{=}k)$ and $p(r_t{=}i|r_{t-1}{=}j, q_{t-1}{=}k)$. While the exact training algorithm for such a model quickly becomes intractable when extended to more than 2 streams, an approximate algorithm which relaxes the requirement to visit every transition (termed the N-heads algorithm) was proposed in [8], and can be tractable for a small number of streams.

Two additional approaches have been proposed recently, and will be the focus of Sections 5 and 6. These are the *Asynchronous HMM* [1], that can handle asynchrony between streams, and the *Layered HMM* [26, 6] than can help in constraining the model according to levels of prior knowledge.

4 Challenges

While there are already several models proposed in the literature to cope with multi channel sequence processing, we believe that there are still several research challenges that have not been adequately addressed yet, including:

1. **How to handle more than two streams?** Most solutions that model the joint probability of the streams need in general exponential resources with respect to the number of streams, the number of states of each underlying Markov chain, or the size of each stream. This practically means that handling more than two streams is already a challenge. One possible alternative is to limit the search space through the use of reasonable heuristics, which should depend on *a priori* knowledge on the interdependencies of the streams.

2. **How to handle learning in high dimensional spaces?** The observation space (the total number of observed features per time step) grows naturally with the number of streams. Furthermore, it is often the case that the total number of parameters of the model grows linearly or more with the number of observations (for instance if the conditional observation distributions are modeled with Gaussian Mixture Models). Hence, one has to fight the well-known *curse of dimensionality* [4].

3. **How to handle long term temporal dependencies?** This problem deals with sequential data where one needs to relate information observed at time t with information observed at time $t + k$ where k is rather large. It has been shown [3] that this becomes exponentially difficult with k when no structural knowledge is built *a priori* in the model. Hence, in order for multi channel processing to be successful, an appropriate structure is necessary.

4. **Joint feature extraction and heterogeneity of sources.** In current systems involving multiple streams of information, features used to represent each stream are extracted independently. On the other hand, if one agrees that there may be some correlation between the streams, one should therefore devise joint feature extraction techniques, which should then yield more robust performance. However, what should we then do with streams of different nature (such as the slides of a presentation, together with the video of the person performing the same presentation)?

5. **How to handle different levels of *a priori* knowledge constraints?** It has been known for decades that in order to obtain good speech recognition performance, one has to constrain the recognition model with a good language model, that only permits valid and probable sequences of words to be recognized. The same idea should thus be applied to other domains, such as videos, which contain rich high level information that should be constrained somehow. Several levels of description should thus be used in such language model; for instance, a visual scene could be described by the pixels of the image, the persons present in the image, the action taking place, the body language, etc. For each of these levels, a probabilistic model of what is possible and what is not should therefore be trained. Furthermore, one should devise **multi channel**

language models in order to take into account information coming from several streams at the same time.

6. **Asynchrony between streams.** Let us consider the simplest multi-channel case, with 2 streams, and let us assume that these 2 streams describe the same sequence of 3 “events” (classes) A, B and C. Furthermore, let us assume, as illustrated in Figure 1, that the best piecewise stationary alignment of each stream to the sequence A-B-C would not coincide temporally with each other (which we refer to “stream asynchrony”). In such a case (which is discussed in more details in Section 5), a naive solution to try to model the joint probability of the two streams (e.g., applying early integration) would need an exponential number of states (with respect to the number of streams), as depicted in the third line of Figure 1. A better solution, depicted in the fourth line of Figure 1, would *stretch* or *compress* the streams along a single HMM model with the goal to *re-align* them during training and decoding. Such a model is described in Section 5.

