Modeling Individual and Group Actions in Meetings with Layered HMMs

Dong Zhang [1] Daniel Gatica-Perez [1]
Samy Bengio [1] Iain McCowan [1]
Guillaume Lathoud [1]

IDIAP–RR 04-33

Dec. 2004

SUBMITTED FOR PUBLICATION

[1] IDIAP, Martigny, Switzerland

IDIAP Research Report 04-33

Modeling Individual and Group Actions in Meetings With Layered HMMs

Dong Zhang Daniel Gatica-Perez Samy Bengio Iain McCowan
Guillaume Lathoud

Dec. 2004

submitted for publication

Abstract

We address the problem of recognizing sequences of human interaction patterns in meetings, with the goal of structuring them in semantic terms. The investigated patterns are inherently group-based (defined by the individual activities of meeting participants, and their interplay), and multimodal (as captured by cameras and microphones). By defining a proper set of individual actions, group actions can be modeled as a two-layer process, one that models basic individual activities from low-level audio-visual features, and another one that models the interactions. We propose a two-layer Hidden Markov Model (HMM) framework that implements such concept in a principled manner, and that has advantages over previous works. First, by decomposing the problem hierarchically, learning is performed on low-dimensional observation spaces, which results in simpler models. Second, our framework is easier to interpret, as both individual and group actions have a clear meaning, and thus easier to improve. Third, different HMM models can be used in each layer, to better reflect the nature of each subproblem. Our framework is general and extensible, and we illustrate it with a set of eight group actions, using a public five-hour meeting corpus. Experiments and comparison with a single-layer HMM baseline system show its validity.

1 Introduction

Devising computational frameworks to automatically infer human behavior from sensors constitutes an open problem in many domains. Moving beyond the person-centered paradigm [29], recent work has started to explore multi-person scenarios, where not only individual but also group actions or interactions become relevant [8, 11, 25, 1].

One of these domains is meetings. The automatic analysis of meetings has recently attracted attention in a number of fields, including audio and speech processing, computer vision, human-computer interaction, and information retrieval [14, 30, 21, 3, 28, 4, 16]. Analyzing meetings poses a diversity of technical challenges, and opens doors to a number of relevant applications.

Group activity plays a key role in meetings [30, 21], and this is documented by a significant amount of work in social psychology [18]. Viewed as a whole, a group shares information, engages in discussions, and makes decisions, proceeding through diverse communication phases both in single meetings and during the course of a long-term teamwork [18]. Recognizing group actions is therefore useful for browsing and retrieval purposes [30, 16], e.g., to structure a meeting into a sequence of high-level items.

Interaction in meetings is inherently group-based [18] and multimodal [13]. In the first place, we can view a meeting as a continuous sequence of mutually exclusive group actions taken from an exhaustive set [16, 6]. Each of these group actions involves multiple simultaneous participants, and is thus implicitly constrained by the actions of the individuals. In the second place, as the principal modality in meetings, speech has recently been studied in the context of interaction modeling [10, 31, 6]. However, work analyzing the benefits of modeling individual and group actions using multiple modalities has been limited [1, 16, 17, 26], despite the fact that actions in meetings, both at the individual (e.g., note-taking or talking), and at the group level (e.g. dictating) are often defined by the joint occurrence of specific audio and visual patterns.

In this paper, we present a two-layer HMM framework for group action recognition in meetings. The fundamental idea is that, by defining an adequate set of individual actions, we can decompose the group action recognition problem into two levels, from individual to group actions. Both layers use ergodic HMMs or extensions. The goal of the lower layer is to recognize individual actions of participants using low-level audio-visual (AV) features. The output of this layer provides the input to the second layer, which models interactions. Individual actions naturally constitute the link between the low-level audio-visual features and high-level group actions. Similar to continuous speech recognition, we perform group action recognition directly on the data sequence, deriving the segmentation of group actions in the process. Our approach is general, extensible, and brings improvement over previous work, which reflects on the results obtained on a public meeting corpus, for a set of eight group actions based on multimodal turn-taking patterns.

The paper is organized as follows. Section 2 reviews related work. Section 3 introduces our approach. Section 4 and Section 5 describe the meeting data, and the feature extraction process. Experiments and discussion are presented in Section 6. Conclusions are drawn in Section 7.

2 Related Work

Current approaches to automatic activity recognition define models for specific activities that suit the goal in a particular domain, and use statistical methods for recognition. Predominately, the recognition of individual actions [29], or interaction involving few people [25, 11] has been investigated using visual features [12, 11, 25, 29, 32], although some work on the speech community can also be categorized as interaction recognition [10, 31]. In [10], recognition of a specific kind of interaction in meetings (agreement vs. disagreement) has been addressed using both word-based features (such as the total number of words, and the number of "positive" and "negative" keywords), as well as prosodic cues (such as pause, frequency and duration). In [31], the relationship between "hot spots" (defined in terms of participants highly involved in the discussion) and dialogue acts has been examined using contextual features (such as speaker identity or type of the meeting), and lexical features (such as utterance length and perplexity).

To our knowledge, however, little work has been conducted on recognition of group-based, multi-modal actions from multiple audio-visual streams captured by cameras and microphones [1, 16, 17]. [1] described automatic discovery of "influence" in a lounge room where people played interactive debating games. The so-called influence model, a Dynamic Bayes Network (DBN) which models group interactions as a group of Markov chains, each of which influences the others' state transitions, has been applied to determine how much influence each participant has on the others. Furthermore, our previous work presented different statistical sequence models to recognize turn-taking patterns in a formal meeting room scenario, where people discuss around a table and use a white-board and a projector screen [17, 16]. Analysis of multimodal group interactions has been explicitly addressed without distinguishing actions at individual and group levels.

Regarding statistical models, most of the existing work has used Hidden Markov Models (HMMs) [27], and extensions, including coupled HMMs, input-output HMMs, multi-stream HMMs, and asynchronous HMMs (see [23] for a recent review of models). Although the basic HMM, a discrete state-space model with an efficient learning algorithm, works well for temporally correlated sequential data, it is challenged by a large number of parameters, and the risk of over-fitting when learned from limited data [24]. This situation might occur in the case of multimodal group action recognition where, in the simplest case, possibly large vectors of AV features from each participant are concatenated to define the observation space [16, 17].

