IDIAP RESEARCH REPORT

MULTIMODAL GROUP ACTION CLUSTERING IN MEETINGS

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

IDIAP–RR 04-24

DEC. 2004

SUBMITTED FOR PUBLICATION

Dalle Molle Institute for Perceptual Artificial Intelligence • P.O.Box 592 • Martigny • Valais • Switzerland

phone +41 – 27 – 721 77 11
fax +41 – 27 – 721 77 12
e-mail secretariat@idiap.ch
internet http://www.idiap.ch

1 IDIAP, Martigny, Switzerland

IDIAP Research Report 04-24

MULTIMODAL GROUP ACTION CLUSTERING IN MEETINGS

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

DEC. 2004

SUBMITTED FOR PUBLICATION

Abstract

We address the problem of clustering multimodal group actions in meetings using a two-layer HMM framework. Meetings are structured as sequences of group actions. Our approach aims at creating one cluster for each group action, where the number of group actions and the action boundaries are unknown a priori. In our framework, the first layer models typical actions of individuals in meetings using supervised HMM learning and low-level audio-visual features. A number of options that explicitly model certain aspects of the data (e.g., asynchrony) were considered. The second layer models the group actions using unsupervised HMM learning. The two layers are linked by a set of probability-based features produced by the individual action layer as input to the group action layer. The methodology was assessed on a set of multimodal turn-taking group actions, using a public five-hour meeting corpus. The results show that the use of multiple modalities and the layered framework are advantageous, compared to various baseline methods.

1 Introduction

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 [23, 16, 5, 13, 9, 24, 7]. Analyzing meetings poses a diversity of technical challenges, and opens doors to a number of relevant applications. On one hand, meetings constitute an important case study of human interaction. Understanding people interaction has been a long-term goal in social psychology [15], so a computational framework to analyze group behavior could be useful to facilitate analysis performed by psychologists in organizations (e.g., for training of staff on issues like interpersonal communication and teamwork management). On the other hand, meetings can be seen as raw, unlabeled data, possibly generated in large amounts, for which automatic analysis could add value for browsing and retrieval purposes, e.g., to structure a single meeting into a sequence of high-level items, or to discover recurrent patterns in a large meeting collection.

Meetings are characterized by their multimodal and group nature [11, 15]. Regarding the first factor, single modalities [23, 16, 9, 24] have been used for various tasks, but there are few works that model individual and group behavior in conversational settings using multiple modalities (as captured by cameras and microphones) [3, 13, 14], despite the experimental evidence supporting this approach. For the second factor, a meeting can be seen as proceeding through diverse phases, where a group disseminates information, discusses, and makes decisions [15]. A simple model can thus be used to define a meeting as a continuous sequence of group actions (i.e., involving multiple simultaneous participants) chosen from one or more pre-defined action dictionaries, which is well suited for supervised learning [13, 7], as long as the action dictionaries are well defined. This implies that the actions comprising each dictionary should be mutually exclusive, exhaustive, and unambiguous to human observers, at least to a degree for which manually labeled data for supervised learning can be reliably generated.

In reality however, meetings are not restricted to pre-defined action sets. Furthermore, high-level group actions in meetings can be ambiguous (and expensive) to label. Roughly speaking, the degree of ambiguity correlates with the actions' level of semantic meaning. Basic actions like writing or speaking can be clearly identified, group actions like discussions are more ambiguous, and high-level actions like information sharing might be very difficult to label reliably, which could seriously challenge supervised methods.

In this view, modeling high-level group actions with unsupervised approaches, which find "action structure" in either individual meetings or whole collections, without the need for labeled data or previous knowledge of the actions, become very attractive options [26, 27], especially given the vast amount of data that is generated in many real cases. Given adequate features, clustering an individual meeting could partition it into action-consistent segments. Clustering an entire collection could further find action-consistent clusters across meetings. Additionally, unsupervised methods could naturally deal with variations (e.g. in the number of participants) that would otherwise need to be modeled explicitly in supervised methods.

