# *Capture, Recognition, and Visualization of Human Semantic Interactions in Meetings*

Zhiwen Yu[1], Zhiyong Yu[1], Hideki Aoyama[2], Motoyuki Ozeki[3], Yuichi Nakamura[2]
[1] Shool of Computer Science, Northwestern Polytechnical University, P. R. China
zhiwenyu@nwpu.edu.cn
[2] Academic Center for Computing and Media Studies, Kyoto University, Japan
[3] Graduate School of Science and Technology, Kyoto Institute of Technology, Japan

*Abstract*—**Human interaction is one of the most important characteristics of group social dynamics in meetings. In this paper, we propose an approach for capture, recognition, and visualization of human interactions. Unlike physical interactions (e.g., turn-taking and addressing), the human interactions considered here are incorporated with semantics, i.e., user intention or attitude toward a topic. We adopt a collaborative approach for capturing interactions by employing multiple sensors, such as video cameras, microphones, and motion sensors. A multimodal method is proposed for interaction recognition based on a variety of contexts, including head gestures, attention from others, speech tone, speaking time, interaction occasion (spontaneous or reactive), and information about the previous interaction. A support vector machines (SVM) classifier is used to classify human interaction based on these features. A graphical user interface called MMBrowser is presented for interaction visualization. Experimental results have shown the effectiveness of our approach.**

***Keywords-smart meeting; human interaction; interaction capture; interaction recognition; visualization***

## I. INTRODUCTION

Meetings are important events in our daily lives for purposes of problem solving, information exchange, and knowledge sharing and creation. Thanks to pervasive computing technologies, meetings tend to be *smart*. A smart meeting system can assist people in a variety of tasks, such as scheduling meetings, taking notes, sharing files, and browsing minutes after a meeting [1, 2, 3, 4, 5, 6].

Current smart meeting systems mainly support meeting preparation (before meeting), information exchange (in meeting), and content review (post meeting) [7], whereas little research has been conducted on social aspects. Meetings encapsulate a large amount of social and communication information. Group social dynamics is particularly important for understanding *how* a conclusion was reached, e.g., whether all members agreed on the outcome, who did not give his opinion, who spoke a little or a lot. They are further useful for determining whether the meeting was well organized and the conclusion well reasoned. An important characteristic of group social dynamics is human interactions.

Unlike physical interactions such as turn-taking and addressing (who speaks to whom), the human interactions considered here are defined as social behaviors among meeting participants with respect to the current topic, such as proposing an idea, giving comments, expressing a positive opinion, and requesting information. When incorporated with semantics (i.e., user intention or attitude toward a topic), interactions are more meaningful for evaluating conclusion drawing and meeting organization. Understanding how people are interacting can also be used to support a variety of pervasive systems [8]. For example, previous meeting capture and access systems could use this technology as a smarter indexing tool to access different parts of the meetings.

In this study, we focus on manipulation of this type of human *semantic* interactions in meetings, specifically capturing, recognizing, and visualizing them. A smart meeting system was built for this purpose. We adopt a collaborative approach for capturing interactions by employing multiple sensors, such as video cameras, microphones, and motion sensors. A multimodal method is proposed for interaction recognition based on a variety of contexts, such as head gestures, attention from others, speech tone, speaking time, interaction occasion, and information about the previous interaction. A support vector machines (SVM) [9] classifier is used to classify human interactions based on these features. A graphical user interface, the MMBrowser (**M**ultimodal **M**eeting **Browser**), is presented for visualizing interactions. It can be used by the people who missed the meeting for quick reviewing and understanding the discussion topics and social dynamics of the meeting. It also offers real-time feedback for improving participants' meeting skills.

The remainder of this paper is organized as follows. Section II discusses related work. Section III introduces the concept of human semantic interactions and its types. In Section IV, we present the architecture of our smart meeting system. Collaborative interaction capture, multimodal interaction recognition, and graphical interaction visualization are described in Sections V, VI, and VII, respectively. Section VIII presents our evaluation results. A discussion of the current approach and result is given in Section IX. Finally, we conclude the paper in Section X.

## II. RELATED WORK

A number of smart meeting systems have been developed in the past using pervasive computing technologies. The Conference Assistant [2] combines context-awareness and wearable computing technologies for enhancing attendee interactions with the environment and other attendees. EasyMeeting [4] aims at providing services to meeting participants based on their situational requirements, such as preparing the data projector and setting the lighting and

978-1-4244-5328-3/09/$25.00 ©2009 IEEE

temperature in the room. The FXPAL conference room [6] monitors a meeting using a variety of devices, and provides tools for accessing and browsing captured meetings. TeamSpace [7] offers supports of meeting preparation, capture and access for distributed workgroups. Ahmed et al [10] built a smart meeting room that detects the beginning and end of a meeting, and supports ephemeral group communication. The Intelligent Meeting Room [11] recognizes activities in a meeting room, such as a person locating in front of the whiteboard, a lead presenter speaking, and other participants speaking. SMeet [12] enables multi-party meeting by integrating user-friendly pointing/tracking interactions, high-resolution tiled displays, hybrid multicast-based networking, and high-quality media services. The shared goal of these works was making an intelligent meeting room that provides adaptive services and relevant information using ubiquitous sensors and context-awareness techniques. This differs from our study, which focuses on capturing human interactions and understanding their semantic meaning.

Numerous studies have focused specifically on physical interaction recognition and visualization in meetings. Stiefelhagen et al [13] proposed an approach for estimating who was talking to whom, based on tracked head poses of the participants. AMI project [14] deals with interaction issues, including turn-taking, gaze behavior, influence, and talkativeness. The Meeting Mediator at MIT [15] detects overlapping speaking time and interactivity level in a meeting by using Sociometric badges and then visualizes the information on mobile phones. Sumi et al [16] analyzed user interactions (e.g., gazing at an object, joint attention, and conversation) during poster presentation in an exhibition room. DiMicco et al [17] presented visualization systems for reviewing a group's interaction dynamics, e.g., speaking time, gaze behavior, turn-taking patterns, and overlapped speech in meetings. Otsuka et al [18] used gaze, head gestures, and utterances in determining interactions regarding who responds to whom in multiparty face-to-face conversations. Although several other systems, e.g., [19], [20], and [21], seem to model and analyze interactions, they deal with very high-level group actions, such as presentation, general discussion, and note-taking. In general, the above-mentioned systems mainly focus on the analysis of physical interactions between participants without any relation to the topics; in other words, they do not include semantic meanings in the analysis. Therefore, they cannot determine clearly a participant's attitude or role in a discussion.

