Karlsruhe Institute of Technology

Fraunhofer IOSB

# High-Level Activity Recognition through Interaction Modeling and Graph Analysis

Diplomarbeit
submitted by

## Sebastian Kaltwang

at the Department of Computer Science
of the Karlsruhe Institute of Technology

Supervisors:

Prof. Dr.-Ing. Rainer Stiefelhagen

Prof. Dr.-Ing. Jürgen Beyerer

Joris Ijsselmuiden, M.Sc.

January 31, 2011

Karlsruhe Institute of Technologie (KIT)
Department of Computer Science
Institute for Anthropomatics

Topic:
High-Level activity recognition through interaction modeling and graph analysis

Author:
Cand. Dipl.-Inform. Sebastian Kaltwang

Sebastian Kaltwang
Feldkreuzstrasse 21
66359 Bous
s.kaltwang@googlemail.com

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig verfasst habe und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Karlsruhe, 31. Januar 2011

# Abstract

Human activity analysis is an important computing task that is needed for many applications, like automatic report generation, surveillance, and human-computer interaction. The goal of this study is (1) to group persons and objects together that interact with each other and (2) to detect the appropriate high-level activity for each group. To achieve these goals, a two layer system is proposed. The first layer detects interactions between each person-person pair and each person-object pair. This is achieved by integrating the visual focus of attention target of each person over time. The result is a fully connected weighted graph, in which the nodes represent persons and objects and the edge weights represent interaction scores. The second layer calculates a score for a number of predefined template graphs to determine which one fits the fully connected weighted graph best. The template graphs represent group-activities such as 'meeting', 'presentation', and 'using an object together'. Experiments were conducted using a database of dynamic scenarios in a meeting room. Different temporal window sizes and parameter settings have been tested. The results showed that our system was able to recognize the correct group clustering in 75% of all frames and the correct activities in 59% of all frames.

# Contents

# 1 Introduction

Since the invention of the computer, human computer interaction (HCI) methods are still in a development process toward more sophisticated methods: starting with text-based interfaces, shifting to graphical interfaces with mouse, the latest technologies use touch-pads, gesture recognition or even movements of all body parts as input (e.g. Microsoft Kinect). However, the basic principle has remained the same: the system waits for commands from the user, it is a *reactive* system.

In the recent years, HCI research tries to overcome that principle. The new goal is to build systems that can choose appropriate actions on their own or with as little user commands as possible: *proactive* systems. The system should behave like a servant that understands what his master does or wants to do and therefore is able to help without explicit commands. And this 'understanding' gathered much attention in the actual research: automatic recognition of human behavior. First, the system needs to understand what the user does and depending on that it can choose an appropriate action.

One form of these new systems are 'smart environments': rooms or areas equipped with sensors to detect human behavior and a back-end system for interpreting the sensor data and interacting with the users through connected electronic devices (like displays, machines, robots, etc.). There are many possible applications for these smart environments: assisted living, smart homes, smart offices, surveillance systems and more.

We now focus on one of these applications: the smart office. Its goal is to support people in an office environment, and therefore it must be able to recognize typical office situations. There exist already some approaches to achieve that, a selection of them is introduced in chapter 2.2. Most of these systems are able to detect basic actions (like 'person speaking', 'person entering the room', etc.) and higher-level activities (like 'discussion', 'presentation', etc.). However, to the best of our knowledge, there exists no system that is able to detect several concurrent high-level activities. Furthermore, it is not possible for actual systems to detect a subset of persons participating in a detected high-level activity: usually it is assumed that all persons in the whole area or room take part in the activity. However, this is not the case in general and therefore a system is needed that provides more detailed information about who is working with whom.

The focus of this work is to build a recognition system, that is able to segment persons in an office environment into activity clusters and to detect for each cluster what activity its members are performing. We detect for each time frame the cluster configuration and the corresponding activities. The system design allows the usage of any relevant data for the activity recognition (like tracking information, detected speech and gesture recognition). However, the actual implementation of this work uses only the visual focus of attention. The system is built and evaluated with a data set recorded in the Interactive Systems Lab at the Karlsruhe Institute of Technology. A detailed description of the data can be found in [16].

Next in chapter 2, some information is given about interaction modeling in general, together with an overview of the related work. Then an in-depth description of the problem is given in chapter 3. After that, we explain the composition of the data which is used for evaluation and how this data is generated in chapter 4. We propose an activity model based on dual interactions together with a two layer system to solve the recognition task in chapter 5. Then an evaluation of this system is conducted in chapter 6. And finally we draw some conclusions about the results, show the limitations of the system and give a perspective for the future in chapter 7.1.

# 2
# Background

The goal of this work is to recognize human activities. This chapter gives an overview what kind of activities exist in section 2.1 and then introduces several actual research projects related to activity recognition in section 2.2.

## 2.1 Terminology

Human activity recognition is the automatic detection of human activities from image sequences (and/or sequences from other sensors like microphones or motion detectors)[1]. The general question to answer is: Who does what and when? We can differentiate the recognition tasks that can be performed on an image sequence regarding their complexity:

- detect exactly one activity (no segmentation of the sequence)
- detect different consecutive activities (with time segmentation of the sequence)
- detect different concurrent and consecutive activities (with space and time segmentation of the sequence)

In this work, the word 'activity' is used for anything humans are able to do, from simple movements of limbs to complex activities involving several persons (like 'building a house'). We differentiate between activities according to a scheme that has been proposed in [1]. The activities are separated according their complexity:

1. *gestures:* elementary movements of body parts; atomic activities (e.g. 'moving arm', 'closing hand')

[1] In the following we speak mostly about vision data (because it is the modality that contains the most information and this work is done in the research group 'Computer Vision for Human-Computer Interaction'). However, the presented methods can also be applied to other modalities and multi-modal data.

2. *actions:* single person activities, they are composed of gestures (e.g. 'walking', 'swimming')
3. *interactions:* activities that involve at least two persons and/or objects (e.g. 'talking to', 'fighting')
4. *group-activities:* activities performed by conceptual groups of multiple persons and/or objects (e.g. 'holding a presentation', 'two groups fighting')

The boundary between interactions and group-activities is not exactly defined. Therefore we augment the definition for group-activities, so that there has to be a group of people performing the same activity. According to this definition, 'presentation' is a group-activity, because a group of people (the audience) is listening to a single person (the presenter).

## 2.2 Related Work

In this overview, we use another categorization scheme that is proposed in [1]. Activity recognition systems are divided into two main classes: single-layer and multi-layer systems. Single-layer systems detect activities directly from image sequences, where multi-layer systems are using intermediate steps. Single-layer systems are good for recognizing gestures and actions, whereas multi-layer systems are better suited for recognition of interactions and group-activities (see section 2.1). This is because the multi-layer system structure can resemble the composition of the gestures and activities involved. For example, actions could be recognized in the first layer and then the second layer detects group-activities based on these actions. The division into layers reduces the complexity of the system and therefore less training data is needed and the execution-time is lower. This work focuses on higher-level activity recognition and thus multi-layer systems are the first choice. In a multi-layer system, the first layer has to process image sequences and the possibilities on what algorithm to use are the same as the ones for single-layer systems. Therefore some single-layer systems are introduced here as well.