In this paper, we present a layered probabilistic framework for group action clustering in meetings, as an alternative to fully supervised methodologies. Through the definition of an adequate set of individual actions, we decompose the group action clustering problem into two layers. The first one performs supervised learning to recognize individual actions of participants using low-level audio-visual (AV) features. Supervision at this level can be especially convenient because individual actions are often well-defined and thus can be reliably labeled. Individual actions constitute the link between low-level AV features and high-level group actions. The second layer models group actions in an unsupervised way, using the output of the first layer as observations, and producing a temporal segmentation of a meeting into group action segments. Both layers use HMM-based approaches for action recognition and clustering, respectively. Our framework is extensible: with minor modifications, it can be used to cluster group actions in either individual meetings or in an entire meeting collection. We apply the methodology to a publicly available meeting corpus, for a set of eight group actions based on multimodal turn-taking patterns, and illustrate its validity with respect to a number of baseline methods. In our view, our methodology constitutes an attractive option for analysis of high-level group actions in meetings, due to its potential to deal with actions that would otherwise be difficult to pre-define and/or expensive to label.

The rest of the paper is organized as follows. Section 2 reviews related work. Section 3 introduces our approach. Section 4 presents experiments and discussion. Concluding remarks are provided in Section 5.

2 Related Work

Learning-based approaches for the automatic interpretation of human activities in videos have been used for the past ten years. Most works have focused on supervised learning methods, defining models for a handful of activities in a particular domain, and using statistical models for recognition. Individual action [22], and interaction recognition [20, 10] have been predominately investigated using visual features, although some work on the speech community can be categorized as interaction recognition [9, 24]. To our knowledge however, little work has been conducted on recognition of group-based, multimodal actions from multiple audio-visual streams captured by cameras and microphones [3, 13, 14]. [3] described automatic discovery of "influence" in a lounge room where people played interactive debating games. [13, 14] are the closest works to ours, which studied various 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.

Most of the existing work has used Hidden Markov Models (HMMs) and extensions (see [18] for a recent review of models). The basic HMM works well for temporally correlated sequential data, but it is challenged by a large number of parameters, and the risk of over-fitting when learned from limited data [19]. This situation might occur in the case of multimodal group actions where, large vectors of audio-visual features from all participants are concatenated to define the observation space [13, 14].

The above problem has been recently addressed with hierarchical representations [19, 7, 25]. In [19], (supervised) layered HMMs were proposed to model multimodal single-person office activities at various time granularities. The lowest layer captured video, audio, keyboard and mouse activity features; the middle layer classifies AV features into basic events; the highest layer uses outputs of previous layers to recognize higher-level office activities. In [7], two methods for meeting structuring from audio-only were presented, using multilevel Dynamic Bayesian Networks (DBNs). In [25], 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. However, in both [25, 7], 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.

Different from supervised methods for activity recognition, unsupervised data-driven approaches find action-based clusters from the data, without a priori knowledge of the action dictionaries [26, 27]. In [26], a normalized-cut approach was used to cluster single-person actions like running, walking, etc., using features at different temporal scales, and a distribution-based distance measure to compute similarity between video segments. One limitation of such approach was the lack of a sound mechanism to detect the number of clusters. Recently, an unsupervised technique was proposed to detect unusual human activity in a surveillance setting, using analysis of co-occurrence between video clips and motion/color features of moving objects, without the need to build models for usual activities [27]. The two approaches relied only on visual information.

Unlike previous work, our work combines supervised HMM recognition and unsupervised HMM cluster-

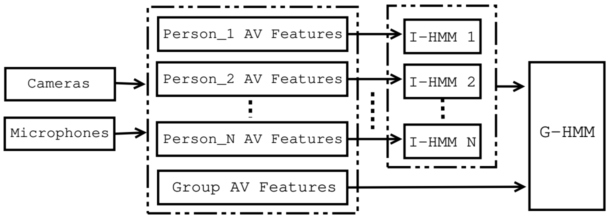

Figure 1: Framework overview

ing in a stratified framework, to model multimodal group actions in meetings. The layered structure in our approach, that explicitly considers different semantic levels (individual and group) coincides with the structure of meetings as modeled in social psychology [15]. The distinct treatment for each layer (supervised vs. unsupervised) tries to respond to the different nature of each of the action types.

3 Group Action Clustering

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

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 to identify and group together all meeting segments of the same group action, 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 clustering layer (group action). *I-HMM* receives as input audio-visual (AV) features extracted from each participant, and outputs the probability for each individual action model (see section 3.2). In this layer, a number of HMM variants that might capture better the characteristics of the data (e.g. asynchrony [4], or different noise conditions [8] between the audio and visual streams) can be used. For the second layer, *G-HMM* uses as input both the output from *I-HMM*, and a set of *group features*, directly extracted from the raw streams, which are not associated to any particular participant (see section 4.2). Our approach can be summarized into three stages (Fig.1):

1. **Feature Extraction:** Extract individual-level and group-level audio-visual features.