A few systems attempted to analyze semantic information in meeting interactions. Hillard et al [22] proposed a classifier for the recognition of an individual's agreement or disagreement utterances using lexical and prosodic cues. The Discussion Ontology [23] was proposed for obtaining knowledge such as a statement's intention and the discussion flow in meetings. Both systems are based on speech transcription that extracts features such as heuristic word types and counts. Garg et al [24] proposed an approach to recognize participant roles in meetings. The speech act theory [25] determines implicit actions (e.g., greeting, apologizing, requesting, promising) behind human speech based on utterances. Our system differs in several aspects. First, besides detecting positive or negative opinions [22], we analyze more types of human semantic interactions in topic discussion such as proposing an idea, requesting information, and commenting on a topic. Second, our system adopts a multimodal approach for interaction recognition by considering a variety of contexts (e.g., head gestures, face orientation, speech tone), which provides robustness and reliability. Third, we present a visualization interface for browsing interactions, as well as relationships between them.

Another area of related work is about meeting browser, which has its root in Waibel et al's work on summarizing and navigating meeting content [26]. Afterwards several other similar systems were developed. Rough'n'Ready [27] provides audio data browsing and multi-valued queries, e.g., specifying keywords as topics, names, or speakers. Ferret [28] offers interactive browsing and playback of many kinds of meeting data including media, transcripts and speaker segmentations. Geyer et al [29] discussed various ways of indexing meeting records and built a viewer for accessing the content. Jaimes et al [30] proposed a meeting browser enhanced by human memory-based meeting video retrieval. Junuzovic et al [31] presented a 3D interface for viewing speaker-related information in recorded meetings. Besides meeting content that existing systems mainly dedicated to, our visualization system provides a multimodal and interactive way for browsing human interactions.

## III. Human Semantic Interaction

Meetings usually encapsulate a large amount of communicative statements and semantic relationships between them that form an interaction network. To enrich knowledge about a meeting with social group dynamics, we need to analyze not only physical interactions such as turn-taking and addressing (who speaks to whom) between participants, but also semantic meanings behind the physical actions.

Human semantic interactions are defined as social behaviors among meeting participants with respect to the current topic. Various interactions imply different user roles, attitudes, and intentions about a topic during a discussion. With semantics incorporated, interactions are more meaningful in understanding conclusion drawing and meeting organization.

The definition of interaction types naturally varies according to usage. For generalizability, we create a set of interaction types based on a standard utterance-unit tagging scheme [32]. It includes the following seven categories of human semantic interactions: *propose*, *comment*, *acknowledgement*, *requestInfo*, *askOpinion*, *posOpinion*, and *negOpinion*. The detailed meanings are as follows: *propose*—a user proposes an idea with respect to a topic; *comment*—a user comments on a proposal, or answers a question; *acknowledgement*—a user confirms someone else's comment or explanation, e.g., "yeah," "uh huh," and "OK"; *requestInfo*—a user requests unknown information about a topic; *askOpinion*—a user asks someone else's opinion about a proposal; *posOpinion*—a user expresses a positive opinion, i.e., supports a proposal; and *negOpinion*—a user expresses a negative opinion, i.e., disagrees with a proposal.

## IV. System Architecture

We developed a prototype smart meeting system for capture, recognition, and visualization of human semantic interactions during a discussion. Fig. 1 illustrates the system architecture, which consists of the following three layers: Interaction Capture, Interaction Recognition, and Interaction Visualization.

In this design, Interaction Capture is the physical layer that deals with capture environment, devices, and methods. Video cameras, microphones, and motion sensors, are used for recording meeting content and tracking participants' head movement.

Interaction Recognition serves as the structural layer, which analyzes contents of the generated audio-visual data and motion data. The Speech Recognition Engine extracts speech features (tone and speaking time) from audio data. The Motion Data Processor is responsible for analyzing sensing data derived from motion sensors, and measuring people's head gestures (e.g., nodding) and face orientation. The Annotator is used for manually labeling features from audio-visual data. Based on low-level features, the SVM Classifier recognizes human interactions. The Interaction Recognition layer makes the meeting content meaningful and provides support for the upper layer.

Interaction Visualization is the presentation layer, offering an interactive user interface for browsing human interactions as well as meeting content.

The details of interaction capture, recognition, and visualization are described in the following sections.

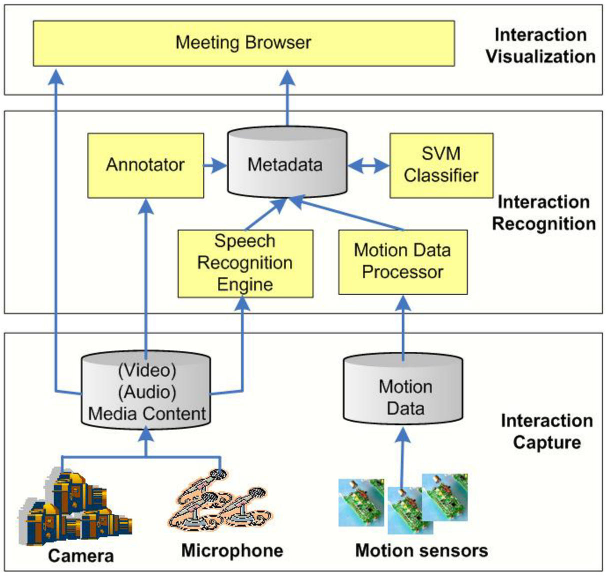


Figure 1. Smart meeting system architecture

## V. Collaborative Interaction Capture

We use multiple devices for capturing human interactions. Fig. 2 is a snapshot of our capturing system setup. An overview of the capture environment is shown in Fig. 2a; Fig. 2b shows a participant wearing a microphone and motion sensors.