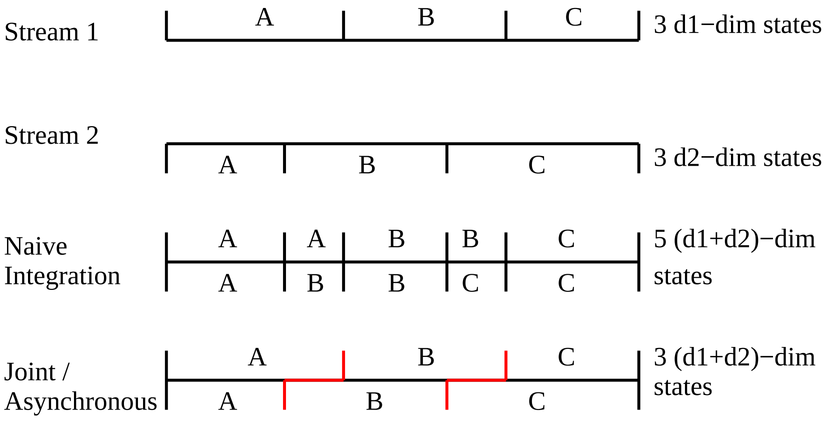

Figure 1: Complexity issue with asynchronous streams.

7. **Available benchmark datasets for evaluation**. One of the reasons of the steady progress of speech recognition has been the ever increasing availability of larger and larger realistic labeled datasets, and the yearly organization of international competitions. It is well known that this is a key point for progress in any scientific research field. However, to date, very little material has been recorded and properly annotated for multi channel sequence processing. Audio-visual speech recognition and person authentication are probably the fields where most available databases can be found. What about other scenarios, such as multimedia analysis, multimodal surveillance, etc? In Section 6, we describe a first initiative of such a benchmark database available for the meeting scenario.

5 Handling Asynchrony

Properly modeling asynchrony and correlation between multiple observation streams is thus a challenging problem. However, as a matter of fact, there are multiple evidences of real life applications involving several asynchronous streams. For instance, audio-visual speech recognition usually exhibits asynchrony. Indeed, the lips of a person often start moving earlier than any sound is uttered, mainly because the person is preparing to utter the sound. Another example is the *speaking and pointing* scenario, where a person complement the speech signal with a pointing gesture (to a point of interest). In this case, of course, although the two streams are related to the same high-level event, the pointing event will usually never occur exactly at the same time as the vocal event. One last example of asynchrony: in a news video, there is almost always a variable delay between the moment when the newscaster says the name of a public personality and the moment when the personality's picture actually appears on the screen.

One can think of several other instances involving asynchrony between streams, and there is thus a need to model this phenomenon in a principled way. As described below, such a solution, referred to as *Asynchronous HMM* was recently proposed.

5.1 The Asynchronous HMM

Let us consider the case where one is interested in modeling the joint probability of two asynchronous streams, denoted here $\mathbf{O}^1$ of length T_1 and $\mathbf{O}^2$ of length T_2 with $T_2 \leq T_1$ without loss of generality[2]. We are thus interested in modeling $p(\mathbf{O}^1, \mathbf{O}^2)$. Following the ideas introduced for HMMs, we represent this distribution using a hidden variable Q which represents the (discrete) *state* of the generating system, which in our case is synchronized with the longest sequence $\mathbf{O}^1$.

Moreover, since we know that $\mathbf{O}^2$ is smaller than $\mathbf{O}^1$, let the system always emit $\mathbf{o}_t^1$ at time t but only sometimes emit $\mathbf{o}_s^2$ at time t, with $s \leq t$. Let us define $\tau_t{=}s$ as the fact that o_t^1 is emitted at the same time as o_s^2; τ can thus be seen as the alignment between $\mathbf{O}^1$ and $\mathbf{O}^2$. Hence, an Asynchronous HMM (AHMM) [1] models $p(\mathbf{O}^1, \mathbf{O}^2, Q, \tau)$.

Using these hidden variables, and using several reasonable independence assumptions, we can factor the joint likelihood of the data and the hidden variables into several simple conditional distributions:

- $P(q_t{=}i|q_{t-1}{=}j)$, the probability to go from state j to state i at time t,
- $p(\mathbf{o}_t^1, \mathbf{o}_s^2|q_t{=}i)$, the joint emission distribution of $\mathbf{o}_t^1$ and $\mathbf{o}_s^2$, while in state i at time t,
- $p(\mathbf{o}_t^1|q_t{=}i)$, the emission distribution of $\mathbf{o}_t^1$ only, while in state i at time t,
- $P(\tau_t{=}s|\tau_{t-1}{=}s-1, q_t{=}i, \mathbf{o}_{1:t}^1, \mathbf{o}_{1:s}^2)$, the probability to emit on both sequences while in state i at time t.