2. **Supervised Individual Action Recognition:** Given individual features for each person, train *I-HMM* and output probabilities for individual action models.

3. **Unsupervised Group Action Clustering:** Apply *G-HMM* clustering using features constructed by concatenating individual action features and group-level features.

Compared with a single-layer HMM, which directly uses audio-visual features for group action clustering, our approach has the following advantages: (1) a single-layer HMM is defined on a possibly large observation space, which might face the problem of over-fitting with limited amount of training data. In contrast, the layers in our approach are defined on small-dimensional observation spaces, which might result in more stable performance in cases of limited data. (2) The *I-HMMs* are person-independent, and in practice can be trained with sufficient data, as each meeting in the training set provides multiple individual streams. Better generalization performance can then be expected. (3) The *G-HMM* is less sensitive to variations 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 handled independently, and so different HMM combination systems can be studied. The framework is therefore simple to interpret, and can be improved at each level. In particular, in this paper we explore models for the lower layer that could be particularly suitable for multimodal asynchronous data sequences.

3.2 Supervised I-HMM

The *I-HMM* layer is learned in a supervised fashion. We investigate three models for the lower-layer, each of which attempts to model specific aspects of the data (please refer to the original references for details):

1. *Early Integration* (*Early Int.*), where a basic HMM [21] 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.

2. *Audio-Visual Multi-Stream* (*MS-HMM*), which combines the audio-only and visual-only streams. Each stream is modeled independently. The final classification is based on the the fusion of the outputs of both modalities by estimating their joint occurrence [8].

3. *Audio-Visual Asynchronous* (*A-HMM*), which also combines audio and visual 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 [4].

As features for the group action clustering algorithm, the lower layer outputs the probability p_k^t for each individual action model $M_k, k = 1, ..., N_I$, given a sequence $x_1^t = x_1, ..., x_t$, where N_I denotes the number of individual actions. Let $\alpha(i,t) \stackrel{\text{def}}{=} P(x_1^t, q_t = i)$ denote the forward variable, which is the probability of having generated the sequence x_1^t and being in the state i at time t in the standard Baum-Welch algorithm [21]. Given that the probabilities of all states sum up to one, $\sum_{j=1}^{N_S} P(q_t = j) = 1$, where N_S is the number of all states for all models, the probability $P(q_t = i|x_1^t)$ of state i given a sequence x_1^t is:

$$P(q_t = i|x_1^t) = \frac{P(q_t = i, x_1^t)}{P(x_1^t)} = \frac{P(q_t = i, x_1^t)}{\sum_{j=1}^{N_S} P(q_t = j, x_1^t)} \tag{1}$$

$$= \frac{\alpha(i,t)}{\sum_{j=1}^{N_S} \alpha(j,t)}. \tag{2}$$

The probability p_k^t of model M_k is then computed as:

$$p_k^t = \sum_{i \in M_k} P(q_t = i|x_1^t) = \sum_{i \in M_k} \frac{\alpha(i,t)}{\sum_{j=1}^{N_S} \alpha(j,t)}, \tag{3}$$

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 all states. The probability p_k^t of model M_k is the sum of the probabilities of all states in model M_k. For each participant, the probabilities for all models are represented by a vector $(p_1^t, ..., p_{N_I}^t)$. We then concatenate the individual vectors from all participants, together with the group 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 for group action clustering.

3.3 Unsupervised G-HMM

For the upper layer, we employ an agglomerative clustering algorithm, recently proposed in the speech community for speaker clustering [2], and that has shown good performance for such a task. The algorithm is based on an ergodic HMM framework with a minimum duration constraint, where the number of clusters and segmentation boundaries are unknown a priori. Each state of the HMM represents a cluster having several identical states in cascade to impose the minimum duration constraint. A three-cluster case is illustrated in Fig.2. The HMM clustering algorithm can be summarized as follows:

1. **Initialization:** Start by over-clustering, i.e. clustering the data into a number of clusters larger than the hypothesized number of actions. The probability density function of each cluster is represented by a Gaussian Mixture Model (GMM) and the parameters of this GMM are estimated using the expectation maximization (EM) algorithm. The initialization for each distribution is done using K-means.