The above problem is general, and has been addressed using hierarchical representations [33, 6, 24]. In [33], an approach for unsupervised discovery of multilevel video structures using hierarchical HMMs was proposed, in the context of sports videos. In this model, the higher-level structure elements usually correspond to semantic events, while the lower-level states represents variations occurring within the same event. In [6], two methods for meeting structuring from audio were presented, using multilevel DBNs. The first DBN model decomposed group actions in meetings as sequences of sub-actions, which have no explicit meanings and obtained from training process. The second DBN model processed independently features of different nature, and integrate them at higher level. In both [33, 6], the low-level actions have no obvious interpretation, and the number of low-level actions is a model parameter learned during training, or set by hand, which makes the structure of the models difficult to interpret. The other work closest to ours is [24], in which layered HMMs were proposed to model multimodal office activities involving only mainly one person at various time granularities. The lowest layer captured one video and two audio channels, plus keyboard and mouse activity features; the middle layer classifies AV features into basic events like *"speech"*,*"music"*, *"one person"*, *"nobody"* , etc. Finally, the highest layer uses the outputs of previous layers to recognize office activities with longer temporal extent. In this way, actions at different semantic levels and with different time granularities have been modeled with a cascade pyramid of HMMs. This hierarchical representation has been tested in SEER, a real-time system for recognizing typical office activities, and produced improvement over a simple baseline HMM.

The solution we present to the problem of group action recognition is novel. On one hand, unlike our previous work [16, 17], the framework presented here explicitly models actions at different semantic levels (from individual to group level) at the same time scale. This layered structure coincides with the structure of meetings as modeled in social psychology, that is, that meetings comprise individual actions and interactions

[18]. On the other hand, our ultimate goal -modeling group activity- is different than that of [24]. Since the two HMM layers are trained independently, our framework is easy to interpret and enhanced at each of the levels. Unlike [24], we have studied a number of models suitable for multimodal data. Furthermore, the number an type of sensors is also different. For our problem, the proposed work has a number of advantages, as described in the next section.

3 Group Action Recognition

In this section, we first introduce our computational framework. We then apply it to a specific set of individual and group actions. Finally, we describe some specific details.

3.1 Framework Overview

In our framework, we distinguish group actions (which belong to the whole set of participants) from individual actions (belonging to specific persons). Our ultimate goal is the recognition of group activity, and so individual actions should act as the bridge between group actions and low-level features, thus decomposing the problem in stages. The definition of both action sets is thus clearly intertwined.

Let *I-HMM* denote the lower recognition layer (individual action), and *G-HMM* denote the upper layer (group action). *I-HMM* receives as input AV features extracted from each participant, and outputs recognition results, either as soft or hard decisions (see section 3.3). In turn, *G-HMM* receives as input the output from *I-HMM*, and a set of *group features*, directly extracted from the raw streams, which are not associated to any particular individual. In our framework, each layer is trained independently, and can be substituted by any of the HMM variants that might capture better the characteristics of the data, more specifically asynchrony [2], or different noise conditions [7] between the audio and visual streams. Our approach is summarized in Figure 1.

```
1.   (Audio-Visual Feature Extraction)
     1-1. extract individual-level AV features
     1-2. extract group-level AV features
2.   (Individual Action Recognition)
     2-1. given individual features for each person, train I-HMM
          selecting best model by cross-validation
     2-2. output individual action recognition results
3.   (Group Action Recognition)
     3-1. construct a feature space by concatenating
          individual action results and group-level features
     3-2. train G-HMM selecting best model by cross-validation
     3-3. output group action recognition results
```

Figure 1: Two-layer HMM framework

Compared with a single-layer HMM, the layered approach has the following advantages, some of which were previously pointed out by [24]: (1) a single-layer HMM is defined on a possibly large observation space, which might face the problem of over-fitting with limited training data. It is important to notice that the amount of training data becomes an issue in meetings where data labeling is not a cheap task. In contrast, the layers in our approach are defined over small-dimensional observation spaces, resulting in more stable performance in cases of limited amount of training data. (2) The *I-HMMs* are person-independent, and in practice can be trained with much more data from different persons, as each meeting provides multiple individual streams of training data. Better generalization performance can then be expected. (3) The *G-HMMs* are less sensitive to slight changes in the low-level features because their observations are the outputs of the individual action recognizers, which are expected to be well trained. (4) The two layers are trained independently. Thus, we can explore different HMM combination systems. In particular, we can replace the baseline *I-HMMs* with models

Table 1: Description of group actions

Action	Description
Discussion	most participants engaged in conversations
Monologue	one participant speaking continuously without interruption
Monologue+ Note-taking	one participant speaking continuously others taking notes.
Note-taking	most participants taking notes
Presentation	one participant presenting using the projector screen
Presentation+ Note-taking	one participant presenting using projector screen, others taking notes
White-board	one participant speaking using the white-board
White-board+ Note-taking	one participant speaking using white-board, others taking notes

that are more suitable for multi-modal asynchronous data sequences, with the goal of gaining understanding of the nature of the data (see Section 3.3.1). The framework thus becomes simpler to understand, and amenable to improvements at each separate level. (5) The framework is general and extensible to recognize new group actions defined in the future.

3.2 Definition of Actions

As an implementation of the proposed framework, we define a set of group actions and individual actions in this section. On one hand, a set of group actions is defined based on multi-modal turn-taking patterns. The list is defined in Table 1. Note that we consider a “monologue” or a “presentation” as a group action, because we define it as the joint occurrence of several individual patterns (e.g., one person speaks while the others listen to her). For meeting browsing and indexing, it might be also desirable to know which specific participant is doing a monologue in the meeting. Therefore, we further divide the “monologue” action into “monologue1”, “monologue2”, etc., according to the number of participants. In a similar way, we divide the “monologue+note-taking” action into “monolgue1+note-taking”, “monologue2+note-taking”, and so on. Thus, for a four-participant meeting, a set of $N_G = 14$ group actions has been defined as: N_G = *{ discussion, monologue1, monologue1 + note-taking, monologue2, monologue2 + note-taking, monologue3, monologue3 + note-taking, monologue4, monologue4 + note-taking, note-taking, presentation, presentation + note-taking, whiteboard, whiteboard + note-taking }* . These group actions are multimodal, and commonly found in meetings. For modeling purposes, they are assumed to define a partition (i.e., the action set is non-overlapping and exhaustive). This set is richer compared to the one that we defined in [17], as it includes simultaneous occurrence of actions, like *“monologue+note-taking”* which could occur during real situations, like dictating or minute-taking. The group actions we defined here can be easily described by combinations of a proper set of individual actions defined in the following. Our framework is general, and other type of group actions could be defined. Note that high-level group actions in semantic terms (e.g. agreement / disagreement) would certainly require language-based features [10].