### 2.2.1 Single-layer systems

Single-layer systems recognize activities from image sequences without intermediate processing steps. A distinct activity is seen as a class of image sequences and the recognition task of the system is to decide if an unseen sequence belongs to that class or not. The start and end points of the activities are usually not known and therefore most of the systems use a window of fixed length and classify all possible sub-sequences of that length. These approaches are able to learn typical movement patterns and are therefore best suited for recognizing short or recurring activities like 'running' or 'swimming'.

There are many possible ways to build a single-layer system. We can differentiate between two categories: space-time approaches and sequential approaches. The first category does not treat the video frames separately. They are concatenated along the time axis and features are extracted from the hole sequence together. On the other hand, sequential approaches extract features separately from every image (or from subsequences of fixed size).

Space-time approaches use all the pixel data as a whole (as 3-dimensional data set: x-, y- and time-dimension) for the classification process or they deduce features from the video data first (e.g. trajectories of tracking points). The classification can then be done by template matching, neighbor-based classifiers or statistical models. For example, in [13] they use the trajectories of several tracking points of human body parts to detect the actions 'sitting', 'standing', 'running' and 'walking'. The classifier is constructed by learning from known sequences of each class. Then the classification of an unknown sequence is done by calculating the likelihood of the membership for each class.

Sequential approaches calculate feature sequences first, and then these are categorized by comparison with template sequences or evaluation of state-models (for every activity class). In [8] they use a sequential approach to detect human interactions in a public pedestrian area by using hidden Markov models (HMMs). A typical interaction to detect is for example: a person that changes his walking direction to meet someone, then talks with him and after that leaves together with him.

### 2.2.2 Multi-layer systems

Multi-layer systems recognize high-level activities by first detecting simpler activities (also called sub-activities or sub-events). Then the high-level activity is deduced from the set of detected basic activities. For example, the group-activity 'presentation' could be recognized by first detecting 'listening' for the people in the audience and 'speaking' for the presenter. Single-layer systems (see section 2.2.1) can be used to detect the basic activities. In general, single-layer systems could also be used to detect complex activities, but the multi-layer system have several advantages:

- *Reduced Complexity:* By dividing the task into the recognition of sub-activities, the complexity is reduced. For example, when using a single-layer HMM: If the complexity of the activity increases, the number of hidden states must be increased. Therefore the number of transition and output probabilities increases. Due to the higher complexity of these probabilities, more training data is needed to learn them and more time is needed for the training and classification calculations than with several smaller HMMs of a multi-layer system.

- *Possibility to include expert knowledge:* If the composition of an activity into sub-activities is known, it can be easily included in the system because of its hierarchical structure.

- *Less Redundancy:* Sub-activities can be re-used for several higher-level activities, which is also decreasing the need for training data.

Within multi-layer systems, we differentiate between statistical, syntactic and description-based approaches. Statistical approaches use probabilistic state-space models to recognize human activities. The most common ones are HMMs and the more general dynamic Bayesian nets (DBNs). In multi-layer systems, training and classification is done separately for each layer. The first layer transforms a feature sequence into a sequence of basic activities. Then the second layer takes them as input (for training and classification) and recognizes the target high-level activities. Two layer systems are the most common ones, but using any number of layers is possible. An example for a statistical approach is the system proposed in [18]. They use multi-layer HMMs to detect activities and track persons in an office environment that consists of several rooms. The goal is to recognize for every room what is happening ('nobody in the office', 'paperwork', 'discussion' or 'meeting') and to detect for every person where (s)he is. Every room is equipped with a camera and a microphone. In the first layer, separate video and audio HMMs are trained for each room. The video HMMs detect tracking events (e.g. 'somebody at user's desk', 'somebody enters') and the audio HMMs detect if there is a conversation or not. At the second layer, two tasks are performed independently from each other: activity recognition and room level tracking. Activity recognition is done by training a set of HMMs *for each room* and room level tracking is done by training a set of HMMs *for each person*. Both sets of HMMs use the results from video and audio HMMs of the first layer as input vector. The final detected result is the activity going on in *each room* and the location of *each person*. Another statistical approach with Markov-switching models is proposed in [9], their goal is to detect gaze patterns in meetings. In [3], they have built a DBN system that is able to detect high-level group-activities in meetings. Also a similar goal with layered HMMs is targeted in [19].

Syntactic approaches model activities as sequences of symbols defined by a formal language's alphabet. Each symbol stands for a (sub-)activity and the language's grammar defines production rules on how higher-level activities are composed of lower-level activities. An example of a syntactical approach can be found in [6], where stochastic context-free grammars are used.

Description-based approaches use spatio-temporal reasoning to recognize human activities. Higher-level activities are described by using lower-level activities and the spatial, temporal and logical relations between them. After deriving the basic activities and their time intervals in the first layer, the high-level activities can be deduced by a logic engine. An example is the system described in [5]. As experimental setup, they use an emergency response control room equipped with several cameras, close-talking microphones and a videowall. The goal is to recognize high-level activities like 'group meeting' and 'people working together at the videowall'. The system uses not the raw video and audio data as input, but already processed data streams from other systems (tracks and identities, visual focus of attention, gestures and speech of the people in the room). As first step, a situation model is build that keeps all basic information about persons and objects in the room. The model is updated each time step by events (e.g. 'person X entering the room'), which are deduced from the input data. Atomic predicates and their temporal relations to high-level activities are predefined by implying expert-knowledge. The truth state of the atomic predicates (e.g. 'person X is close to person Y at time T') can directly be derived from the situation model. Finally, the high-level activities are recognized from atomic predicates by temporal logic.

# 3 Problem Description

In the recent years, much effort has been spent in building activity recognition systems that detect *what* is going on (see section 2.2). However, most of the systems do not detect *who* is together in a group or performing a group-activity. Usually they assume that in one room there is exactly one situation and everybody in there is participating in the same group-activity (like in [18], [3], [7] or [9]). Only a few systems include the ability to separate all persons into groups, e.g. [5] and [11]. The goal of this work is to build a combined group-activity recognition system that detects who is together in a group and what activity each group is performing (including activities that involve using of objects). Therefore the following information must be detected for each point in time:

- which of the currently present persons are together in a group
- what activity is each group performing
- what objects are being used and by whom

The activities to detect are typical office situations. An example classification for one point in time is shown in figure 3.1. These are all activities that should be detected:

1. *no interaction* (single person activity)
   Someone is not interacting with any person or object.
2. *discussion* (group-activity)
   Several people are talking to each other. The speaking person changes over time.
3. *monologue* (group-activity)
   Someone is giving a speech for an audience.
4. *single object interaction* (single person activity)
   One person is using an object.