We showed in [1] that using these simple distributions, new algorithms could be developed to (1) estimate the *joint likelihood* of the two streams, (2) *train a model* to maximize the joint likelihood of pairs of streams, and (3) jointly estimate the *best sequence of states* Q and the best alignment between pairs of streams.

Furthermore, one can still constrain the model to consider only reasonable alignments, e.g., integrating some minimum and maximum asynchrony between the streams. Using this constraint and denoting N_q the number of states of the model, the training and decoding complexity become $\mathcal{O}(N_q^2 \cdot T_1 \cdot k)$, which is only k times the usual HMM complexity.

5.2 Audio-Visual Speech Recognition

The proposed AHMM model was applied to several tasks, including audio-visual speech recognition and speaker verification [2], as well multi-channel meeting analysis [26]. We report here results on the M2VTS database [18] for the task of audio-visual speech recognition, where the speech features where standard Mel-Frequency Cepstral Coefficients (MFCCs), while the visual features where shapes and intensities around the mouth region, obtained by lip tracking. In order to evaluate the robustness of audio-visual speech recognition, various levels of noise were injected into the audio stream during decoding, while training was always done using clean audio only. The noise was taken from the Noisex[3] database [22], and added to the speech signal injected to reach segmental signal-to-noise ratios (SNR) of 10dB, 5dB and 0dB.

[2] Since all the reasoning below can easily be generalized to sequences (even of the same length) where the warping (stretching and compressing) can occur at different instances in the different streams.

[3] We took the *stationary speech noise*.

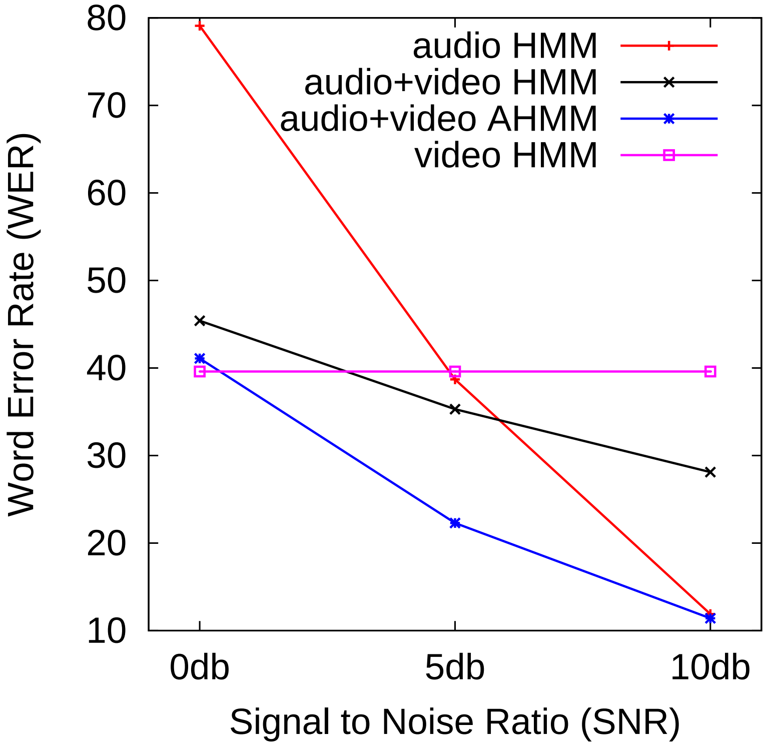

Figure 2: Word Error Rates (in percent, the lower the better), of various systems under various noise conditions.

Asynchronous HMMs were compared to classical HMMs using only the audio stream, only the video stream, or both streams combined using the *early integration* scheme. Figure 2 presents the results in terms of *Word Error Rate* (WER), a commonly used measure in the field of speech recognition, which takes into account the number of insertions, deletions and substitutions[4]. As observed from Figure 2, the AHMM consistently yielded lower WER as soon as the noise level was significant. Actually, it did not yield significantly lower performance (using a 95% confidence interval) than the video stream alone in case of very low (0dB) SNR, while performing as well as the audio stream alone in case of "clean" speech (10dB).