2. **Segmentation:** Obtain the segmentation using the Viterbi algorithm [21] on the current HMM topology and parameters.

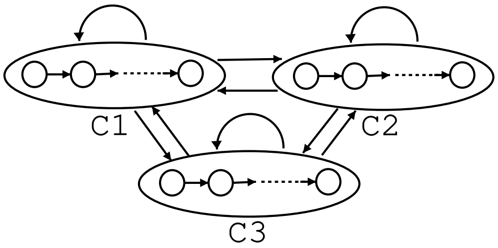

Figure 2: Fully connected HMM topology

3. **Training:** Reestimate the parameters of all clusters based on this segmentation.

4. **Merging:** Search for the best candidate pair of clusters for merging based on the criterion described in [2].

The segmentation-training-merging process is iterated until no more cluster pairs satisfy the merging criterion.

The HMM clustering algorithm has a number of advantages [2]. First, the final number of clusters is decided automatically using a robust merging criterion. Secondly, instead of making local threshold-based decisions, the HMM clustering algorithm produces a global segmentation of the meeting video without using any predefined threshold, which is optimal in the maximum likelihood sense, while avoiding the need for development data. Thirdly, the clustering algorithm can be applied directly on the data sequences, deriving the segmentation in the process without assumptions regarding the number of clusters and their boundaries.

The clustering algorithm can be applied to one individual meeting, as well as to a complete meeting collection, with a minor difference. When clustering a collection, the features for all meetings are concatenated. However, the inter-meeting boundaries are known a priori, so this particular knowledge is used as part of the clustering process.

3.4 Definition of Actions

As an implementation of the proposed framework, we define a set of group actions and individual actions in this section. Motivated by the relevance of turn-taking patterns in meetings [11, 15], we defined a set of $N_G = 8$ group actions based on "multimodal turn-taking" actions, commonly found in meetings. The list is defined in Table 1. The set is somewhat richer than the one defined by other authors [13, 14, 7], as it includes simultaneous occurrence of actions, like *"monologue+note-taking"* which could occur during real situations, like dictating or minute-taking. As discussed in section 1, this group action set is assumed to be non-overlapping and exhaustive for modeling purposes, although such situation does not strictly hold in practice. Also note that this action set would likely be labeled with a good degree of agreement by people (see section 4.1 for details on ground-truth generation), so a fully supervised approach would also be appropriate. For our purposes, this action set is especially useful to thoroughly evaluate the performance of the unsupervised modeling of group actions.

For individual actions, we define a small set ($N_I = 3$) which, as stated earlier, will help bridge the gap between group actions and low-level AV features. The list appears in Table 1. While the list of potentially interesting individual actions in meetings is large, our ultimate goal is to cluster group-level actions defined in Table 1.

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 group features described in the previous subsection. Their detailed description will be presented in section 4.2.

The logical relations between individual actions, group actions, and group features are summarized in Table 2. 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. Our approach is not rule-based, but Table 2 is useful to conceptually relate the two layers.

Table 1: Description of actions

Group 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
Individual action description	
Speaking	one participant speaking
Writing	one participant taking notes
Idle	one participant neither speaking nor writing

Table 2: Relationships between group actions, individual actions and group features. The symbol "$\star$" indicates that the white-board or screen are in use when the corresponding group action takes place. The 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	/	$\star$	

4 Experiments and Results

In this section, we first describe the data set we used in the experiments. We then describe the audio-visual feature extraction process. We later present the performance measures used to evaluate our results. Finally, we present results for group action clustering and discuss our findings.

4.1 Meeting Data Set

We used a public meeting corpus [13], which was collected in a room equipped with synchronized multi-channel audio and video recorders[1]. The sensors include three fixed cameras and twelve microphones. Each meeting consists of four participants seated at a table in a typical workplace setting. 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 are illustrated in Fig.3. Audio was recorded using lapel microphones for all participants, and an eight-microphone array located in the center of the table. The corpus consists of 59 short meetings with five-minute average duration. The group action structure was scripted before recording, according to a group action set simpler than the one we defined here [13], so for our work part of the group actions labels were already available as part of the

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

Table 3: Number of frames *(N_F)* and number of actions *(N_A)* in different data 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
Group Actions	**test**			
	N_F		N_A	
discussion	14450		49	
monologue	7585		26	
monologue + note-taking	6695		23	
note-taking	320		3	
presentation	3345		9	
presentation + note-taking	3865		9	
white-board	265		1	
white-board + note-taking	6875		19	
total	43400		139	

public corpus. We manually relabeled the rest of the group actions, and labeled the entire corpus in terms of individual actions.