5. *cooperative object interaction* (group-activity)
   Several persons are using an object together, but they are also talking to each other (e.g.: some people discussing a slide on the projection screen).

6. *separate object interaction* (group-activity)
   Several persons are using an object, but they are not interacting with each other (e.g.: some people are watching a movie).

7. *presentation* (group-activity)
   Someone is giving a speech for an audience about an object (e.g.: someone is giving a presentation by using slides on the projection screen).

These activity definitions are not exact. For example, the boundary between monologue and discussion is not well defined: discussion means that several people alternately speak, but this is the same as holding consecutive monologues. The common understanding is, that in a discussion the speaker should change within a short time, whereas a monologue should last for more time. However, there is no exact time boundary defined. The decision during ground-truth annotation is left to the annotator.

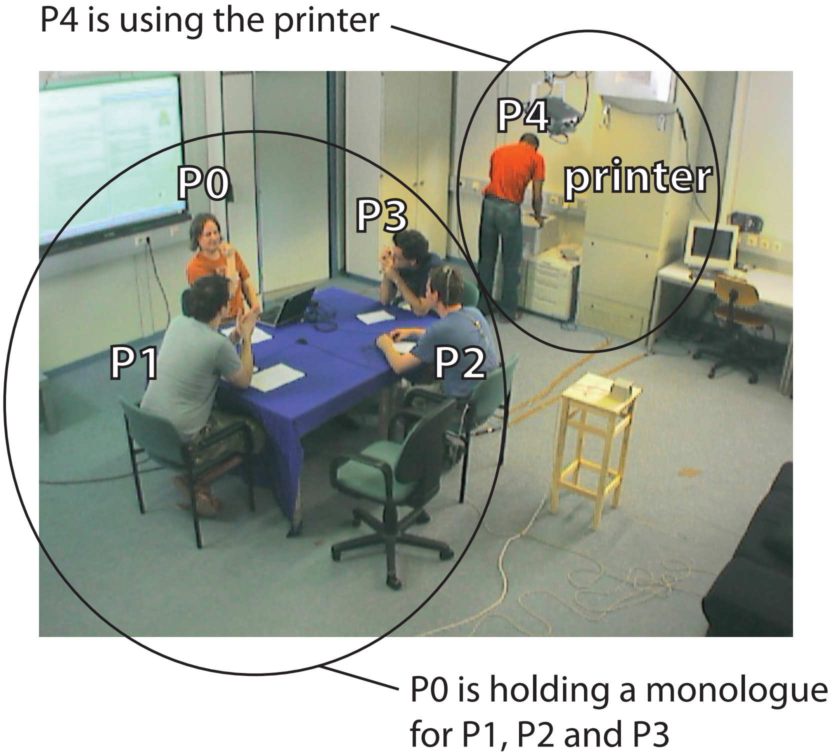


Figure 3.1: **Example activity recognition for one point in time.** The fact that the shown group-activity is 'monologue of P0' cannot be recognized from this image alone, because it is not known if P0 was speaking for a longer time and P1 is looking at P4 at this moment. To unambiguously determine the activity, the video of this situation must be watched (including sound).

# 4

# Input Data

This work focuses on the SmartOffice data [16]. Training and evaluation were conducted on that data. However, an interface for the SmartControlRoom data [5] has been implemented as well and integrating the support for other input data can be done without changing the system itself (see section 5). In the following, we provide a description of both setups.

## 4.1 SmartOffice

The SmartOffice data has been recorded in a meeting room at the Interactive Systems Laboratory of Karlsruhe Institute of Technology. An overview of the setup is shown in figure 4.1. The data consists of 10 videos that show dynamic meeting scenes, each one about 10 minutes long. Each video shows 3-4 persons having a meeting and one additional person causing disruptive events. The disrupting person and 1-2 of the others are actors that trigger predefined events according to a plot. The remaining 1-2 persons are not clued-in about the plot, thus we can assume that their behavior is spontaneous. The activities during the meeting include:

- entering the room (each person separately)
- greeting each other
- giving a presentation
- holding a monologue
- having a discussion
- writing notes

One of the actors causes events that intend to disrupt the other people, like:

- joining an ongoing discussion

- looking for a ringing cell phone
- playing loud noise from speakers
- using the printer

All meeting videos have been recorded by 4 cameras in the upper corners of the room and 1 fish eye camera at the center of the ceiling. An image from one of the cameras with a typical situation is shown in figure 3.1. The videos have a resolution of 640x480 pixel and the frame rate is 15 fps. The audio sequences have been recorded by 4 microphone arrays located at the center of each wall (for audio source localization) and 1 table-top microphone (for speech detection).

The raw video and audio data has been annotated and processed by the Interactive Systems Laboratories research group (see also [16]), hence the following information is available for every time frame and every person: identity and position (also for all objects), speech activity, head pose angle and visual focus target. The visual focus target and the head bounding box (including identity) have been independently annotated by several students. 35 objects in the room were possible visual focus targets and their position is also included in the data. The position of the persons is calculated by intersecting the centers of the head bounding boxes of the different camera views. The speech activity is calculated by matching the audio source locations with the positions of the people. The head pose angle is estimated from head images with neuronal nets, the method is explained in [16]. The current implementation of the system uses *only* the visual focus of attention target as input data, but using other information would be possible as well (see sections 5.2 and 7.3).

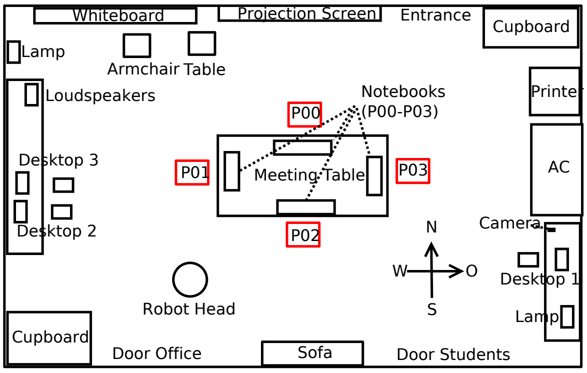


Figure 4.1: Overview of the SmartOffice (source: [16])

## 4.2 SmartControlRoom

The SmartControlRoom is a computer vision laboratory built by the research group *Perceptual User Interfaces* at Fraunhofer IOSB in Karlsruhe[1]. The main goal of this research group is to develop advanced human computer interaction methods for a smart environment. The concrete scenario of an emergency response control room is replicated in the laboratory. The main interaction component is a videowall of the dimension 4.0

[1] Web page: `http://www.iosb.fraunhofer.de/servlet/is/20718/`

x 1.5 meter and a resolution of 4096 x 1536 pixel. The low-level perception components include:

- four cameras in the upper corners of the room
- one fish-eye camera at the center of the ceiling
- four cameras looking down from the ceiling at the area in front of the videowall
- two active pan-tilt-zoom cameras on the walls at head height
- four wireless close talking microphones

All cameras have a resolution of 640 x 480 pixel and a frame rate of 30 fps. Eight standard PCs are available for the system software and all components are connected by a LAN middleware. Four low-level recognition modules have been build: 'Tracks and Identities', 'Visual Focus of Attention', 'Gestures' and 'Speech'. Each of these recognition modules uses a subset of the aforementioned perception components. Person tracking is done by particle filtering [2] and face identification uses a DCT-based local appearance model [4]. The visual focus of attention detection for the persons in the room is performed by an voxel-based approach that uses a Bayesian Surprise framework [17]. Gestures for interaction with the videowall are detected through voxel carving [12]. Speech recognition is done by a system described in [14]. Although only the visual focus of attention is used in the current implementation (see section 5), the long-term goal is to use all four of these recognition modules (see section 7.3).