An interesting side effect of the model is to provide the "optimal" alignment between the audio and the video streams, as a by-product of the decoding process. This is illustrated in Figure 3 showing the audio-visual stream alignment resulting from the AHMM decoding of a specific digit sequence corrupted with 10dB Noisex noise. As it can be seen, the alignment is far from being linear. This shows that computing and maximizing the joint stream probability using AHMM appears more informative than using a naive alignment and a normal HMM.

6 A Layered Approach

6.1 The Meeting Scenario

Automatic analysis of meetings (including, e.g., automatic modeling of human interaction in meetings by modeling the joint behavior of participants through multiple audio and visual features) is a

[4] Basically, the edit (Levenshtein) distance between the recognized and reference word sequences.

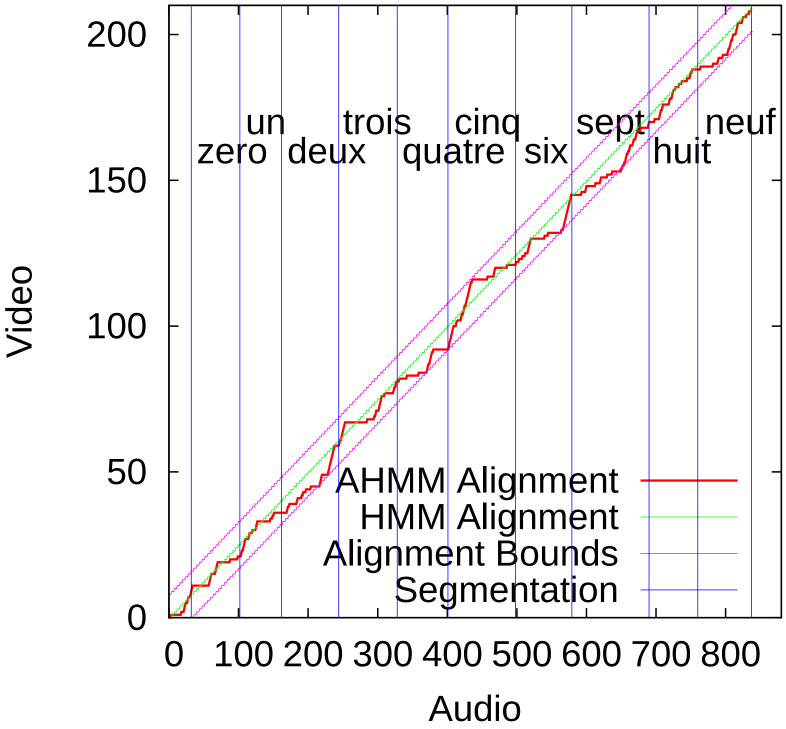

Figure 3: Alignment obtained by the model between video and audio streams on a typical sequence corrupted with a 10dB Noisex noise. The vertical lines show the obtained segmentation between the words. The alignment bounds represent the maximum allowed stretch between the audio and the video streams.

particularly challenging application of multi-channel sequence processing. It is multimodal by nature (meetings can be recorded with several cameras and microphones, as well as with other devices capturing information coming from the white-board, the slide projector, etc) and is also a rich case study of human interaction.

In [15], a principled approach to the automatic analysis of meetings was proposed, defining meetings as continuous sequences of *group actions* chosen from a predefined dictionary of actions (including, for instance, monologue, discussion, white-board presentation, with or without note-taking, agreement/disagreement, etc). This made the problem well suited for supervised learning approaches. The group actions should be mutually exclusive, exhaustive, and as much as possible unambiguous to human observers. To this end, we have collected a corpus of 60 short meetings of about 5 minutes each (30 for training, and 30 for test purposes) in a room equipped with synchronized multi-channel audio and video recorders. The resulting corpus, including annotation, is now publicly available at `http://mmm.idiap.ch`[5]. Each meeting consisted of four participants seated at a table in a typical workplace setting. Three cameras captured the participants, the projector screen and white-board. Audio was recorded using one lapel microphone per participant and an eight-microphone array located in the center of the table. The overall goal was to minimize the *Action Error Rate* (AER), similarly to what is done in speech recognition with Word Error Rate (WER), but over sequences of high level group actions. To this end, several extensions of HMMs, including AHMMs, were tested and results

[5] In the framework of the AMI European Integrated Project (`http://www.amiproject.org`) this corpus is now extended to about 100 hours of multimodal meeting data.

are reported in [15].