On the other hand, we define a small set of $N_I = 3$ multimodal individual actions which, as stated earlier, will help bridge the gap between group actions and low-level AV features. The list appears in Table 2. While the list of potentially interesting individual actions in meetings is large, our ultimate goal is recognition of the group-level actions defined in Table 1. It is interesting to note that, although at first glance one would not think of *“speaking”* or *“writing”* as multimodal, joint sound and visual patterns do occur in these cases and are useful in recognition, as the results in later sections confirm.

Finally, meeting rooms can be equipped with white-boards or projector screens which are shared by the group. Extracting features from these group devices also helps recognize group actions. They constitute the

Table 2: Description of individual actions

Action	Description
Speaking	one participant speaking
Writing	one participant taking notes
Idle	one participant neither speaking nor writing

Table 3: Relationships between group actions, individual actions and group features. Symbol “$\star$″ indicates that the white-board or projector screen are in use when the corresponding group action takes place. Symbol “/″ indicates that the number of participants for the corresponding action is not certain.

Group Actions	Individual Actions			Group Features	
	speaking	writing	idle	white-board	projector screen
discussion	>2	/	/		
monologue	1	0	/		
monologue+note-taking	1	>=1	/		
note-taking	0	>2	0		
presentation	1	0	/		$\star$
presentation+note-taking	1	>=1	/		$\star$
white-board	1	0	/	$\star$	
white-board+note-taking	1	>=1	/	$\star$	

group features described in the previous subsection. Their detailed description will be presented in section 5.

The logical relations between individual actions, group actions, and group features are summarized in Table 3. The group actions can be seen as combinations of individual actions plus states of group devices. For example, *“presentation + note-taking”* can be decomposed into *“speaking”* by one individual, with more than one *“writing”* participant, while the group device of *projector screen* is in use. Needless to say, our approach is not rule-based, but Table 3 is useful to conceptually relate the two layers.

3.3 Implementing the Two-layer Framework

In this section, we present some details about the architecture of our framework. To facilitate description, we first define the following symbols:

- $\mathbf{O}^a$: a sequence of audio-only feature vectors.
- $\mathbf{O}^v$: a sequence of visual-only feature vectors.
- $\mathbf{O}^{a+v}$: a sequence of concatenated audio-visual feature vectors.
- $\mathbf{o}_{1:t} \triangleq \mathbf{o}_1, \mathbf{o}_2, ..., \mathbf{o}_t$: a sequence (audio, visual, or audio-visual stream) up to time t.
- q_t: the HMM state at time t

3.3.1 Individual Action Models

We investigate three models for the lower-layer *I-HMM*, each of which attempts to model specific properties of the data. The investigated models are:

- *Early Integration HMM (Early Int.)*, where a basic HMM [27] is trained on combined AV features. This method involves aligning and synchronizing AV features to form one concatenated set of features which is then treated as a single stream of data. Early integration selects the set of parameters θ_i^* of the model corresponding to action i that maximizes the likelihood of L audio-visual observation sequences as follows:

$$\theta_i^* = \arg\max_{\theta_i} \prod_{l=1}^{L} P(\mathbf{O}_l^{a+v}|\theta_i). \tag{1}$$

- *Audio-visual Multi-Stream HMM (MS-HMM)*, which combines the audio-only and visual-only streams. Each stream is modeled independently. $\theta_i^* = (\theta_{i,a}^*, \theta_{i,v}^*)$ are the best model parameters for action i to maximize the likelihood of audio-only and visual-only sequences respectively.

$$\theta_{i,a}^* = \arg\max_{\theta_{i,a}} \prod_{l=1}^{L} P(\mathbf{O}_l^{a}|\theta_{i,a}). \tag{2}$$

$$\theta_{i,v}^* = \arg\max_{\theta_{i,v}} \prod_{l=1}^{L} P(\mathbf{O}_l^{v}|\theta_{i,v}). \tag{3}$$

The final classification is based on the the fusion of the outputs of both modalities by estimating their joint occurrence [7], as follows:

$$P(\mathbf{O}_l^{a+v}|q_t) = P(\mathbf{O}_l^{a}|q_t, \theta_{i,a})^{\omega} P(\mathbf{O}_l^{v}|q_t, \theta_{i,v})^{(1-\omega)}, \tag{4}$$

where the weighting factor $\omega (0 \leq \omega \leq 1)$ represents the relative reliability of the two modalities.

- *Audio-visual Asynchronous HMM (A-HMM)*, which also combines audio-only and visual-only streams, by learning the joint distribution of pairs of sequences when these sequences are not synchronized and are not of the same length or rate [2]. The A-HMM for action i models the joint distribution of the two streams by maximizing the likelihood of L observation sequences as follows:

$$\theta_i^* \quad = \quad \arg\max_{\theta_i} \prod_{l=1}^{L} P(\mathbf{O}_l^{a}, \mathbf{O}_l^{v}|\theta_i). \tag{5}$$

This is done by considering all possible alignments between sequences $\mathbf{O}_l^a$ and $\mathbf{O}_l^v$ during training and decoding.

For space reasons, the HMM models are described here briefly. Please refer to the original references for details [27, 7, 2].

3.3.2 Linking Two Layers

Obviously, a mechanism to link the two HMM layers has to be specified. There are two approaches to do so, based on different *I-HMM* outputs. Let $a^t = (a_1^t, ..., a_{N_I}^t) \in \mathbb{R}^{N_I}$ denote a vector in a continuous space of dimension equal to the number of individual actions, which indicates the degree of confidence in the recognition of each individual action at time t for a sequence $\mathbf{o}_{1:t}$.

The first approach directly outputs the probability P_k^t for each individual action model $M_k, k = 1, ..., N_I$, as input feature vector to *G-HMM*, $a_k^t = P_k^t$ for all k. We refer to it as *soft decision*.