# 5 System Description

Before starting a description of the system itself in sections 5.2 and 5.3, we first analyze the goal that should be achieved, explain how the problem is modeled and what we define as interaction in section 5.1.

## 5.1 Activity Model

As explained in section 3, the goal is to detect several (group-)activities in dynamic meeting scenarios. These activities are composed of interactions between the participants and/or objects (see section 2.1). The idea behind the system is to detect dual interactions first (i.e. interactions between two persons or one person and one object) and then deduce activities based on these interactions. We choose *dual* interactions because single actions of one person do not suffice for detecting groups. For example, if we only know that someone is performing the action 'speaking', then we cannot recognize the group he belongs to because we obviously need to know who he is speaking to. Therefore, we need to observe at least dual interactions (e.g. 'person A is speaking to person B'). It would be possible to recognize interactions that involve more than two people in the bottom layer, but we choose dual interactions to keep the system simple.

Each of the activities to detect is composed of several interactions. However, there is not one exactly defined composition but a wide range of possibilities. We explain this with the help of the example group-activity 'monologue'. As rough description we can say, that the activity 'monologue' consists of one person giving a speech for an audience. That means that at least two people are involved (we exclude the self-monologue). As concrete example, we now assume that person P0 is holding a monologue for persons P1, P2 and P3. The performed interactions are 'P0 speaking to P1, P2, P3' and 'P1, P2, P3 listening to P0'. We can express the same as dual interactions: 'P0 speaking to P1' and 'P0 speaking to P2' and 'P0 speaking to P3' (and respectively the same for 'listening'),

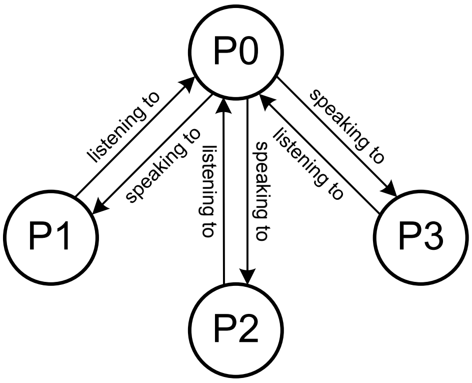


Figure 5.1: Monologue composition into sub-activities

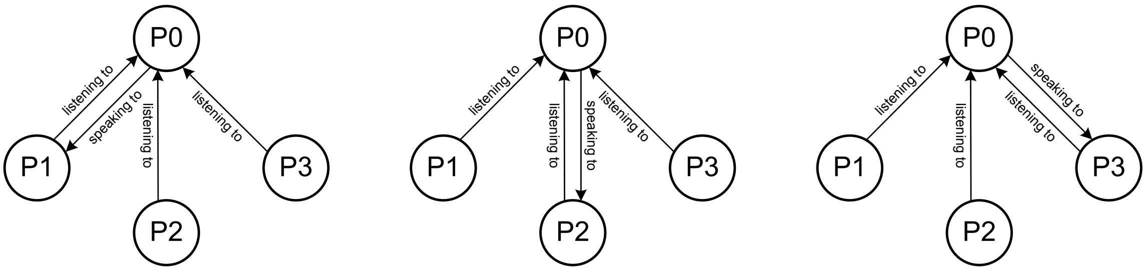


Figure 5.2: Different monologue compositions into sub-activities

see figure 5.1. Next, we need to define the meaning of 'A speaking to B'. As rough description, it means that 'A is speaking' and 'A is looking at B' and 'B is not speaking' and 'B is looking at A'. At this point, we need to include the time dimension, because it is not possible that P0 looks at P1, P2 and P3 at the same time. Therefore we revise the presumption that P0 is speaking to all others and assume that P0 is speaking to one member of the audience, say P2. However, any sequence of P0 speaking to different members of the audience would still compose a monologue, see figure 5.2. There also exists a problem with the interaction 'looking at' in general, because it will not hold all the time during the monologue. Usually humans look around during a conversation (because they get distracted, bored or simply the eyes get tired). For example, in figure 3.1, P1 gets distracted by P4. However, he was focusing on the speaker most of the recent time. According to [15], if someone is listening, then there is a chance of 88% that he is actually looking at the speaker. Therefore any sequence of 'P1 looking at X' would fit for the monologue, as long as X is sufficiently often P0. We have to note here, that in this work it is *not* defined how often the focus should be on the speaker. The ground-truth for evaluation purposes is defined by manual annotations and the (human) annotator had to decide intuitively if it is a monologue or not. Also, people could perform other interactions during listening to the monologue. If P3 takes notes from time to time, then this would not interrupt the activity 'monologue'. However, there is a threshold: if he is spending too much attention to the notebook, then he would not be listening to the monologue anymore. Similar problems as described here for 'monologue' exist for all activities we want to detect: different compositions into sub-activities are possible and other non-related sub-activities occur because people get distracted or do something else at the same time. Therefore the search space for detecting high-level activities from sub-activities by using composition constraints is huge.

For simplifying the problem, our system focuses on one general interaction instead of detecting different kinds of interactions. We call this general interaction 'paying attention to' and it stands for any kind of interaction. Whether you are listening, speaking, looking at, greeting, punching, using or doing anything else with someone/something, you are always paying attention to him/it. The monologue structure then looks like figure 5.3. 'Paying attention to' is not the same as just 'looking at', as described above: humans tend to look around, they do not focus their target of attention all the time. However, we can assume that someone/something is the target, if he/it was focused sufficiently often. For additional simplification, we do not use directed interactions. Which means that 'A is paying attention to B' and 'B is paying attention to A' are both reduced to 'there is attention between A and B', see figure 5.4.