More recently, we proposed a multi-layered solution [26, 6] intended at simplifying the complexity of the task, based on an approach presented in [17].

6.2 A Two-Layer Approach

Let us define two sets of actions, whether they are specific to individual participants or to the group. While the overall goal is at the level of group actions, we believe that individual actions could act as a bridge between high level complex group actions and low level features, thus decomposing the problem into stages, or layers.

To this end, we defined the group action vocabulary set with the following 8 actions: *discussion*, *monologue*, *monologue+note-taking*, *note-taking*, *presentation*, *presentation+note-taking*, *white-board*, *white-board+note-taking*. Furthermore, we defined the individual action vocabulary with the following 3 actions: *speaking*, *writing*, *idle*.

Obviously, individual actions should be easier to annotate in the corpus (as being less ambiguous) and should also be easier to learn with some training data, as they are obviously more related to low level features that can be extracted from the raw multiple channels. Furthermore, knowing the sequence of individual actions of each participant, one should easily be able to infer the underlying sequence of group actions. Thus considering every meeting participant as a "multi-stream generator", each of the participant's streams should be processed by a first layer of HMMs, and the resulting HMM's outputs (likelihoods/posteriors) will then be combined by a second HMM layer yielding, higher level, group actions.

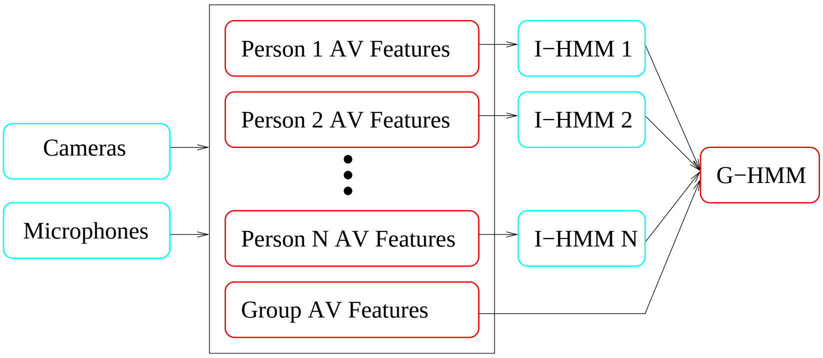

Figure 4: A two-layer approach

Figure 4 illustrates the overall strategy. Audio-visual features are first extracted for each of the meeting participants [26], complemented by more general *group-level* features. An *individual HMM* (I-HMM) is then trained for each participant, using the individual action vocabulary. To have these I-HMMs as much "participant independent" as possible, all parameters are shared among all models, yielding up to 4 times more data to train the I-HMMs. Several models were compared, including early integration, multi-stream, and asynchronous HMMs (AHMM).

We then estimate for each participant i the posterior probability of each individual task $\mathbf{v}_{i,j}$ at each time step t given the individual observation sequence up to time t, $p(\mathbf{v}_{i,j}|\mathbf{o}^i_{1:t})$. These posterior probabilities, together with *group-level features*, are then used as observations for the second layer, the *group* HMM, (G-HMM), which are trained on the group action vocabulary. Again, this G-HMM was implemented in various flavors, including early integration, multi-stream and asynchronous HMMs. Section 6.3 below further discusses this aspect and shows how these (lower level) posterior probabilities can be estimated to guarantee some form of "optimality", while preserving maximum information (i.e., avoiding local decisions) across the different layers.

Table 1: Action error rates (AER) for various systems applied to the meeting scenario.