In soft decision, the probability P_k^t of model M_k given a sequence $\mathbf{o}_{1:t}$ is computed in two steps. In the first step, we compute the probability of having generated the sequence and being in the state i at time t. We denote this probability as $\rho(i,t)$. For different *I-HMMs*, the probability $\rho(i,t)$ is computed in different ways.

- *Early integration normal HMM*:

$$\rho(i,t) = \alpha(i,t), \tag{6}$$

where forward variable $\alpha(i,t) \triangleq P(\mathbf{o}_{1:t}, q_t = i)$ in the standard Baum-Welch algorithm [27]. $\mathbf{o}_{1:t}$ could be audio-only, visual-only or audio-visual stream.

- *Multi-stream HMM:* $\rho(i,t)$ is calculated as follows,

$$\rho(i,t) = P(\mathbf{o}^a_{1:t}, \mathbf{o}^v_{1:t}, q_t = i) \quad (7)$$

$$= P(\mathbf{o}^a_{1:t}, q_t = i)^{\omega} P(\mathbf{o}^v_{1:t}, q_t = i)^{1-\omega}, \quad (8)$$

  where $\mathbf{o}^a_{1:t}$ is the audio-only sequence and $\mathbf{o}^v_{1:t}$ is the visual-only sequence. ω is the weighting factor defined in equation 4.

- *Asynchronous HMM:* $\rho(i,t)$ is calculated as follows,

$$\rho(i,t) = \sum_{s=t-\triangle t}^{t+\triangle t} P(\mathbf{o}^a_{1:t}, \mathbf{o}^v_{1:s}, q_t = i, \tau_t = s), \quad (9)$$

  where the variable $\tau_t = s$ can be seen as the alignment between sequence $\mathbf{o}^a_{1:t}$ (audio-only stream) and $\mathbf{o}^v_{1:s}$ (visual-only stream).

In the second step, we normalize the probability $\rho(i,t)$ for all states of all the models. The probabilities of all states for all models sum up to one,

$$\sum_{j=1}^{N_S} P(q_t = j) = 1, \quad (10)$$

where N_S is the number of all states for all models. Then the probability $P(q_t = i|\mathbf{o}_{1:t})$ of state i given a sequence $\mathbf{o}_{1:t}$ is

$$P(q_t = i|\mathbf{o}_{1:t}) = \frac{P(q_t = i, \mathbf{o}_{1:t})}{P(\mathbf{o}_{1:t})} \quad (11)$$

$$= \frac{P(q_t = i, \mathbf{o}_{1:t})}{\sum_{j=1}^{N_S} P(q_t = j, \mathbf{o}_{1:t})} \quad (12)$$

$$= \frac{\rho(i,t)}{\sum_{j=1}^{N_S} \rho(j,t)}. \quad (13)$$

With this, the probability P^t_k of model M_k given a sequence $\mathbf{o}_{1:t}$ is then computed as

$$P^t_k = \sum_{i \in M_k} P(q_t = i|\mathbf{o}_{1:t}) \quad (14)$$

$$= \sum_{i \in M_k} \frac{\rho(i,t)}{\sum_{j=1}^{N_S} \rho(j,t)}, \quad (15)$$

where i is the state in model M_k, which is a subset of the states of all models, and N_S is the total number of states. The probability P^t_k of model M_k is the sum of the probabilities of all states in model M_k.

In the second approach, the individual action model with the highest probability outputs a value of 1, while all other models output a zero value. The vector a^t generated in this way is used as input to *G-HMM*. We refer to it as *hard decision*.

We concatenate the individual recognition vectors from all participants, together with the group-level features, into a ($N_I \times N_P + N_{GF}$)-dimensional vector (where N_P is the number of participants, and N_{GF} is the dimension of the group features) as observations to *G-HMM* for group action recognition.

4 Meeting Database

We used the public meeting corpus we first described in [16], which was collected in a meeting room equipped with synchronized multi-channel audio and video recorders[1]. The sensors include three fixed cameras and

[1] http://mmm.idiap.ch/

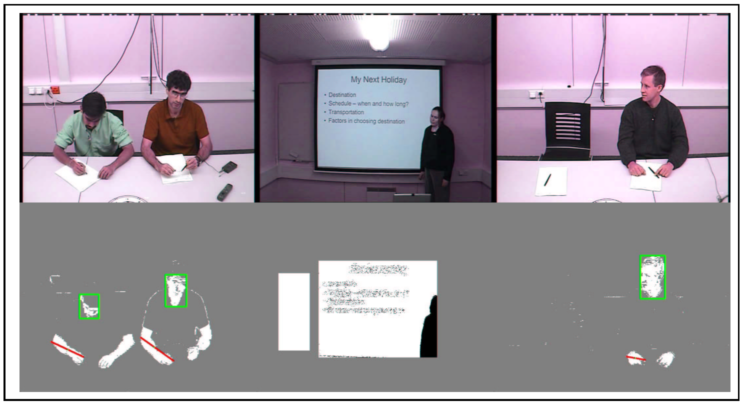
My Next Holiday
• Destination
• Schedule – when and how long?
• Transportation
• Factors in choosing destination

Figure 2: Multi-camera meeting room and visual feature extraction

twelve microphones [20]. Two cameras have an upper-body, frontal view of two participants including part of the table. A third wide-view camera captures the projector screen and white-board. The multi-camera meeting room and visual feature extraction is shown in Figure 2. Audio was recorded using lapel microphones for all participants, and an eight-microphone array place in the center of the table. The corpus consists of 59 short meetings at five-minute average duration, with four participants per meeting. The group action structure was scripted before recording, so part of the group actions labels we define were already available as part of the public corpus. However, we needed to relabel the rest of the group actions (e.g. *monologues* into either *monologues* or *monologues+note-taking*), and to label the entire corpus in terms of individual actions. All ground-truth was produced using *Anvil*, a publicly available video annotation tool [2].

5 Multi-Modal Feature Extraction

In this section, we describe the process to extract the two types of AV features used in this work: person-specific AV features and group-level AV features. The former are extracted from individual participants. The latter are extracted from the whiteboard and projector screen regions.

5.1 Person-Specific AV Features

Person-specific visual features were extracted from the cameras that have a close view of the participants. Person-specific audio features were extracted from the lapel microphones attached to each person, and from the microphone array. The complete set of features is listed in Table 4.