Among the 59 meetings, 30 are used as training and the remaining 29 for testing. The number of actions and the number of frames in the different data sets are summarized in Table 3. The number of individual actions is much larger than that of group actions for two obvious reasons. First, for individual action recognition, there are 30 meetings $\times$ 4 participants = 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. For group action clustering, there is no need for a training set.

4.2 Feature Extraction

We now describe the process to extract the two types of features used in this work. Person-specific features are extracted from participants. Group-level features are extracted from the white-board and projector screen regions.

4.2.1 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.

Regarding visual features, for each video frame, the raw image is converted to a skin-color likelihood

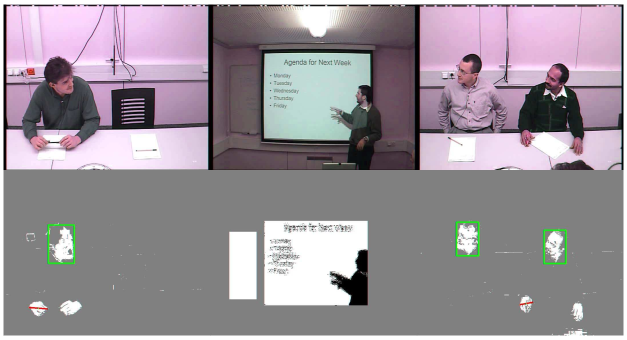

Figure 3: Multi-camera meeting room and visual feature extraction

Table 4: Audio-visual feature list

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

image, using a 5-component skin-color GMM. We use the chromatic color space, known to be less variant to the skin color of different people. 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 represented by its vertical centroid position and eccentricity [22]. The hand blob is represented by its horizontal centroid position, eccentricity, and angle. The motion magnitude for head and right hand are also extracted and summed into one single feature.

For audio, we extracted two types of features using the microphone array and the lapels. 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 [6], 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, as it is well suited for multiparty speech. We used the SIFT algorithm [12] to extract pitch, and a combination of estimators [17] to extract speaking rate.

4.2.2 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 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 Fig.3). On these difference images, we use the average intensity over a grid of 16×16 blocks as features.

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

4.3 Performance Measures

Two measures (*action error rate* and *frame error rate*) were proposed to evaluate results of supervised continuous group action recognition in [7, 14]. However, these measures cannot be used in unsupervised group action

Table 5: Clustering results for individual meetings

Method	N_c mean	N_c std	*aap* (%)	*acp* (%)	*K* (%)
two-layer HMM					
Visual	6.20	2.19	41.4	77.0	56.8
Audio	3.10	1.12	71.3	56.1	63.7
Early Int.	3.59	0.95	69.5	71.3	70.1
MS-HMM	4.17	1.13	72.7	70.8	71.8
A-HMM	3.51	0.78	78.6	70.0	73.8
Baseline: single-layer HMM					
Visual	8.72	2.17	33.6	76.1	50.6
Audio	3.03	1.94	61.1	57.8	58.6
AV	4.10	1.35	68.8	64.2	65.7
Baseline: true number of clusters ($N_c = N_a$)					
B_1	3.93	0.73	64.3	60.1	62.1
B_2			78.4	70.9	74.1
$B_2 - 1$	2.93	0.73	83.5	62.7	71.8
$B_2 + 1$	4.93	0.73	72.6	70.9	71.1

clustering because the labels of the clusters are unknown. Instead, we use three measures used in speaker clustering to evaluate our results: *average cluster purity* (acp), *average action purity* (aap) and overall evaluation criterion K [1, 2]. These measures are explained below. First we define:

- n_{ij}: total number of frames in cluster i by action j
- $n_{\bullet j}$: total number of frames of action j
- $n_{i\bullet}$: total number of frames in cluster i
- N_a: total number of actions
- N_c: total number of clusters
- N: total number of frames

The purity of a cluster $p_{i\bullet}$ and the acp are defined as

$$p_{i\bullet} = \sum_{i=1}^{N_a} \frac{n_{ij}^2}{n_{i\bullet}^2}, \qquad acp = \frac{1}{N} \sum_{i=1}^{N_c} (p_{i\bullet} \times n_{i\bullet}). \tag{4}$$

Similarly, the action purity $p_{\bullet j}$ and the aap are given by

$$p_{\bullet j} = \sum_{i=1}^{N_c} \frac{n_{ij}^2}{n_{\bullet j}^2}, \qquad aap = \frac{1}{N} \sum_{j=1}^{N_a} (p_{\bullet j} \times n_{\bullet j}). \tag{5}$$

The acp gives a measure of how well a cluster is limited to only one action, while the aap gives a measure of how well one action is limited to only one cluster. In the ideal case (one cluster for each group action), $acp = aap = 1$.