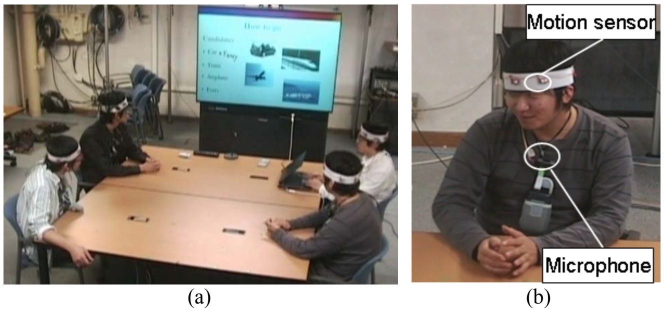


Figure 2. Capture system: (a) an overview of the capture environment; (b) a participant wearing a microphone and motion sensors

### A. *Video Capture*

Six video cameras (SONY EVI-D30) are deployed in our capture system. Four capture the upper-body motions of four participants (breast shot), one camera records presentation slides (screen shot), and the other captures an overview of the meeting including all participants (overview shot).

The breast-shot camera is controlled automatically based on face recognition of its assigned participant. The face region is extracted from the image of each breast-shot camera using face recognition software (provided by Toshiba Ltd.). If the center of the face does not coincide with the center of the image, the camera is adjusted until the face appears at image center. If the face is already at the center, the camera is zoomed in or out so that the appropriate size of the face in the image is obtained.

The video signal from each camera is input to an encoder board and stored in the MPEG2/PS format with a frame size of 720 × 480 pixels, frame rate of 29.97 fps, and bit rate of 2 Mbps.

### B. *Audio Capture*

To capture audio data, a head-mounted microphone (SHURE WH30XLR) is attached to each participant. A fixed microphone records global meeting sound. The audio signal (MPEG1-Layer II 48000 Hz [CBR Stereo] 384 kbps) is also imported into the encoder board in sync with the video signal. The computers attached with the encoder boards are synchronized with each other by Network Time Protocol (NTP).

### C. *Head Tracking*

We use an optical motion capture system (PhaseSpace IMPULSE) [33] for head tracking. It mainly consists of three parts: LED (Light-Emitting Diode) module, camera, and server. The tracking system uses multiple CCD (Charge-Coupled Device) cameras for three-dimensional measurement of LED tags (sensors), and obtains their exact position. We placed six LED tags around each person's head. The LED tags are scanned 30 times per second, and position data can be obtained in real time with less than 10-ms latency. Through the three-dimensional position data, head gestures (e.g., nodding) and face orientation can be detected.

## VI. Multimodal Interaction Recognition

### A. Context Extraction

The contexts considered in our interaction recognition include head gestures, attention from others, speech tone, speaking time, interaction occasion, and the type of the previous interaction. These features are extracted from the raw audio-visual data and motion data.

*a) Head gesture recognition:* Head gestures (e.g., nodding and shaking of the head) are very common, and are often used in detecting human response (acknowledgement, agreement or disagreement).

We determine nodding through the vertical component of the face vector calculated from the position data. The nodding recognition method is schematically shown in Fig. 3a. We first determine the maximum and minimum value of the vertical component of the face vector in a time window (here we set one second). Next, we calculate $\theta_1$ (the difference between the maximum and minimum values) and $\theta_2$ (the difference between the current value and the minimum). Then the nodding score is calculated using the following formula.

$$Score = \frac{\theta_1}{11.5} \times \frac{\theta_2}{11.5} \tag{1}$$

Here, 11.5 is empirically set as the normalization constant. If the calculated score is above the preset threshold (e.g., 0.80), we consider that the head gesture is nodding.

Head-shaking is detected through the horizontal component of the face vector. The method is illustrated in Fig. 3b. We first calculate a projection vector of the face vector on the horizontal plane. To distinguish head-shaking from normal face orientation change, we count the switchback points of the projection vector's movement in a time window (e.g., 2s). If the number of the switchback points is larger than the preset threshold (e.g., 2), we consider that it is a shaking action.

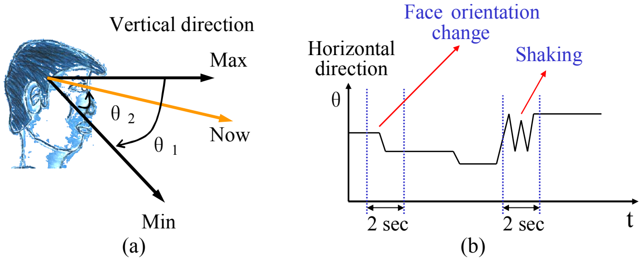


Figure 3. Head gesture recognition method: (a) nodding; (b) shaking

*b) Detection of attention from others:* Attention from others is an important determinant of human interaction. For example, when a user is proposing an idea, he is usually being looked at by most of the participants. The number of people looking at the target user during the interaction can be determined by their face orientation. We measure the angles between the reference vectors (from the target person's head to the other persons' heads) and the target user's real face vector (calculated from the position data). Face orientation is determined as the one whose vector makes the smallest angle.

*c) Speech tone and speaking time extraction:* Speech tone indicates whether an utterance is a question or a non-question statement. Speaking time is another important indicator in detecting the type of the interaction. When a user puts forward a proposal, it usually takes relatively longer time, but it takes a short time to acknowledge or ask a question. Speech tone and speaking time are automatically determined using the Julius speech recognition engine [34]. It segments the input sound data into speech durations by detecting silence interval longer than 0.3 sec. We classify segments as questions or non-questions using the pitch pattern of the speech based on Hidden Markov Models [35] trained with each person's speech data. The speaking time is derived from the duration of a segment.

*d) Identification of interaction occasion:* An interaction is either spontaneous or reactive. In the former case, the interaction is initiated spontaneously by a person (e.g., proposing an idea or asking a question). The latter denotes an interaction triggered in response to another interaction. Discussion tags [23] can be used to explicitly indicate the interaction occasion. We manually label this feature in our current system using Anvil [36], which is a widely used video annotation tool with user-defined hierarchical layers and attributes.