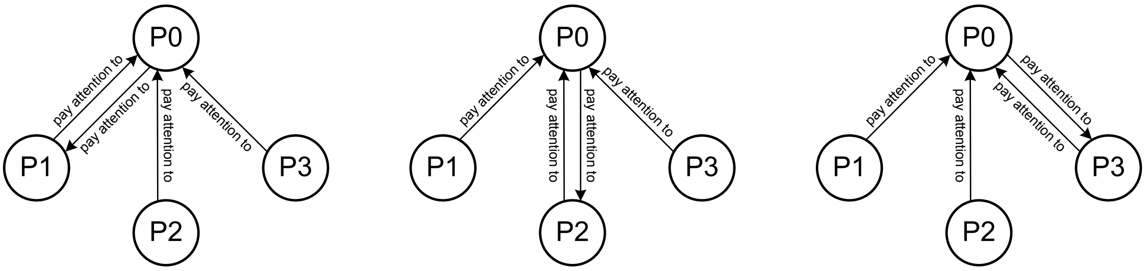


Figure 5.3: Different monologue compositions into 'paying attention'

Obviously, information is lost by using the aforementioned simplifications. However, for clustering persons and objects into groups (the first part of the goal of this work) the lost information is not relevant. We are only interested in who are interacting with each other, the actual type of interaction does not matter. The situation is different for classifying the activity of a group, because the high-level activity depends on what exact interactions have happened. For example, two persons shaking hands (high-level: 'greeting') is different from two persons punching each other (high-level: 'fighting'). In our case however, if more than two people are involved, then we can distinguish between different group-activities on the basis of their interaction structure. The interaction structure for a monologue of 4 persons is shown in figure 5.4. Other group-activities

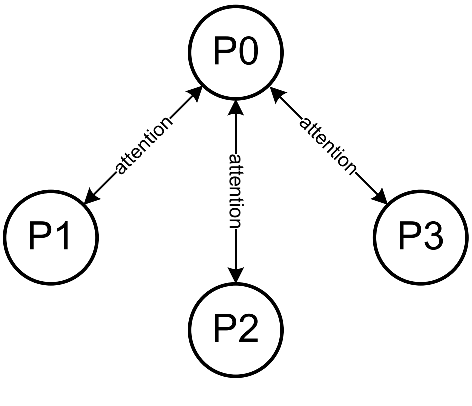


Figure 5.4: Monologue composition into undirected 'paying attention'

have different structures, a full list of all activities we want to recognize is shown in table 5.1. For every group-activity the table shows an example composition into dual interactions, together with the derived structure of the attention graph. For 'monologue' and 'cooperative object interaction' are two example compositions shown, because the speaker changes over time. The possibility is excluded that a person or group is using multiple objects at the same time. In reality this is possible (e.g. one persons watching multiple screens), however we concentrate on the object that gets the most attention.

Each of the group-activities can be performed by any number of persons. The attention graph structures for up to 4 persons is shown in table 5.2. In the following part these graph structures are referred to as *template graphs*, because these are the predefined group-activity graphs that should be recognized. Also the interaction 'attention between' is now referred to as 'interaction', because it is the only interaction that is used. Usually, we distinguish between 5 possible activities for a group, see chapter 3. However, this is only possible, if there are at least 3 person in the group. If there is only one person, then it is a single person activity and group-activities like 'discussion' and 'monologue' make of course no sense. If there are two persons in a group, our system cannot distinguish between 'discussion' and 'monologue'. Usually, a monologue is recognized if all persons can be divided in one speaker and the audience. The audience is interacting with the speaker, but not interacting with each other. In contrast to that, all persons are interacting with each other in a 'discussion'. Respectively the same holds for 'cooperative object interaction' and 'presentation' with 2 persons. In summary that means we can only distinguish 3 cases if 2 persons are in a group (see the row for 2 persons of table 5.2).

Through using the simplifications explained above, the original problem is narrowed down to detecting template graphs. The template graphs consist of dual interactions, therefore we propose a two layer recognition system (see figure 5.5): the first layer (see section 5.2) detects interactions between every person/person and person/object pair and the second layer (see section 5.3) tries to find the best explanation of template graphs for the detected interactions.

## 5.2 Interaction Layer

The goal of this layer is to determine for each person-person and person-object pair whether they are interacting with each other. The result should not be a hard decision, but a probability like result (according to the needs of the group-activity layer, see section 5.3). That means the result should be a number between 0 and 1 with a decision threshold of 0.5 (numbers above 0.5 are assumed to be 'interaction', but the higher the number the more certain is the decision). We first focus on interactions between two persons. In our data set (see section 4), the following information is available: position, head pose, speech, and the visual focus target of each person. In [9] for example, they use speech detection and head pose. We decide to use visual focus of attention instead, because it is quite a reliable source according to [15] (listeners gaze at the speaker with a probability of 88% and speakers gaze to their addressee with a probability of 77%). Of course, other approaches are possible, see also section 7.3.

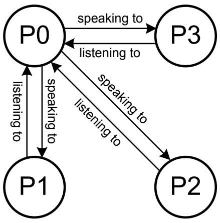
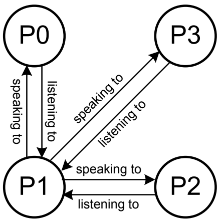
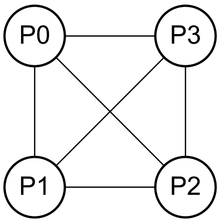
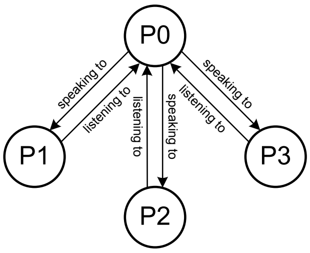
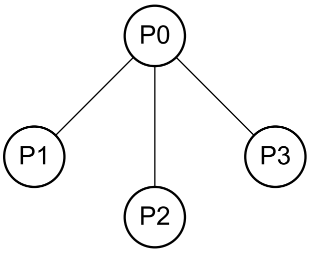
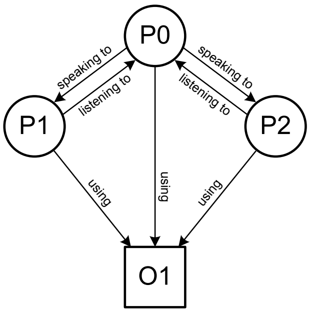
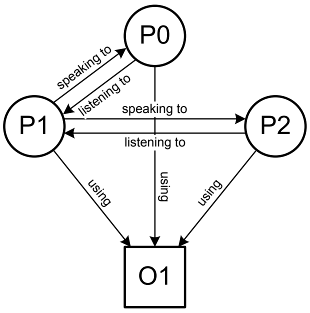
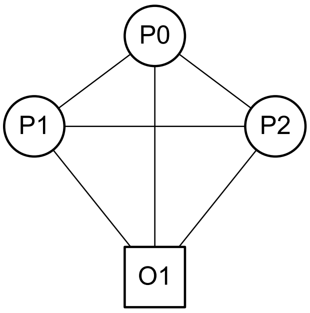
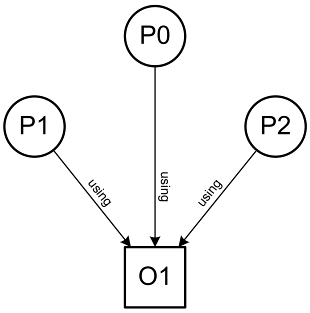
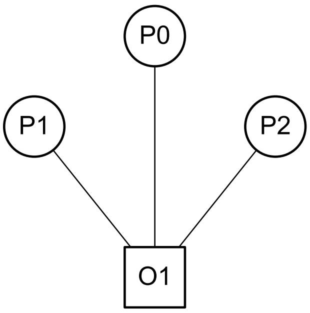
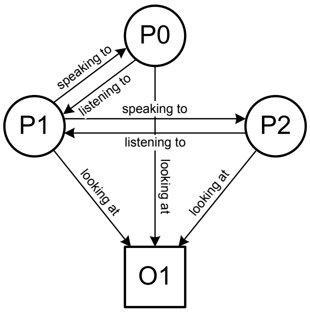
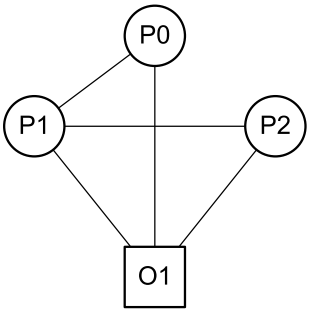
| | Example Composition | | Attention Structure |
|---|---|---|---|
| discussion (speaker changes over time) |  |  |  |
| monologue of P0 |  | |  |
| cooperative object interaction (speaker changes over time) |  P0<br>P1<br>P2<br>O1<br>speaking to<br>listening to<br>using |  P0<br>P1<br>P2<br>O1<br>speaking to<br>listening to<br>using |  |
| separate object interaction |  | |  |
| presentation of P1 |  P0<br>P1<br>P2<br>O1<br>speaking to<br>listening to<br>looking at | |  |