Method		AER (%)
Single-layer	Visual only	48.20
	Audio only	36.70
	Early Integration	23.74
	Multi-Stream	23.13
	Asynchronous	22.20
Two-layer	Visual only	42.45
	Audio only	32.37
	Early Integration	16.55
	Multi-Stream	15.83
	Asynchronous	15.11

Table 1 reports the AER performance achieved by the different systems. It can be seen that (1) the two-layer approach always outperforms the single-layer one, and (2) the best I-HMM model is the Asynchronous HMM, which probably means that some asynchrony exists in this task, and is actually well captured by the model.

6.3 General Multi-Layered (Hierarchical) HMM Approach

As illustrated from the above meeting scenario, the complexity resulting from the processing of multiple channels of information, in order to extract low-level as well as high-level information (such as the analysis of multimodal meetings in terms of high level meeting actions), is often such that it will often be necessary to *break down* the problem in terms of multiple layers of sub-problems, probably using different constraints and prior knowledge information sources. The layered approach is one possible and principled solution to achieve this. Given a complex task, the goal is then to break it down into several hierarchically embedded sub-tasks, for which one can devise proper models (from enough training data), and use adequate (level specific) constraints.

We recently proposed such an approach for the task of speech recognition [6], where a general theoretical framework was proposed to compute low-level (e.g., phoneme) class posteriors, based on all the acoustic context, and to hierarchically combine those posteriors to yield higher-level (e.g., sentence) posteriors. In this approach, each layer is integrating its own prior constraints.

More precisely, a first layer, which could be an HMM or an AHMM, as in the meeting scenario, or any other model such as an Artificial Neural Network (ANN), is used to estimate posterior probabilities $p(q_t = i|\mathbf{O})$ of sub-classes i (such as phonemes, for the case of speech recognition) at each time step t given all the available information (for instance, all the acoustic sequence $\mathbf{O}$). In HMM, as well as in hybrid HMM/ANN systems, this posterior probability estimate is given by the so-called $\gamma(i,t) = p(q_t = i|\mathbf{O})$, which can be obtained by running and combining the so-called α and β recurrences through the appropriate HMM. Ideally, this HMM should embed all known lexical constraints about legal and probable sequences of phonemes. One should then use the resulting posterior probabilities (of every sub-class at every time step) as input to the next layer model, which would then estimate the posterior probabilities (again through new γ's) of higher level classes, such as words, constraining the underlying HMM model with all known language constraints that pertains to legal and probable sequences of words. In theory, this operation could be repeated up to the the level of sentences, and even to the level of summarization, always using posterior probabilities resulting from the previous layer as intermediate features.

Initial results on several speech tasks, as well as on the meeting task discussed previously, resulted in significant improvements. In [6], speech recognition results where presented on Numbers'95 (speaker independent recognition of free format numbers spoken over the telephone) and on a reduced vocabulary version (1,000 words) of the DARPA Conversational Telephone Speech-to-text (CTS) task, and both resulted in significant improvements.

7 Conclusion

This paper discussed several issues arising from the processing of complex multi-channel data, including large multimodal problems (meeting data). More specifically, this paper focused on two important issues, namely stream asynchrony and complexity of high-level decision processes. The proposed Asynchronous HMMs (AHMM) actually maximize the likelihood of the joint observation sequences through a single HMM, while also automatically allowing for stretching and/or compressing of the different streams. However, in the case of very complex problems, using AHMMs is often not enough, and the problem needs to be broken down into simpler processing blocks. A solution to this problem, referred to as "multi-layered/hierarchical HMMs" (and where each layer can integrate different levels of constraints and prior information) was also proposed and shown to be effective in modeling the joint behavior of participants in multimodal meetings. A full theoretical motivation of this approach is described in [6].

Acknowledgements

This work was partly funded (through OFES) by the European PASCAL Network of Excellence and the AMI Integrated Project. The results reported here also benefited from financial support from the Swiss National Science Foundation, through the National Center of Competence in Research IM2 (Interactive Multimodal Information Management).

References

[1] S. Bengio. An asynchronous hidden markov model for audio-visual speech recognition. In S. Becker, S. Thrun, and K. Obermayer, editors, *Advances in Neural Information Processing Systems 15*, 2003.

[2] S. Bengio. Multimodal speech processing using asynchronous hidden markov models. *Information Fusion*, 5(2):81–89, 2004.

[3] Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166, 1994.