*e) Detection of type of previous interaction:* The type of the previous interaction also plays an important role in detecting the current interaction. It is intuitive that certain patterns or flows frequently occur in the course of a discussion in a meeting. For instance, *propose* and *requestInfo* are usually followed by a comment. This feature can be obtained from the recognition result of its previous interaction.

### B. Interaction Recognition Based on Support Vector Machines

With the basic context aggregated, we adopt the SVM classifier for interaction recognition. Given the features of an instance, the SVM classifier sorts it into one of the seven classes of interactions. SVM has been proven to be powerful in classification problems, and often achieves higher accuracy than other pattern recognition techniques. It is well known for its strong ability to separate hyper-planes of two or more dimensions.

Given a training set of instance-label pairs ($x_i$, $y_i$), $i = 1, \dots, l$, where $x_i \in R^n$, and $y \in \{1, -1\}^l$, SVM requires the solution of the following optimization problem:

$$\begin{aligned} &\min_{w,b,\xi} \quad \frac{1}{2} w^T w + C \sum_{i=1}^{l} \xi_i \\ &y_i (w^T \phi(x_i) + b) \geq 1 - \xi_i \\ &\xi_i \geq 0, \end{aligned} \tag{2}$$

where $\boldsymbol{w}$ is the weight vector, and $b$ is the bias. The training vectors $x_i$ are mapped into higher-dimensional space using the function $\phi$. Then the SVM finds a linear separating hyperplane with the maximal margin in this higher dimensional space. $C >$

0 is the penalty parameter of the error term, and $K(x_i, x_j) \equiv \phi(x_i)^T \phi(x_j)$ is called the kernel function. Please refer to [37] for more details about the SVM estimation and kernel selection.

Our system uses the LIBSVM library [38] for classifier implementation. The default kernel type is the radial basis function (RBF): $K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2), \gamma > 0$ . Two parameters are important while using LIBSVM: $C$ and $\gamma$. Good ($C$, $\gamma$ ) enables the classifier to accurately predict unknown data (i.e., testing data). In our system, we adopt a "grid search" to find the best parameter $C$ and $\gamma$ using cross-validation. Basically pairs of ($C$, $\gamma$ ) are tried and the one with the best cross-validation accuracy is picked.

The meeting content is first segmented into a sequence of interactions. Sample interactions are selected and fed to the SVM as training data, while others are used as a testing set (details of training and test data are presented in Section VIII).

## VII. Interaction Visualization

To efficiently review and understand the human interactions, we present a tool called MMBrowser (**M**ultimodal **M**eeting **Browser**) for interaction visualization. It acts as the interface between the smart meeting system and the end users. The visualization tool allows those who missed the meeting (e.g., those on vacation) to view it later. It helps such people to quickly understand the meeting content and human interactions in a multimodal and interactive way using graphics, video images, speech transcription, and video playback.

Besides post-meeting review, the interaction visualization tool mirrors group activity so the group can monitor its own operation. It helps meeting in the organization and improves people's meeting participation skills. For instance, knowing the current status of the meeting (e.g., did all members agree on a conclusion, who was quiet, who was extroverted), the organizer can make some adjustments to make the meeting more efficient. Social psychologists suggest that a group can improve its interaction and consequently its productivity by understanding its emotional and social interactions [39]. In addition, balanced participation is essential for solving a problem properly. Through interaction visualization, the members become aware of their own and others' behavior in a discussion (e.g., one person speaks for a long time, two people always discuss in a subgroup), and can then make changes to increase the group's satisfaction with the discussion process. An individual, a group, and an entire organization could benefit from the participants' awareness of their own behavior during meetings [40].

Fig. 4 shows the interfaces for interaction visualization. An overview of the MMBrowser is presented in Fig. 4a. It consists of seven rows. At the top is the timeline, while the images of the overview video are sampled and presented in the second row. The following four rows display the interaction dynamics of the four participants using thumbnails of each. We use rectangles filled with different colors to represent different types of interactions (e.g., pink for the interaction of *propose*). The interaction's abbreviated name appears at the top of each rectangle, and speech transcription is placed within the rectangle. The length of the rectangle denotes the speaking time. The arrow between two interactions denotes that one interaction (the start of the arrow) triggers the other (the end of the arrow). Interactions without arrows pointing to them are spontaneous. We use a blue bar to represent the degree of attention from others during the interaction. The bottom lane displays images extracted from the screen video.

The visualization tool provides zoom in and zoom out functions, which allow people to browse meeting information with different resolution levels and from different viewpoints. Fig. 4b shows the zoomed-out view of the main interface. End users or participants can have a bird's eye view of interactions in the meeting.

Using the MMBrowser, the end user can easily access an area of interest (where he/she wants to know the details) in the original video (i.e., overview, individual, and screen videos). The user can simply select the area by dragging the mouse (Step 1 in Fig. 4a), and click the "play" button (Step 2 in Fig. 4a). The corresponding video segment will play back. Fig. 4c shows the extracted video segment playing, corresponding to the selected time and duration.

## VIII. Evaluation Results

We evaluate the following three aspects of performance of our system: (1) the efficacy of context extraction, (2) the efficacy of interaction recognition, and (3) the effectiveness of interaction visualization. The evaluation data includes two meetings, one soccer preparation meeting (23 min, talking about the players and their role and position in a coming match) and one trip planning meeting (18 min, discussing about time, place, activities and transportation of a summer trip). Both meetings had four participants.

### *A. Evaluation of Context Extraction*

We first evaluate the performance of context extraction, including detection of head gestures (nodding and shaking), speech tone (questions), and attention from others (face orientation). We did not include interaction occasion (whether spontaneous or reactive) in this evaluation as they were manually labeled.

The recognition rate is adopted to evaluate the accuracy of our detection mechanisms. It is the ratio between the number of correctly recognized objects and the total number of objects. The results are reported in Table I. We ultimately achieved accuracies of 76.4%, 80.0% and 72.2% in the detection of head nodding, head shaking and face orientation, respectively. The recognition rate of question tone is a little low. The reason might be that the Julius speech recognition engine omitted numerous short sentences, but questions often exist in these short sentences. It also verified that speech recognition in meeting is challenging due to highly conversational and noisy nature of meetings, and lack of domain specific training data [41].