Table 5.1: **Activities and their attention structure.** A description of the activities is provided in chapter 3.

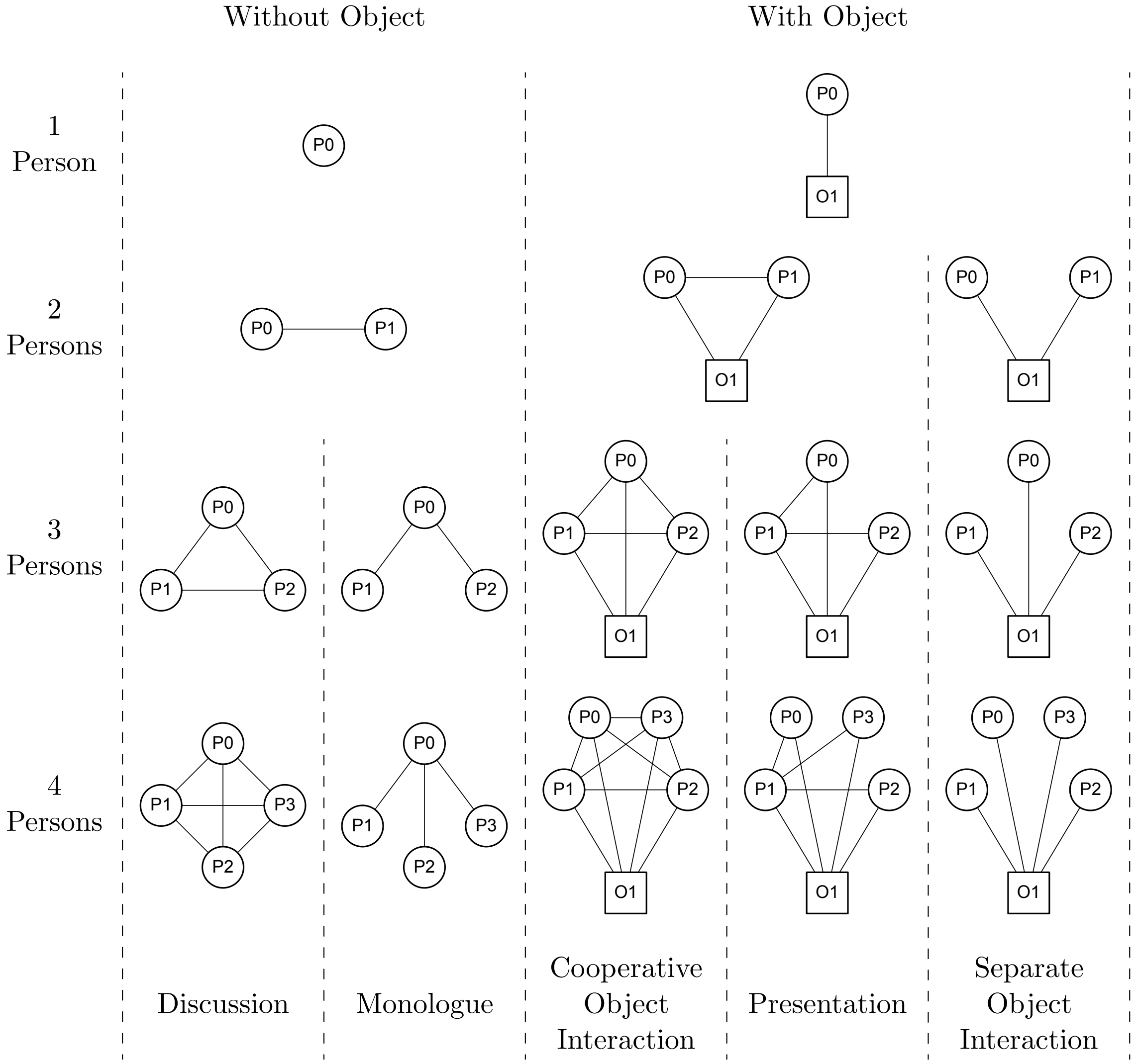


Table 5.2: All template graphs for 1 to 4 persons

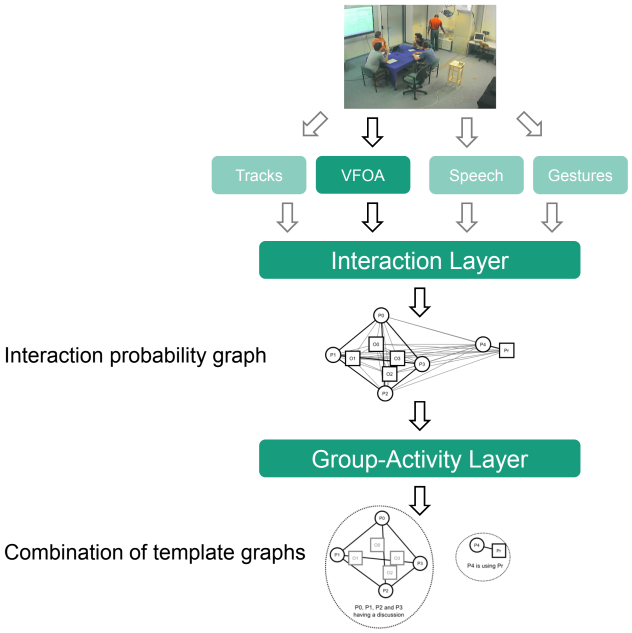

Tracks
VFOA
Speech
Gestures
Interaction Layer
Interaction probability graph
P0
P1
P2
P3
P4
O0
O1
O2
O3
Pr
Group-Activity Layer
Combination of template graphs
P0, P1, P2 and P3 having a discussion
P4 is using Pr