Person-specific visual features. For each video frame, the raw image is converted to a skin-color likelihood image, using a 5-component skin-color Gaussian mixture model (GMM). We use the chromatic color space, known to be less variant to the skin color of different people [34]. The chromatic colors are defined by a normalization process: $r = \frac{R}{R+G+B}, g = \frac{G}{R+G+B}$. Skin pixels were then classified based on thresholding of the skin likelihood. A morphological postprocessing step was performed to remove noise. The skin-color likelihood image is the input to a connected-component algorithm (flood filling) that extracts blobs. All blobs whose areas are smaller than a given threshold were removed. We use 2-D blob features to represent each participant in the meeting, assuming that the extracted blobs correspond to human faces and hands. First, we use a multi-view face detector to verify blobs corresponding to the face. The blob with the highest confidence output by the face detector is recognized as the face. Among the remaining blobs, the one that has the rightmost centroid horizontal position is identified as the right hand (we only extracted features from the right hands since the participants in the corpus are predominately right-handed). For each person, the detected face blob is

[2] http://www.dfki.de/ kipp/anvil/

Table 4: Audio-visual feature list

		Description
Person-Specific Features	Audio	SRP-PHAT from each seat
		speech relative pitch
		speech energy
		speech rate
	Visual	head vertical centroid
		head eccentricity
		right hand horizontal centroid
		right hand angle
		right hand eccentricity
		head and hand motion
Group Features	Audio	SRP-PHAT from white-board
		SRP-PHAT from projector screen
	Visual	mean difference from white-board
		mean difference from projector screen

represented by its vertical centroid position and eccentricity [29]. The hand blob is represented by its horizontal centroid position, eccentricity, and angle. Additionally, the motion magnitude for head and right hand are also extracted and summed into one single feature.

Person-specific audio features. Using the microphone array and the lapels, we extracted two types of person-specific audio features. On one hand, speech activity was estimated at four seated locations, from the microphone array waveforms. The seated locations were fixed 3-D vectors measured on-site. The speech activity measure was SRP-PHAT [5], which is a continuous, bounded value that indicates the activity at a particular location. On the other hand, three acoustic features were estimated from each lapel waveform: energy, pitch and speaking rate. We computed these features on speech segments, setting a value of zero on silence segments. Speech segments were detected using the microphone array, because it is well suited for multiparty speech. We used the SIFT algorithm [15] to extract pitch, and a combination of estimators [22] to extract speaking rate.

5.2 Group AV Features

Group AV features were extracted from the white-board and projector screen regions, and are listed in Table 4.

Group visual features. These were extracted from the camera that looks towards the white-board and projector screen area. We first get difference images between a reference background image and the image at each time, in the white-board and projector screen regions (see Figure 2). On these difference images, we use the average intensity over a grid of 16×16 blocks as features.

Group audio features. These are SRP-PHAT features extracted using the microphone array from two locations corresponding to the white-board and projector screen.

6 Experiments

In this section, we first describe the measures used to evaluate our results, and then present results for both individual action recognition and group action recognition.

6.1 Performance Measures

We use the *action error rate* (*AER*) and the *frame error rate* (*FER*) as measures to evaluate the results of group action recognition and individual action recognition, respectively.

AER is equivalent to the word error rate widely used in speech recognition, and is defined as the sum of insertion (Ins), deletion (Del), and substitution (Subs) errors, divided by the total number of actions in the

Table 5: Individual actions in different sets

Individual Actions	**train**		**test**	
	N_F	N_A	N_F	N_A
speaking	35028	1088	33747	897
writing	15803	363	27365	390
idle	127569	1426	112488	1349
total	178400	2877	173600	2636

ground-truth, $AER = \frac{Subs+Del+Ins}{total\ actions} \times 100\%$. For group action recognition, we have $N_G = 8$ possible actions which in many cases have no clear-cut temporal boundaries. Furthermore, at least five actions occur in each meeting in the corpus. We believe that *AER* is a thus good measure to evaluate group action recognition, as we are more interested in the recognition of the correct action sequence rather than the precise time alignment of the recognized action segments.

However, *AER* overlooks the time alignment between recognized and target action segments. For individual action recognition, there are only $N_I = 3$ possible actions. Furthermore, some streams (participants) in the corpus consist of only two individual actions (e.g., a person who talks only once during the course of a meeting). *AER* might not provide a meaningful assessment in such cases. In this view, we adopt *FER* as the performance measure for individual action recognition. *FER* is defined as one minus the ratio between the number of correctly recognized frames and the number of total frames, $FER = (1 - \frac{correct frames}{total frames}) \times 100\%$. This measure reflects well the accuracy of the boundaries (begin and end time) of the recognized actions, compared to manually labeled action boundaries.

With limited number of training and testing actions, results are likely to vary due to the random initialization of the training procedure (based on Expectation-Maximization [27]). For this reason, and to assess consistency in the results, we report the mean and standard deviation (*STD*) for *AER* and *FER*, computed over 10 runs.

Finally, we also use confusion matrices, whose rows and columns index the recognized and ground-truth actions, respectively; The element c_{ij} of the confusion matrix corresponds to either the percentage (for individual actions) or the instances (for group actions) of action *j* recognized as action *i*. The confusion matrix for group actions is based on *AER*, so there are substitution, insertion, and deletion errors. For individual actions, there are neither insertions nor deletions because the peformance measure is *FER*.

6.2 Experiments Protocol

For both individual and group action recognition, we use 6-fold cross-validation on the training set to select the best parameters. For cross-validation, we split the training set into a training set and a validation set. For individual action recognition, we split the training streams according to person identities. There are no people overlapping across training, validation, and test sets. For group actions, there is no overlap in participants between training and test sets. We intend the models to be person-independent, so they can be used for future meetings without need for re-training. After the best model parameters are selected, we train models on the whole training set and apply them on the test set.

From the 59 meetings, 30 are used as training data, and the remaining 29 are used for testing. The number of frames (N_F) and number of actions (N_A) for individual action and group action in the different data sets are summarized in Table 5-6. The number of individual actions is much larger than that of group actions. There are two reasons. First, for individual action recognition, there are four participants for each meeting. Therefore, there are $30 \times 4 = 120$ streams for training and $29 \times 4 = 116$ streams for testing. Second, the duration of individual actions is typically shorter than that of group actions.