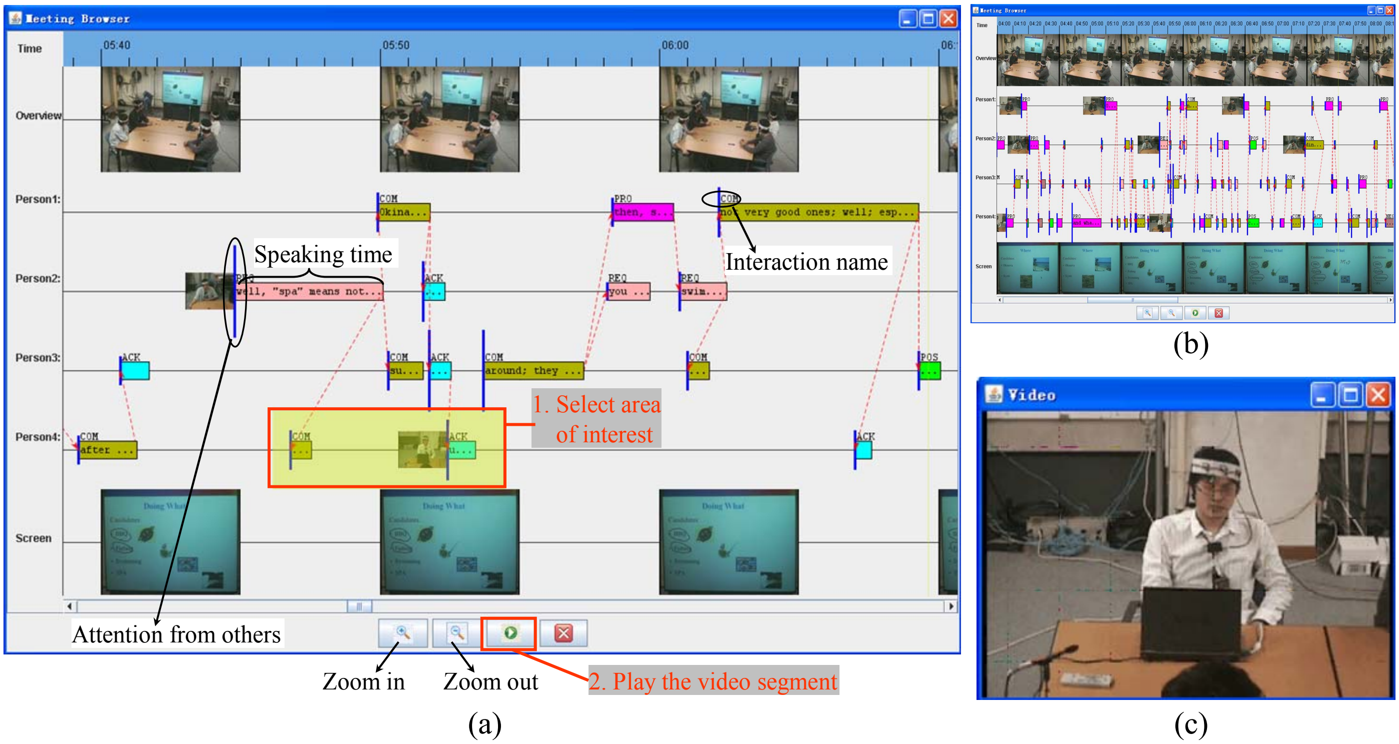


Figure 4. Interaction Visualization: (a) Meeting Browser overview; (b) Zoomed-out view; (c) Video segment playing

TABLE I. CONTEXT EXTRACTION RESULTS

| Context | Recognition rate |
|---|---|
| Head nodding | 76.4% |
| Head shaking | 80.0% |
| Question tone | 65.0% |
| Face orientation | 72.2% |

### *B. Evaluation of Interaction Recognition*

We further evaluate interaction detection by measuring the recognition rate. Through the video and audio data, we manually labeled 518 interactions in the soccer meeting. We chose 370 of them as a training set, and the other 148 were used for testing. Our system achieved a recognition rate of 74.3%. Table II shows the result of different interaction recognition. We can observe that the algorithm performs well to recognize categories of *comment*, *acknowledgment*, *reqestInfo*, and *posOpinion*. It also means that these four types of interactions can be easily detected with the current feature setting. The other three categories (*propose*, *askOpinion*, and *negOpinion*) are difficult to recognize. Specifically, some *propose* interactions were wrongly classified into the category of *comment* and all *askOpinion* interactions were recognized as *reqestInfo*. Surprisingly, recognition is much better in detecting *posOpinion* than *negOpinion*. We would have expected the *negOpinion* class to perform well as we used head-shaking to signal disagreement. However there were very few shaking actions during the meeting. The participants usually expressed their positive opinions explicitly but implicitly gave a disagreement.

TABLE II. RESULT OF DIFFERENT INTERACTION RECOGNITION

| Interaction type | Recognition rate |
|---|---|
| propose | 22.2% |
| comment | 76.3% |
| acknowledgment | 86.7% |
| reqestInfo | 83.3% |
| askOpinion | 0.0% |
| posOpinion | 80.0% |
| negOpinion | 0.0% |
| Total | 74.3% |

We also examined the recognition performance on different persons. The result is shown in Table III. We can see that the recognition differences are very small between the participants except that the accuracy of Person A is a little higher than the others. It indicates that our method can work with different persons.

TABLE III. RESULT OF INTERACTION RECOGNITION OF DIFFERENT PERSONS

| Person | Recognition rate |
|---|---|
| Person A | 87.5% |
| Person B | 73.1% |
| Person C | 70.4% |
| Person D | 76.5% |

We then used the training data of soccer meeting to test the trip meeting and achieved an accuracy of 72.6%. It nearly equals the result of training and testing with the same meeting (74.3%). This indicates that our model can be trained once and used to test other meetings.