However, from only *acp* or *aap* taken separately, it is hard to evaluate the overall performance because *acp* can achieve a high value with more number of clusters than really required, and *aap* can achieve a high value with less number of clusters. In the extreme case, *acp=1* if a cluster has only one frame and *aap=1* if there is only one cluster for the whole meeting. In order to facilitate comparison between systems, an overall evaluation criterion K is defined as follows, where larger K indicates better overall performance.

$$K = \sqrt{acp \times aap}. \tag{6}$$

As a percentage, the average criterion K is around 70% for the robust speaker clustering algorithm described in [1].

4.4 Results and Discussion

To test our approach, we investigated the following combinations of modalities and models for the lower layer:

Early integration, visual-only. The clustering algorithm was applied on the concatenation of the results produced by an early integration *I-HMM* trained on visual-only features, and the visual group features.

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

Early integration, AV. Same as above, but using AV data.

Multi-stream, AV. Same as above, but using the *MS-HMM* approach described in section 3.2 as *I-HMM*.

Asynchronous, AV. Same as above, but using the *A-HMM*.

Additionally, to analyze the benefit of the layered approach, we investigated a number of single-layer clustering schemes, which use the same clustering algorithm directly applied on the low-level features (visual, audio, and AV).

The performance regarding model selection was also studied. We define two baseline systems based on K-means (B_1), and HMM clustering (B_2) respectively, which model an "ideal" case, in which the final number of clusters is exactly the same as the number of group actions (as indicated by the ground-truth). For these systems, the model used for the lower layer was *A-HMM*, as it produced the best performance for the two-layer method (see discussion below).

Finally, we investigated two clustering cases. In the first case, we cluster group actions for each meeting. Usually, the number of group actions within one meeting is less than the complete set of eight actions. In the second case, we cluster the whole test meeting collection, which produces a segmentation for each meeting where segments belonging to the same cluster get consistent labels across the corpus. In this case, there are eight group actions.

Parameter Selection. For the individual action layer, parameters were selected by six-fold cross-validation, splitting the training set into training and validation subsets. For the group action layer, we obtained results by varying the number of initial clusters (10-30), the number of Gaussians (5-10), and the minimum duration of each cluster (15-30s). In Tables 5-6, the results for the number of clusters *(N_c)* are shown in terms of mean and standard deviation. We report mean values for average action purity *(aap)* and cluster purity *(acp)*, and for the overall criterion (K).

The results can be summarized as follows.

Single- vs. multi-modality and single- vs. two-layer HMM. For both the single- and the two-layer cases, the use of AV features produced better results than using only one modality. Audio-only features were more discriminant than video-only, which is not surprising given the type of group actions we addressed. We noticed that methods using audio features got high *aap* and low *acp* while methods using video features showed the opposite trend. This is because, according to the ground-truth, the number of clusters (N_c) was usually underestimated using audio, while overestimated using visual features. Audio-only features thus seem to be described better by simpler models, while visual-only features describe a more complex cluster structure. Additionally, the layered approach outperformed the single-layer method under the same conditions (using one or multiple modalities, and when clustering individual meetings or the whole data set). Given the large total number of frames ($> 43,000$), these improvements are significant, which confirms the effectiveness of the layered approach, and the multimodal nature of group actions in meetings.

Comparison between I-HMM methods. Although multi-stream HMM improved over early integration, the asynchronous HMM produced the best results among all HMM systems for the two meeting clustering cases. This indicates that the probability-based features obtained from this model were more discriminative, and suggests the presence of asynchrony between the audio-visual streams for individual actions. In Tables 5-6, both *acp* and *asp* of *A-HMM* are above 70%. This means that more than 70% of all group actions are in the right clusters, while more than 70% of all clusters are composed of data from the same group action.