6.3 Individual Action Recognition

The three methods described in Section 3.3.1 were tested for individual action recognition.

Early integration (Early Int.). trained on three feature sets: audio-only, visual-only. and audio-visual.

Table 6: Group actions in different sets

Group Actions	train		test	
	N_F	N_A	N_F	N_A
discussion	17760	48	14450	49
monologue	7615	26	7585	26
monologue + note-taking	6260	17	6695	23
note-taking	640	6	320	3
presentation	3170	6	3345	9
presentation + note-taking	3455	5	3865	9
white-board	2155	5	265	1
white-board + note-taking	3545	11	6875	19
total	44600	124	43400	139

Table 7: Results of individual action recognition

Method	Features	FER (%)	STD
Early Int.	Visual	34.17	3.64
	Audio	23.48	2.70
	Audio-visual	9.98	2.65
MS-HMM	Audio-visual	8.58	1.76
A-HMM	Audio-visual	7.42	1.13

Audio-visual multi-stream HMM (MS-HMM). combining individual audio and visual streams.

Audio-visual asynchronous HMM (A-HMM), combining individual audio and visual streams.

Multi-stream HMMs allow us to give different weights to different modalities. Following the discussion presented in [17], we use (0.8,0.2) to weight the audio and visual modalities, respectively. For asynchronous HMM, the allowed asynchrony ranges from $\pm 2.2s$. Results are presented in Table 7 in terms of *FER* mean and standard deviation, obtained over 10 runs.

From Table 7, we observe that all methods using AV features produced less than 10% *FER*, which is about 15% absolute improvement over using audio-only features, and about 25% absolute improvement over using visual-only features. Asynchronous HMM produced the best result. Given that the total number of frames is over $43,000$, the improvement using asynchronous HMM over the other HMM methods is statistically significant with a confidence level above 99%, using a standard proportion test [9]. The improvement suggests that there exist asynchronous effects between the audio and visual modalities. Additionally, we tested the MS-HMM system with equal-weight scheme. The performance decreased compared to the MS-HMM with larger weight on audio. This is not surprising given the predominant role of audio in the defined actions.

The confusion matrices for visual-only, audio-only, and audio-visual streams, corresponding to a randomly chosen single run, are shown in Tables 8, 9, and 10, respectively. We can see that *"speaking"* is well detected using audio-only features, and that *"writing"* is well detected using visual-only features. Using audio-visual features, both *"speaking"* and *"writing"* are generally well detected. Using AV features, *"writing"* tends to get confused with *"idle"*, which in turn is the action with the highest *FER*. This is likely due to the catch-all role that this action plays. In practice, *"idle"* includes all other possible AV patterns, (e.g. pointing, laughing, etc.), which makes its modeling more difficult, compared with the other two well-defined actions.

In order to empirically investigate asynchronous effects in the individual actions, we performed forced alignment decoding on the audio-only and visual-only streams independently. A similar approach was taken to establish empirical evidence for asynchrony in multi-band automatic speech recognition in [19]. The decoder in each stream was constrained by the ground-truth individual action sequence, and so the output action sequences differ only in their temporal boundaries. We calculated the time misalignment (start-time difference of corresponding actions) between the two sequences. Actions having absolute misalignments larger than $5s$ were discarded, as the misalignments were more likely caused by recognition errors, rather than asynchronous

Table 8: Confusion matrix of recognized individual actions (using visual-only features)

	Speaking	Writing	Idle
Speaking	51.92%	3.00%	8.22%
Writing	45.87%	85.93%	34.65%
Idle	2.21%	11.07%	57.13%

Table 9: Confusion matrix of recognized individual actions (using audio-only features)

	Speaking	Writing	Idle
Speaking	91.74%	1.26%	1.78%
Writing	1.16%	35.23%	22.10%
Idle	7.10%	63.51%	76.12%

effects. Figure 6.3 shows the resulting histogram of misalignments, assumed due to asynchronous effects, for these individual actions. The histogram can be approximated by a Gaussian distribution, with a mean of $-0.13s$ (effectively zero, as misalignments happened in both directions) and a standard deviation of 2.05. More than 80% of the individual actions are distributed in the range of $\pm 2.2s$, while there are 17% individual actions without any asynchronous effects ($P(t = 0) = 17\%$). This suggests that, for most individual actions having evidence in both streams, allowing asynchrony between streams should more accurately model the data.

6.4 Group Action Recognition

Using the outputs from *I-HMM* and the group-level features, concatenated as described in Section 3.3.2, we investigated the following cases for recognition of group actions:

- *Early integration, visual-only, soft decision*. A normal HMM is trained using the combination of the results of the *I-HMM* trained on visual-only features, and the visual group features. The soft decision criteria is used.

- *Early integration, audio-only, soft decision*. Same as above, but replacing visual-only by audio-only information.

- *Early integration, AV, hard decision*. Same as above, but replacing visual-only by audio-visual information. The hard decision criteria is used.

- *Early integration, AV, soft decision*. Same as above, but changing the criteria to link two HMM layers.

- *Multi-stream, AV, hard decision*, using the multi-stream HMM approach as *I-HMM*. The hard decision criteria is used.

- *Multi-stream, AV, soft decision*. Same as above, but changing the criteria to link two HMM layers.

- *Asynchronous HMM, AV, hard decision*. We use the asynchronous HMM for individual action layer and audio-visual features. The hard decision criteria is used.