To test the effect of different features on interaction recognition, different context sets were configured and fed into the SVM classifier. In this test, the trip meeting data was used. A total of 406 interactions were labeled of which 311 were chosen for training and the other 95 for testing. The context set configuration and recognition results are presented in Table IV. There are a total of five different context sets in this experiment. Set 1 is a complete configuration with all the six categories of contexts. Sets 2, 3, 4, and 5 include all contexts except head movement (i.e., head gesture and attention from others), speech features (i.e., speech tone and speaking time), interaction occasion, or type of previous interaction, respectively. With all contexts, the system achieved a recognition rate of 74.7%. We can also observe that without the context of speech or head movement, the recognition accuracies decrease considerably, indicating that these contexts play a significant role in the detection of the interactions. On the other hand, interaction occasion and previous interaction are not as important, because they do not have much influence on the detection results.

TABLE IV. INTERACTION RECOGNITION RESULTS WITH DIFFERENT CONTEXT SETS

| Context sets | Recognition rate |
|---|---|
| Set 1 - all contexts: {c1, c2, c3, c4, c5, c6} | 74.7% |
| Set 2 - all contexts except head movement: {c3, c4, c5, c6} | 66.8% |
| Set 3 - all contexts except speech features: {c1, c2, c5, c6} | 57.9% |
| Set 4 - all contexts except interaction occasion: {c1, c2, c3, c4, c6} | 70.5% |
| Set 5 - all contexts except type of previous interaction: {c1, c2, c3, c4, c5} | 72.6% |

Note: c1 - head gesture, c2 - attention from others, c3 - speech tone, c4 - speaking time, c5 - interaction occasion, c6 - type of previous interaction.

### C. Evaluation of Interaction Visualization

For the visualization tool, we first evaluate its efficiency for understanding the meeting topics and social interactions. Before the tests, we manually identified a number of observations of interest from the trip meeting video. Observations of interest include the topics that meeting participants might be interested in, such as where to go, when to go, how to travel, and what to do. The observations also include group interaction dynamics, e.g., who spoke a little or a lot, who received the most attention from the others, who proposed a lot of ideas, which topic was discussed the longest, and whether all members agreed on a particular topic.

Nine subjects (students and staff in the media center of Kyoto University) were invited to take part in this experiment in March, 2008. They did not know anything about the meeting content before the test. The subjects were divided into two groups, Group A (three males and two females) and Group B (four males). They were asked to review the meeting, and answer questions designed based on the manually identified observations of interest (10 questions each for topics of interest and interaction dynamics). Group A was invited to use our meeting browser, whereas Group B watched the raw video. Both groups were required to finish the test within 10 min (the entire trip meeting lasted about 18 min). The time was limited to prevent a simple playback of the entire meeting to determine observations of interest. We then calculated the average scores of each group in correctly identifying the topics of interest and interaction dynamics.

The result is illustrated in Fig. 5. It shows that with our visualization tool, Group A was able to correctly answer about 60% of the questions in both categories, i.e., meeting topics and social interactions. We see that Group A outperformed Group B in content understanding, but the difference is not substantial. For social interaction understanding, Group A performed much better than Group B. The average score of Group B is only 15%, which is much less than that of Group A. This result verifies the effectiveness of our visualization tool in helping users understand the meeting content and especially group interaction dynamics.

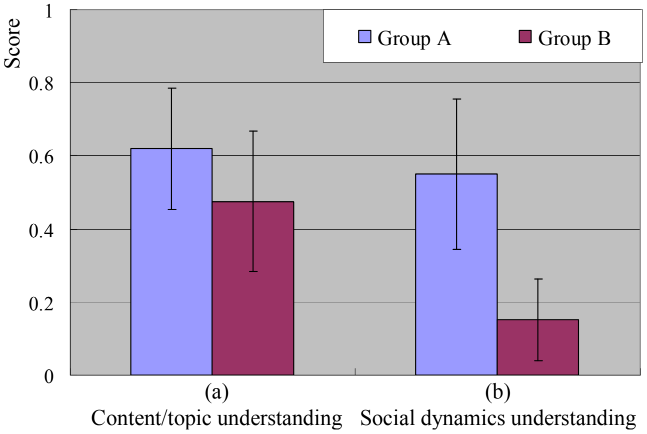


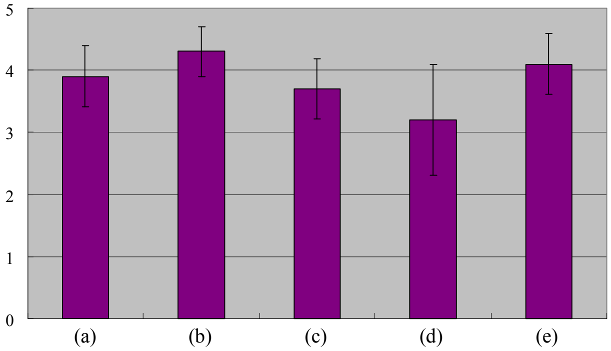
Figure 5. Experimental results of meeting topic and social interaction understanding

For evaluating the usability of our visualization tool, a user study was conducted. The members of Group A were asked to complete a brief survey based on their experience of using our system for reviewing meeting content and human interactions. We mainly measured user impressions of interface usefulness and acceptance of the visualization tool. The questionnaire and result are shown in Fig. 6.

In general, the participants were satisfied with the performance of the tool for understanding the meeting content and human interactions. They also agreed that being able to browse the interaction status was useful. Several participants indicated that the images of the screen video were extremely useful for comprehending topics. Most of the participants said that the zoom-in and zoom-out functions helped them master human interactions quickly.

One important note about the user interface is that the participants had mixed feelings about the language displayed on the interfaces. One subject complained that the characters on the interface were not from her mother tongue, which interfered with her usage and caused a negative experience. In addition, regarding the interface, one subject said that she would do better if given more time for training on how to use it.

Despite these issues, all the participants appreciated the features of the visualization tool, and would use it again if they had to review a meeting.



Questionnaire:

(a) I was able to understand the meeting topics in short time using the meeting browser.
(b) The browsing tool was very effective for understanding group social dynamics.
(c) Browsing the interaction status was useful.
(d) Using the tool was easy and convenient.
(e) I would use this meeting browser again if I had to review a meeting.