Figure 5.5: Overview of the system

We assume that the visual focus of attention (VFOA) target is provided for each person and each time frame. The goal is to calculate a score between 0 (certain no interaction) and 1 (certain interaction) for the interaction between two persons A and B for each time frame. A simple approach would be to set the score to 0 if both persons are looking at something else and to 1 if at least one person is looking at the other at the actual frame. However, humans do not look at their interaction target all the time, see section 5.1. Therefore we do not use only the actual frame information but a window around the actual frame. If the window size is 5 (several sizes are tried), then the 2 frames before and the 2 frames after the actual frame count as well. First, the percentages of A looking at B ($= l(A,B) = \frac{number\ of\ frames\ A\ looks\ at\ B}{window\ size}$) and B looking at A ($= l(B,A)$) are calculated for that time window. Next, these two percentages must be combined to one score number. The combining function $f(l(A,B), l(B,A))$ should satisfy the following constraints:

- $0 \leq f(x,y) \leq 1, \ \forall x,y \in [0,1]$
  The score should be between 0 and 1.

- $f(x,y) = f(y,x), \ \forall x,y \in [0,1]$
  We calculate the score for an undirected interaction, that means that both directions should evaluate to the same score.

- $f(0,0) = 0$
  The score should be 0, if both persons are not looking at each other.

- $f(1,x) = f(x,1) = 1, \ \forall x \in [0,1]$
  The score should be 1, if one person is looking at the other for the whole time window. Although one is not looking at the other at all, it is certain that there is an interaction. See section 5.1 for the intuition behind this choice.

- $f_c(x) = f(x, c) = f(c, x)$ should be linear ($\forall c \in [0, 1]$, c constant)
  This is not a necessary constraint, but the function should be as simple as possible.

According to these constraints, we choose the function:

$$f(x, y) = x + y - (x * y) \tag{5.1}$$

The graph of the function is shown in figure 5.6. The score for person-object interactions is calculated with help of a time window as well, but there is only one looking direction (the object cannot look at the person). Therefore the percentage of the person looking at the object in relation to the window size can be directly used as score.

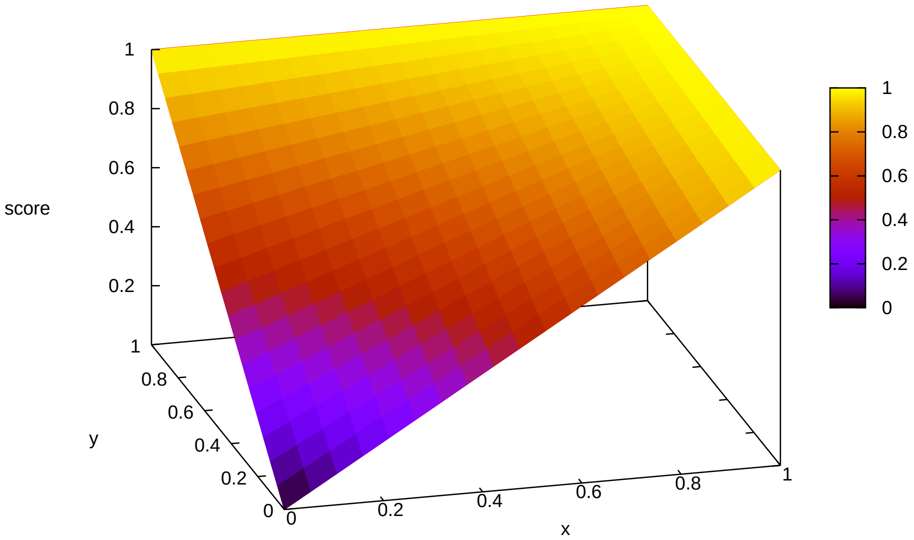


Figure 5.6: **The score function** $f(x, y) = x + y - (x * y)$, x is the percentage of the frames person A looks at person B (in relation to the window size) and y is respectively the other direction (B looking at A).

The next step is to detect the interactions from the score, which is done by defining a classification threshold $t \in (0, 1)$. A soft decision with probability like output is needed. In the following, we refer to that value as probability, but we have to keep in mind that it is not a *true* probability, because no distribution is defined. We could use the scores directly, but we do not know how many percent of mutual gaze is needed to assume that an interaction happens. Therefore we introduce a variable threshold $t$, $t \in (0, 1)$, that specifies this percentage of mutual gaze. A transformation function $g_t(score)$ is needed that maps each score value to the probability. The transformation function should satisfy the following constraints:

- $0 \leq g_t(x) \leq 1, \ \forall x \in [0, 1]$
  The probability should be between 0 and 1.

- $g_t(0) = 0$
  If the score is equal to 0, then the probability should be 0.

- $g_t(1) = 1$
  If the score is equal to 1, then the probability should be 1.

- $g_t(t) = 0.5$
  If the score is equal to the threshold, then the probability should be 0.5.

- $g'_t(x) > 0, \ \forall x \in [0,1]$
  g must be strictly increasing, because higher score values should also lead to higher probability values.

According to these constraints, we choose the function:

$$g_t(x) = x^{\frac{\log(0.5)}{\log(t)}} \tag{5.2}$$

The graph of the function is shown in figure 5.7. For $t = 0.5$, $g_t$ is the identity function, that means the score is used directly as probability. Otherwise, $g_t$ is convex for $t < 0.5$, and $g_t$ is concave for $t > 0.5$. Finally the probability is compressed by the function:

$$h(x) = 0.8 * x + 0.1 \tag{5.3}$$

The function h is the linear mapping from range $[0,1]$ to $[0.1, 0.9]$. This is done to avoid the probabilities 0 and 1, because they represent certain decisions (and certain decisions should be avoided in the first layer to keep all possibilities open for the second layer, see also section 5.3). The final probability calculation for person A and B is shown in equation 5.4, and respectively the same for person A and object O in equation 5.5. In the following part, the threshold $t$ is referred as person-person threshold (for calculating the person-person interaction) and person-object threshold (for calculating the person-object interaction).