Table 10: Confusion matrix of recognized individual actions (using AV features)

	Speaking	Writing	Idle
Speaking	94.23%	2.12%	4.73%
Writing	1.03%	89.60%	10.89%
Idle	4.74%	8.28%	84.38%

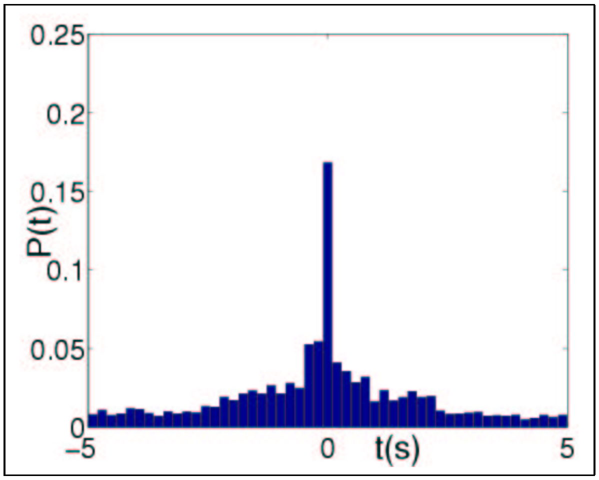

Figure 3: Histogram of asynchronous effects of individual actions

Table 11: Results of group action recognition

Method			AER (%)	STD
Single-layer HMM	Visual		48.20	3.78
	Audio		36.70	4.12
	Audio-visual		23.74	2.97
Two-layer HMM	Visual		42.45	2.85
	Audio		32.37	2.10
	Early Int.	hard	17.98	2.75
		soft	16.55	1.40
	MS-HMM	hard	17.27	2.01
		soft	15.83	1.61
	A-HMM	hard	17.85	2.87
		soft	15.11	1.48

- *Asynchronous HMM, AV, soft decision*. Same as above, but changing the criteria to link two HMM layers.

As baseline methods for comparison, we tested single-layer HMMs, using low-level audio-only, visual-only, and AV features as observations [16], and trained by cross-validation following the same experimental protocol. The results appear in Table 11, in terms of *AER* mean and standard deviation over 10 runs.

We observe from Table 11 that the use of AV features outperformed the use of single modalities for both single-layer HMM and two-layer HMM methods. This result supports the hypothesis that the group actions we defined are inherently multimodal. Furthermore, the best two-layer HMM method (A-HMM) using AV features improved the performance by over 8% compared to the AV single-layer HMM. Given the small number of group actions in the corpus, a standard proportion test indicates that the difference in performance between AV single-layer and the best two-layer HMM is significant at the *96%* confidence level. Additionally, the standard deviation for the two-layer approach is half the baseline's, which suggests that our approach might be more robust to variations in initialization, given the fact that each HMM stage in our approach is trained using an observation space of relatively low dimension. Regarding hard vs. soft decision, soft decision produced a slightly better result, although not statistically significant given the number of group actions. However, the standard deviation using soft-decision is again around half the corresponding to hard-decision. Overall, the soft decision two-layer HMM appears to be favored by the results.

To further analyze results, we provide the confusion matrices for single-layer HMM using AV features, and two-layer HMM using AV, soft-decision and asynchronous HMM in Tables 12 and 13, respectively. We showed

Table 12: Confusion matrix of recognized group actions for single-layer HMM using audio-visual features

	D	M1	M1+N	M2	M2+N	M3	M3+N	M4	M4+N	N	P	P+N	W	W+N	DEL
D	45														1
M1	2	6	3												
M1+N			3												
M2				6	1										
M2+N				2	3										1
M3						2									1
M3+N						3	7								
M4								2							
M4+N								3	5						
N										2					1
P											6	5			1
P+N											1	3			
W	1					1					1		1	2	
W+N												1		17	
INS					1		1								

discussion (D), monologue (M1· · · M4), monologue+note-taking (M1+N· · · M4+N), note-taking (N), presentation (P), presentation+note-taking (P+N), white-board (W), and white-board+note-taking (W+N). Empty cells represent zero values. It is evident that the two-layer method greatly reduced the number of errors, compared with the single-layer method. For both matrices, we see that most substitution errors come from confusions between actions with and without note-taking. This might be mainly because several instances of *"writing"* could not be reliably detected as individual actions, as mentioned in the previous subsection. There are several *"presentation"* actions confused with *"white-board"*, which might be because some speakers moved around the white-board and projector-screen regions during a presentation. On the other hand, *"discussion"* and *"note-taking"* actions can be recognized reasonably well.

6.5 Recognizing Actions in Unconstrained Meetings

To facilitate training and evaluation, the previoous experiments were conducted on scripted meetings recorded in constrained conditions. To assess the proposed framework on natural multi-party conversations, we use a one-hour publicly available natural meeting recorded in the same setup, with which the AV single-layer HMM was compared to the best two-layer method, i.e., AV asynchronous HMM with soft-decision. All parameters used for both methods are the same as in previous experiments.

The two methods were evaluated independently by two observers. The subjects watched and listened to the meeting recording, and judged the correctness of the actions automatically recognized using the single-layer and the two-layer methods. A final decision was made by the third person, for those actions in disagreement among each pair of observers. The results are shown in Table 14 (N_A denotes the number of recognized actions for each system).

We can see that the results obtained with the two-layer HMM approach are better than those of the single-layer HMM, which again suggests the benefits of the proposed framework. For the one-hour natural meeting, over 70% group actions were correctly recognized using the layered method, which could be quite useful to meeting browsing and indexing. In practice, we noticed that it is difficult to determine clear-cut differences between the monologue and discussion actions, which constituted the main source of disagreement between the subjects that evaluate the results. Therefore, in future work, we need to address the ill-defined nature of some actions in real data.

Table 13: Confusion matrix of recognized group actions for two-layer HMM (using asynchronous HMM with soft decision)

	D	M1	M1+N	M2	M2+N	M3	M3+N	M4	M4+N	N	P	P+N	W	W+N	DEL
D	44														1
M1	2	6	2												
M1+N			4												
M2				7											
M2+N				1	5										
M3						5									1
M3+N						1	6								
M4								4							
M4+N								1	5						
N							1			3					
P											6				
P+N												8			
W	2										2	1	1	1	
W+N											1			18	
INS				1								1	1		

Table 14: Results on unconstrained meetings

Method	N_A	correct rate (%)
Single-layer HMM	40	57.5
Two-layer HMM	37	70.3

7 Conclusions and Future Work

In this paper, meetings were defined as sequences of multi-modal group actions. We addressed the problem of modeling and recognizing such group actions, proposing a two-layer HMM framework to decompose the group action recognition problem into two layers. The first layer maps low-level AV features into individual actions. The second layer uses results from the first layer as input to recognize group actions. Experiments on a public 59-meeting corpus demonstrate the effectiveness of the proposed framework to recognize a set of eight multimodal turn-taking actions, compared to a baseline, single-layer HMM system. We believe our methodology to be promising. In the short term, we will explore its applicability to other sets of group actions.