Figure 6. User study results. All questions were answered using the following scale: 5 = strongly agree, 4 = agree, 3 = neutral, 2 = disagree, 1 = strongly disagree.

## IX. Discussion

### A. Sensor Deployment

Our current system employs cameras, microphones and motion sensors for interaction capture and detection in meetings. Wearing many sensors like microphone and LED tags might be intrusive and reduce usability. But using wearable microphone is useful to capture high-quality audio data that is important for later speech processing. Actually, capturing audio in a meeting room is challenging as it needs to remove a variety of noises and reverberation. The cameras could also be used to detect human face orientation and head gesture by adopting computer vision or image processing techniques. However, this approach is sensitive to human and environment (e.g., sudden illumination change). Recognizing human faces through image processing in meeting room is difficult because of low quality of input images, poor illumination, unrestricted head poses and continuously changing facial expressions and occlusion [42]. Instead the optical motion capture system we used is robust for head tracking.

Although the system achieved an accuracy rate around 75% in interaction recognition, it was still in an early stage and the provided approaches were not well matured. Further improvements and tunings are required to improve the recognition accuracy and also make the system unintrusive and applicable in real-life settings.

### B. Security and Privacy

An important issue we have not involved is security and privacy. Information shared during meetings is highly sensitive [43]. Interaction and information exchange between entities must be secure and private. Security includes three main properties: confidentiality, integrity, and availability [44]. For instance, some internal meetings in a company involve business secrets. It is necessary to protect the content from unauthorized access or restrict access to portions of some data. Privacy is the personal claim of attendees for when, how, and to what extent information is recorded. In fact, users are often anxious about themselves when they are in an environment with many sensors, because they do not know what will be captured and stored, and how it will be used. Privacy mechanisms in ubiquitous computing such as [45] could be explored and used in smart meeting systems.

### C. Applications

The interaction recognition result is expected to be of particular use in some applications such as activating discussion and facilitating decision making. For example, knowing the current interaction status of the meeting, the organizer can make some adjustments to activate discussion (e.g., encourage speaking and suggest related interesting topics for discussion). From the viewpoint of decision making, an application can infer common interest and conflict from user interactions. We can use such social context to help making rational decision and avoid two common problems in discussion, i.e., group think and group polarization [46].

Another potential application is to use the result for indexing meeting semantics. For instance, with the interaction recognition result, it is possible to make index about “the persons who proposed a lot of ideas”, “the persons who were critical”, “the topics that were common interest”, “the topics that was not agreed by all the members”, etc. Previous meeting capture systems could use this as a smarter indexing tool to access different semantics of the meetings.

## X. Conclusion

Exploring how people interact can be used to enhance a variety of pervasive systems. This paper presents our efforts in capturing, recognizing, and visualizing human semantic interactions in a smart meeting. The initial evaluation results show that the approach provides appropriate support for human interaction detection and understanding. In the future, we plan to integrate more contexts (e.g., lexical cues) into the detection process to improve the recognition accuracy. We also plan to conduct knowledge discovery based on the existing results by applying data mining methods.

## Acknowledgments

This work was partially supported by the National Natural Science Foundation of China (No. 60903125), the High-Tech Program of China (863) (No. 2009AA011903), the Program for New Century Excellent Talents in University, and the Doctorate Foundation of Northwestern Polytechnical University of China (No. CX200814).

# REFERENCES

[1] Yu, Z. and Nakamura, Y. Smart Meeting Systems: A Survey of State-of-the-Art and Open Issues. *ACM Computing Surveys*, 42(2), 2010.

[2] Dey, A. K., Salber, D., Abowd, G. D., and Futakawa, M. The Conference Assistant: Combining Context-Awareness with Wearable Computing, In *Proc. ISWC'99*, 21-28.

[3] Sumi, Y. and Mase, K. Digital Assistant for Supporting Conference Participants: An Attempt to Combine Mobile, Ubiquitous and Web Computing. In *Proc. Ubicomp 2001*, 156-175.

[4] Chen, H., Finin, T., and Joshi, A. A Context Broker for Building Smart Meeting Rooms. In *Proc. of the Knowledge Representation and Ontology for autonomous systems symposium* (AAAI spring symposium), AAAI, 2004, 53-60.

[5] Koike, H., Nagashima, S., Nakanishi, Y., and Sato, Y. EnhancedTable: Supporting a Small Meeting in Ubiquitous and Augmented Environment, In *Proc. PCM 2004*, 97-104.

[6] Chiu, P., Kapuskar, A., Reitmeier, S., and Wilcox, L. Room with a Rear View: Meeting Capture in a Multimedia Conference Room, *IEEE Multimedia*, 7(4), 2000, 48-54.

[7] Richter, H., Abowd, G. D., Geyer W., Fuchs, L., Daijavad, S., and Poltrock, S. Integrating Meeting Capture within a Collaborative Team Environment. In *Proc. Ubicomp 2001*, 123-138.

[8] Ark, W. S. and Selker, T. A look at human interaction with pervasive computers. *IBM Systems Journal*, 38(4), 1999, 504-507

[9] Vapnik, V. N. The Nature of Statistical Learning Theory. Springer Verlag, Heidelberg, DE, 1995.

[10] Ahmed, S., Sharmin, M., and Ahmed, S. I. A Smart Meeting Room with Pervasive Computing Technologies. In *Proc. SNPD/SAWN'05*, IEEE Computer Society Press (2005), 366-371.

[11] Mikic, I., Huang, K., and Trivedi, M. Activity Monitoring and Summarization for an Intelligent Meeting Room. *IEEE Workshop on Human Motion*, 2000, 107-112.

[12] Kim, N., Han, S., and Kim, J. W. Design of Software Architecture for Smart Meeting Space. In *Proc. PerCom 2008*, IEEE Press (2008), 543-547.

[13] Stiefelhagen, R., Chen, X., and Yang, J., Capturing Interactions in Meetings with Omnidirectional Cameras. *International Journal of Distance Education Technologies*, 3(3), 2005, 34-47.