Comparison with "ideal" baseline systems. The layered method using AV features outperformed the K-means baseline (B_1), while performed slightly worse than HMM clustering baseline (B_2). We can also see that with a slight increase/decrease of the number of clusters, the performance of this baseline system decreased. (In Tables 5-6, "$B_2 - 1$" and "$B_2 + 1$" denote the baseline system, in which we deliberately increase or decrease the number of clusters by 1.) Interestingly, the best two-layer HMM method outperforms these two cases, which somewhat suggests that our approach is not too far from the "ideal" case.

Table 6: Clustering results for meeting collection

Method	N_c		*aap (%)*	*acp (%)*	*K (%)*
	mean	*std*			
two-layer HMM					
Visual	11.67	2.16	31.0	47.2	38.2
Audio	3.50	2.65	77.7	41.4	56.7
Early Int.	10.60	1.93	70.9	65.7	68.3
MS-HMM	7.28	1.41	74.8	65.2	69.8
A-HMM	7.10	1.70	74.0	70.5	72.2
Baseline: single-layer HMM					
Visual	16.33	4.08	20.2	46.3	30.6
Audio	3.16	2.40	76.1	33.6	50.6
AV	6.73	2.51	64.3	60.1	62.1
Baseline: true number of clusters ($N_c = N_a$)					
B_1	8	0	47.3	51.1	49.2
B_2			71.5	73.8	72.6
$B_2 - 1$	7	0	74.5	67.1	70.7
$B_2 + 1$	9	0	63.9	78.5	70.8

Single meeting vs. entire meeting collection. The results of clustering the whole collection are slightly worse than the results of clustering single meetings for the multimodal layered models (between 1.6-2.0%); the degradation is more pronounced for the single-modality approaches. This decrease in the clustering quality could be explained by the larger variation in the data (the number of meeting participants in the test set taken as a whole is 10), but mainly by the increasing possibility of overlap between different group actions in the feature space, due to the larger number of actions. Note however that clustering the whole corpus generates consistent action labels across meetings; this important benefit was traded by the decrease in performance.

Model selection. For both individual meeting clustering and whole collection clustering, the methods using AV features obtained a number of clusters closer to the true number of actions. For the first case, there are 3.93 group actions on average in the ground-truth. The average number of clusters found using AV features ranges from 3.51 to 4.17, which is close to the true number (Table 5). For the second case, there are 8 group actions in the ground-truth. The two-layer AV systems *MS-HMM* and *A-HMM* both converged around 7 clusters (Table 6), which is in good accordance with the true number, although slightly underestimated.

To evaluate the quality of the clustering results, we display the found clusters and ground-truth actions in Fig.4, for the top 13 meetings ranked by decreasing order, based on the criterion K (the symbol $M_{\#}$ is the meeting index in the test set). Dashed-line rectangles denote automatic clusters (with labels $\{1, 2, ...\}$), which are compared against the ground-truth actions denoted by solid-line rectangles, showing *discussion* (D), *monologue* (M), *monologue + note-taking* (MN), *note-taking* (N), *presentation* (P), *presentation + note-taking* (PN), *white-board* (W) and *white-board + note-taking* (WN). The left and right columns of Fig.4 show the results of clustering individual meetings and the entire meeting collection, respectively. For both cases, we can see that for meetings with large overall criterion K, the obtained alignments between clusters and actions are better. The results degrade with decreasing K. Notice that, for the case of clustering the meeting collection (Fig.4: right-column), cluster labels are consistent across meetings. For example, most clusters with label “3” correspond to “MN” (*monologue + note-taking*) group action, and clusters with label“1” often correspond to the “D” (*discussion*) action.

5 Conclusions

In this paper, meetings were defined as sequences of multimodal group actions. We addressed the problem of clustering group actions, proposing a layered HMM framework to decompose the group action clustering problem into two subproblems. The first layer maps low-level AV features into probability-based, individual-action features. The second layer groups such features into clusters, which correspond reasonably well to group actions. Experiments on a public meeting corpus demonstrate the effectiveness of our framework. For