[4] Christopher Bishop. *Neural Networks for Pattern Recognition.* Oxford University Press, London, UK, 1995.

[5] J. S. Boreczky and L. D. Wilcox. A Hidden Markov Model framework for video segmentation using audio and image features. In *Proc. of ICASSP*, volume 6, 1998.

[6] H. Bourlard, S. Bengio, M. Magimai Doss, Q. Zhu, B. Mesot, and N. Morgan. Towards using hierarchical posteriors for flexible automatic speech recognition systems. In *Proc. of DARPA EARS Rich Transcription Workshop*, 2004.

[7] H. Bourlard and S. Dupont. Subband-based speech recognition. In *Proc. IEEE ICASSP*, 1997.

[8] M. Brand. Coupled hidden markov models for modeling interacting processes. Technical Report 405, MIT Media Lab Vision and Modeling, November 1996.

[9] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum-likelihood from incomplete data via the EM algorithm. *Journal of Royal Statistical Society B*, 39:1–38, 1977.

[10] S. Dupont and J. Luettin. Audio-visual speech modeling for continuous speech recognition. *IEEE Transactions on Multimedia*, 2(3):141–151, September 2000.

[11] S. Eickeler and S. Müller. Content-based video indexing of TV broadcast news using Hidden Markov Models. In *Proc. of ICASSP*, 1999.

[12] D. Gatica-Perez, G. Lathoud, I. McCowan, and J-M. Odobez. A mixed-state i-particle filter for multi-camera speaker tracking. In *Proc. of WOMTEC*, September 2003.

[13] F. Jelinek. A fast sequential decoding algorithm using a stack. *IBM Journal of Research and Development*, 13:675–685, 1969.

[14] S. Mann. Smart clothing: The wearable computer and wearcam. *Personal Technologies*, March 1997. Volume 1, Issue 1.

[15] I. McCowan, D. Gatica-Perez, S. Bengio, G. Lathoud, M. Barnard, and D. Zhang. Automatic analysis of multimodal group actions in meetings. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3):305–317, 2005.

[16] A. Morris, A. Hagen, H. Glotin, and H. Bourlard. Multi-stream adaptive evidence combination for noise robust ASR. *Speech Communication*, 2001.

[17] N. Oliver, E. Horvitz, and A. Garg. Layered representations for learning and inferring office activity from multiple sensory channels. In *Proc. of the Int. Conf. on Multimodal Interfaces*, October 2002.

[18] S. Pigeon and L. Vandendorpe. The M2VTS multimodal face database (release 1.00). In *Proc. of the Conf. on AVBPA*, 1997.

[19] G. Potamianos, C. Neti, J. Luettin, and I. Matthews. Audio-visual automatic speech recognition: An overview. In G. Bailly, E. Vatikiotis-Bateson, and P. Perrier, editors, *Issues in Visual and Audio-Visual Speech Processing.* MIT Press, 2004.

[20] L. R. Rabiner and B.-H. Juang. *Fundamentals of Speech Recognition.* Prentice-Hall, 1993.

[21] S. Renals, D. Abberley, D. Kirby, and T. Robinson. Indexing and retrieval of broadcast news. *Speech Communication*, 32:5–20, 2000.

[22] A. Varga, H.J.M. Steeneken, M. Tomlinson, and D. Jones. The noisex-92 study on the effect of additive noise on automatic speech recognition. Technical report, DRA Speech Research Unit, 1992.

[23] A. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, pages 260–269, 1967.

[24] T. Westerveld, A. P. de Vries, A. van Ballegooij, F. de Jong, and D. Hiemstra. A probabilistic multimedia retrieval model and its evaluation. *EURASIP Journal on Applied Signal Processing*, 2(Special issue on unstructured information management from multimedia data sources), 2003.

[25] L. Xie, S.-F. Chang, A. Divakaran, and H. Sun. Structure analysis of soccer video with Hidden Markov Models. In *ICASSP*, 2002.

[26] D. Zhang, D. Gatica-Perez, S. Bengio, I. McCowan, and G. Lathoud. Modeling individual and group actions in meetings: a two-layer hmm framework. In *IEEE Workshop on Event Mining at CVPR*, 2004.