[14] Nijholt, A., Rienks, R. J., Zwiers, J., and Reidsma, D. Online and Off-line Visualization of Meeting Information and Meeting Support. *The Visual Computer*, 22(12), 2006, 965-976.

[15] Kim, T., Chang, A., Holland, L., and Pentland, A. Meeting Mediator: Enhancing Group Collaboration using Sociometric Feedback. In *Proc. CSCW 2008*, 457-466.

[16] Sumi, Y., et al. Collaborative capturing, interpreting, and sharing of experiences. *Personal and Ubiquitous Computing*, 11(4), 2007, 265-271.

[17] DiMicco, J. M., et al.: The Impact of Increased Awareness While Face-to-Face. Human-Computer Interaction, 22(1), 47-96 (2007)

[18] Otsuka, K., Sawada, H., and Yamato, J. Automatic Inference of Cross-modal Nonverbal Interactions in Multiparty Conversations. In *Proc. ICMI 2007*, 255-262.

[19] Dielmann, A. and Renals, S. Dynamic Bayesian Networks for Meeting Structuring. In *Proc. ICASSP 2004*, 629-632.

[20] McCowan, I., Gatica-Perez, D., Bengio, S., Lathoud, G., Barnard, M., and Zhang, D. Automatic Analysis of Multimodal Group Actions in Meetings. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3), 2005, 305-317.

[21] Rybski, P. E. and Veloso, M. M. Using Sparse Visual Data to Model Human Activities in Meetings. In *Proc. Of IJCAI Workshop on Modeling Other Agents from Observations (MOO 2004).*

[22] Hillard, D., Ostendorf, M., and Shriberg, E. Detection of Agreement vs. Disagreement in Meetings: Training with Unlabeled Data. In *Proc. HLT-NAACL 2003*, 34-36.

[23] Tomobe, H. and Nagao, K. Discussion Ontology: Knowledge Discovery from Human Activities in Meetings. In *Proc. JSAI 2006*, 33-41.

[24] Garg, N. P., Favre, S., Salamin, H., Tur, D. H., and Vinciarelli, A. Role Recognition for Meeting Participants: an Approach Based on Lexical Information and Social Network Analysis. In *Proc. ACM Multimedia 2008*, 693-696.

[25] Searle, J. Speech Acts, Cambridge University Press, 1969.

[26] Waibel, A., Bett, M., and Finke, M. Meeting Browser: Tracking and Summarizing Meetings. *Proc. of the Broadcast News Transcription and Understanding Workshop*, Lansdowne, Virginia, February 1998, 281-286.

[27] Colbath, S. and Kubala, F. Rough'n'Ready: A Meeting Recorder and Browser. In *Proc. of the Perceptual User Interface Conference*, San Francisco, CA, November 4-6, 1998, 220-223.

[28] Wellner, P., Flynn, M., and Guillemot, M. Browsing Recorded Meetings with Ferret. *Proc. of the First International Workshop on Machine Learning for Multimodal Interaction (MLMI'04)*, Martigny, Switzerland, June 21-23, 2004, 12-21.

[29] Geyer W., Richter, H., and Abowd, G. D. Towards a Smarter Meeting Record – Capture and Access of Meetings Revisited. *Multimedia Tools and Applications*, 27(3), 2005, 393-410.

[30] Jaimes, A., Omura, K., Nagamine, T., and Hirata, K. Memory Cues for Meeting Video Retrieval. *The first ACM Workshop on Continuous Archival and Retrieval of Personal Experiences (CARPE'04)*, New York, NY, USA, October 15, 2004, 74-85.

[31] Junuzovic, S., Hegde, R., Zhang, Z., Chou, P. A., Liu, Z., and Zhang, C. Requirements and Recommendations for an Enhanced Meeting Viewing Experience. In *Proc. of ACM Multimedia 2008*, 539-548.

[32] Araki, M., Itoh, T., Kumagai, T., and Ishizaki, M. Proposal of a standard utteranceunit tagging scheme. *Journal of Japanese Society for artificial intelligence*, 14(2), 1999, 251-260.

[33] PhaseSpace IMPULSE system.
http://www.phasespace.com/.

[34] Julius speech recognition engine. http://julius.sourceforge.jp/en/.

[35] Rabiner, L. A tutorial on Hidden Markov Models and selected applications in speech recognition. In *Proc. IEEE*, 77(2), 1989, 257-286.

[36] Kipp, M. Anvil - A Generic Annotation Tool for Multimodal Dialogue. In *Proc. Eurospeech 2001*, 1367-1370.

[37] Hsu, C. W., Chang, C. C., and Lin, C. J. A practical guide to support vector classification. *Technical Report*, 2005.

[38] Chang, C. C., and Lin, C. J. LIBSVM: a library for support vector machines, 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm

[39] Hackman, J. R. Groups That Work (And Those That Don't). Jossey-Bass, San Francisco, 1990.

[40] Pianesi, F., Zancanaro, M., Not, E., Leonardi, C., Falcon, V., and Lepri, B. Multimodal support to group dynamics. *Personal and Ubiquitous Computing*, 12(3), 2008, 181-195

[41] Yu, H., Finke, M., and Waibel, A. Progress in Automatic Meeting Transcription. *Proc. of 6th European Conference on Speech Communication and Technology (Eurospeech-99)*, September 5-9, 1999, Budapest, Hungary, Vol. 2, 695-698.

[42] Gross, R., Yang, J., and Waibel, A. Face Recognition in a Meeting Room. In *Proc. of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition*, 2000, 294-299.

[43] Jaimes, A. and Miyazaki, J. Building a Smart Meeting Room: From Infrastructure to the Video Gap (Research and Open Issues), *the 21st International Conference on Data Engineering Workshops (ICDEW 05)*, Tokyo, Japan, April 5-8, 2005, 1173-1182.

[44] Stajano, F. Security for Ubiquitous Computing, Wiley, 2002.

[45] Moncrieff, S., Venkatesh, S., and West, G. Privacy and the access of information in a smart house environment. In *Proc. of ACM Multimedia*, 2007, 671-680.

[46] Brown, R. Group Polarization in Social Psychology, 2nd ed., Free Press, 1986.