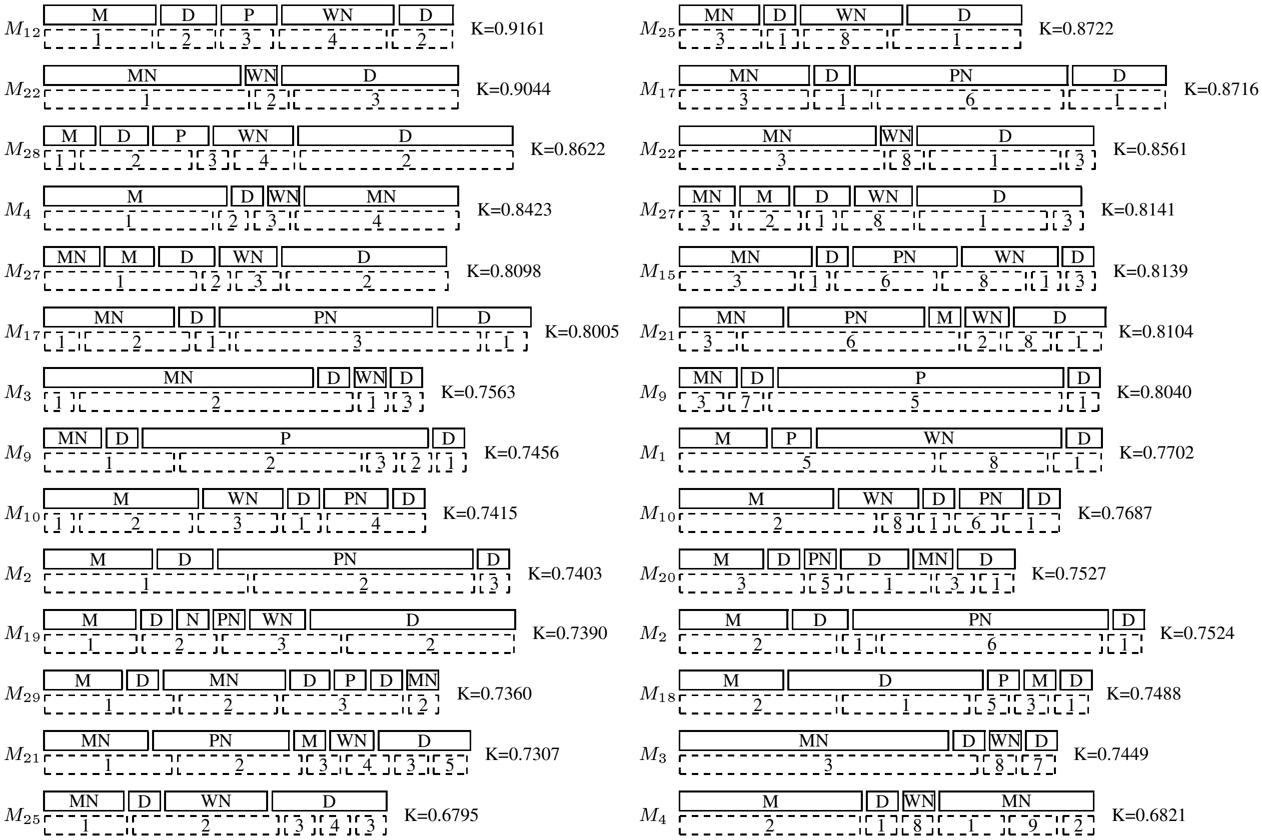

Figure 4: Results of clustering individual meetings (left column), and entire meeting collection (right column).

future work, we will consider the use of semi-supervised approaches to assign semantic labels to the clustering outputs, and the extension of our approach to other dictionaries of group actions.

References

[1] J. Ajmera, H. Bourlard, I. Lapidot, and I. McCowan. Unknown-multiple speaker clustering using HMM. In ICSLP, Colorado, 2002.

[2] J. Ajmera and C. Wooters. A robust speaker clustering algorithm. In *IEEE ASRU Workshop*, 2003.

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

[4] S. Bengio. An asynchronous hidden Markov model for audio-visual speech recognition. In *Proc. NIPS*, 2003.

[5] 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.

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

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

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

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

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

[11] R. Krauss, C. Garlock, 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),1977.

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

[13] 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.

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

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

[16] 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. of the HLT Conference*, San Diego, CA, March 2001.

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

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

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

[20] N. Oliver, B. Rosario, and A. Pentland. A Bayesian computer vision system for modeling human interactions. *IEEE PAMI*, 22(8), Aug. 2000.

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

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

[23] 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.

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

[25] 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.

[26] L. Zelnik-Manor and M. Irani. Event-based video analysis. In *Proc. CVPR*, Dec. 2001.

[27] H. Zhong, J. Shi, and M. Visontai. Detecting unusual activity in video. In *Proc. CVPR*, June. 2004.