Acknowledgments

This work was partly supported by the EU 5th FWP IST project M4 (IST-2001-34485), the EU 6th FWP IST Integrated Project AMI (FP6-506811, pub. AMI-11), the EU 6th FWP IST PASCAL Network of Excellence (IST-2002-506778), all three funded by the Swiss OFES, and by the Swiss NCCR IM2. We also thank Stephan Reiter (TUM) for providing part of the ground-truth for individual actions.

References

[1] S. Basu, T. Choudhury, B. Clarkson, and A. Pentland. Towards measuring human interactions in conversational settings. In *Proc. IEEE CVPR Workshop on Cues in Communication*, Kawai, Dec. 2001.

[2] S. Bengio. An asynchronous hidden Markov model for audio-visual speech recognition. In S. Becker, S. Thrun, and K. Obermayer, editors, *Proc. NIPS 15*, 2003.

[3] P. Chiu, J. Boreczky, A. Girgensohn, and D. Kimber. LiteMinutes: An Internet-based system for multimedia meeting minutes. In *Proc. Tenth World Wide Web Conference*, 140-149, 2001.

[4] R. Cutler, Y. Rui, A. Gupta, J. Cadiz, I. Tashev, L. He, A. Colburn, Z. Zhang, Z. Liu, and S. Silverberg. Distributed meetings: A meeting capture and broadcasting system. In *Proc. ACM Multimedia,* 2002.

[5] J. DiBiase, H. Silverman, and M. Brandstein. Robust localization in reverberant rooms. In M. Brandstein and D. Ward, editors, *Microphone Arrays*, chapter 8, pages 157–180. Springer, 2001.

[6] A. Dielmann and S. Renals. Dynamic Bayesian networks for meeting structuring. In *Proc. IEEE ICASSP*, 2004.

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

[8] A. Galata, N. Johnson, and D. Hogg. Learning behavior models of human activities. *In British Machine Vision Conference,* 1999.

[9] D. Gibbon, R. Moore, and R. Winksi. Handbook of Standards and Resources for Spoken Language Systems. *Mouton de Gruyter*, 1997.

[10] D. Hillard, M. Ostendorf, and E. Shriberg. Detection of agreement vs. disagreement in meetings: Training with unlabeled data. In *Proc. HLT-NAACL Conference,* Edmonton, May 2003.

[11] S. Hongeng and R. Nevatia. Multi-agent event recognition. In *Proc. IEEE ICCV,* Vancouver, July 2001.

[12] D.M. Gavrila. The Visual Analysis of Human Movement: A Survey. *Computer Vision and Image Understanding: CVIU*, 73:82–98, 1999.

[13] R. Krauss and C. Garlock and P. Bricker and L. McMahon. The role of audible and visible back-channel responses in interpersonal communication. *Journal of Personality and Social Psychology*, 35(7):523-529, 1977.

[14] F. Kubala. Rough'n'Ready: a meeting recorder and browser. In *ACM Computing Surveys*, 31, 1999.

[15] J. D. Markel. The SIFT algorithm for fundamental frequency estimation. *IEEE Transactions on Audio and Electroacoustics*, 20:367–377, 1972.

[16] I. McCowan, S. Bengio, D. Gatica-Perez, G. Lathoud, F. Monay, D. Moore, P. Wellner, and H. Bourlard. Modeling human interactions in meetings. In *Proc. IEEE ICASSP*, Hong Kong, April 2003.

[17] I. McCowan, D. Gatica-Perez, S. Bengio, G. Lathoud, M. Barnard, and D. Zhang Automatic analysis of multimodal group actions in meetings. IDIAP-RR 27, IDIAP, Martigny, Switzerland, May 2003.

[18] J. E. McGrath. *Groups: Interaction and Performance*. Prentice-Hall, 1984.

[19] N. Mirghafori and N. Morgan *Transmissions and Transitions: A Study of Two Common Assumptions in Multi-Band ASR.* In *Proc. ICASSP*, Seattle, 1998.

[20] D. Moore. The IDIAP smart meeting room. IDIAP-COM 07, IDIAP, 2002.

[21] N. Morgan, D. Baron, J. Edwards, D. Ellis, D. Gelbart, A. Janin, T. Pfau, E. Shriberg, and A. Stolcke. The meeting project at ICSI. In *Proc. HLT Conference*, San Diego, CA, March 2001.

[22] N. Morgan and E. Fosler-Lussier. Combining multiple estimators of speaking rate. *in Proc. ICASSP*, 1998.

[23] K. Murphy. Dynamic Bayesian networks: Representation, inference and learning. *Ph.D. dissertation, UC Berkeley*, 2002.

[24] N. Oliver, E. Horvitz, and A. Garg. Layered representations for learning and inferring office activity from multiple sensory channels. In *Proc. ICMI*, October 2002.

[25] N. Oliver, B. Rosario, and A. Pentland. A Bayesian computer vision system for modeling human interactions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8), August 2000.

[26] E. Padilha and J. Carletta. Nonverbal behaviours improve a simulation of small group discussion. In *Proc. First Int. Nordic Symposium of Multi-modal Communication*, Copenhagen, Sep. 2003.

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

[28] R. Stiefelhagen Tracking focus of attention in meetings. *IEEE International Conference on Multimodal Interfaces*, Pittsburgh, PA, 2002.

[29] T. Starner and A. Pentland. Visual recognition of american sign language using HMMs. In *Proc. Int. Work. on AFGR*, Zurich, 1995.

[30] A. Waibel, M. Bett, F. Metze, K. Ries, T. Schaaf, T. Schultz, H. Soltau, H. Yu, and K. Zechner. Advances in automatic meeting record creation and access. *in Proc. IEEE ICASSP*, May 1999.

[31] B. Wrede and E. Shriberg. The relationship between dialogue acts and hot spots in meetings. In *Proc. ASRU*, Virgin Islands, Dec. 2003.

[32] C.R. Wren, A. Azarbayejani, T. Darrell and A. Pentland. Pfinder: Real-Time Tracking of the Human Body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):780-785, 1997.

[33] L. Xie, S.-F. Chang, A. Divakaran, and H. Sun. Unsupervised discovery of multilevel statistical video structures using hierarchical hidden Markov models. *in Proc. ICME*, July 2003.

[34] J. Yang, L. Weier, and A. Waibel. Skin-color modeling and adaptation. *in Proc. ACCV*, 1998.