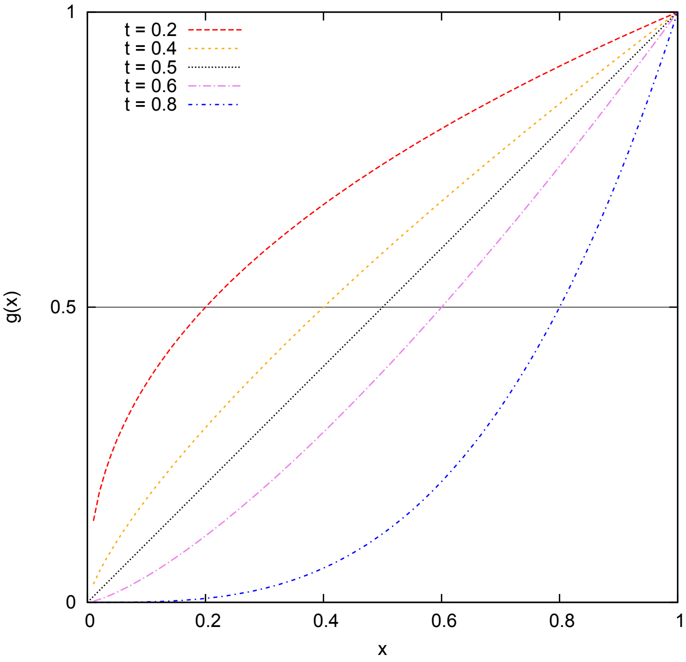


Figure 5.7: **The probability function** $g_t(x) = x^{\frac{\log(0.5)}{\log(t)}}$

$$
\begin{aligned}
p(A,B) &= p(B,A) \\
&= h\Big(g_t\Big(f\big(l(A,B), l(B,A)\big)\Big)\Big) \\
&= 0.8 * \Big(l\,(A,B) + l\,(B,A) - \big(l\,(A,B) * l\,(B,A)\,\big)\Big)^{\frac{\log(0.5)}{\log(t)}} + 0.1
\end{aligned}
\tag{5.4}
$$

$$
\begin{aligned}
p(A,O) &= p(O,A) \\
&= h\Big(g_t\big(l(A,O)\big)\Big) \\
&= 0.8 * l\,(A,O)^{\frac{\log(0.5)}{\log(t)}} + 0.1
\end{aligned}
\tag{5.5}
$$

Each time frame, the above explained interaction probability is calculated for every person-person and every person-object pair. A pair of objects cannot interact, therefore the probability is set to 0 ($p(O1,O2) = 0$). The result is a fully connected weighted graph and the edge weights represent the probability for interaction between nodes.

## 5.3 Group-Activity Layer

The goal of this layer is to find the most probable group-activity explanation for the fully connected weighted graph calculated before. That means we want to know which group-activities are most likely happening. All possible group-activities for up to 4 persons are shown in table 5.2. A possible explanation is a template graph or a combination of template graphs that represent(s) the ongoing activities. Several concurrent group-activities are detected, if the explanation consists of more than one template graph. However, some assumptions are made to reduce the search space:

1. Every person is participating in exactly one group-activity.
2. Every group is interacting with at most one object.

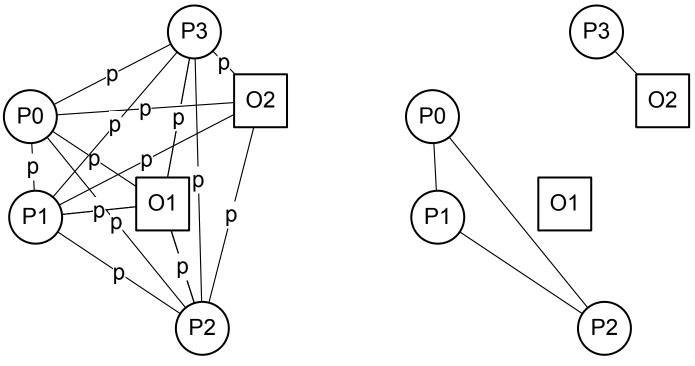
In reality, the first assumption does not always hold. For example, it is possible to listen to a presentation and take notes at the same time. In this case, our system should detect the most dominant activity (regarding the example before: if someone is interacting more with his notebook than with the presenter, then the activity 'using notebook' is detected instead of 'presentation). Weighting the activities is implicitly done by choosing the time window length. A smaller window leads to a system, that adapts faster. The case is similar for the second assumption: in reality it is possible that a group interacts with several objects at the same time. However, the system should again focus on the most dominant object.

According to the assumptions above, the goal of this layer can be expressed as: Find a combination of template graphs (see section 5.1) that fits the calculated probability graph (see section 5.2) best. The idea is to cover the probability graph with one or more template graphs, so that each node in the probability graph is covered by exactly one node from the template graphs. We first exactly define the term 'fitting combination of template graphs' for a given probability graph. Then a score calculation is defined, thus the best fitting combination is the combination with the highest score.

We define $P = \{all\ present\ persons\}$, $O = \{all\ present\ objects\}$, $V = P \cup O$ the set of vertices and $p : V \times V \to [0,1]$ is the probability (weight) function calculated at the interaction layer. A combination graph $G_c = (V_c, E_c)$ from a set of template graphs $S = \{T_1, T_2, ...T_n\}$, $T_i = (V_i, E_i)$, $T_i\ like\ in\ table$ 5.2 is defined as:

$$V_c = \bigcup_{i=1}^{n} V_i, \quad E_c = \bigcup_{i=1}^{n} E_i$$

A combination graph fits the probability graph, if a bijective function $F : V \to V_c$ exists, that maps each person in $V$ to a person in $V_c$ and each object in $V$ to an object in $V_c$. In the following part, we consider $V$ and $V_c$ to be the same for a fitting combination graph. We should note here, that the probability graph is fully connected and has edge weights, whereas the combination graph has binary edges and is in general not fully connected. An example is shown in figure 5.8. The bijective vertex mapping is expressed by giving $V$ and $V_c$ the same labels. A combination graph with a corresponding mapping function defines a classification hypothesis. This hypothesis groups the persons and objects together that belong to the same template graph and assigns the group-activity defined by each template graph. Furthermore, every fitting combination graph is a valid hypothesis according to the assumptions defined above. Assumption 1 is satisfied, because the bijective mapping asserts that every person belongs to exactly one template graph and thus is performing one group-activity. Assumption 2 is satisfied, because every template graph contains at most one object.



(a) Probability graph

(b) Combination graph

Figure 5.8: Example probability graph with fitting combination graph

The best fitting combination graph is found through a weighting function $w(V, p, E_c)$. This function assigns a score to every fitting combination graph $(V, E_c)$ for a given probability graph $(V, p)$. There is more than one possibility to define a weighting function, but we propose the following method: We consider the probability graph as a set $\mathcal{R}$ of $\frac{|V|*(|V|-1)}{2}$ binary random variables that are independent from each other. Every variable $R \in \mathcal{R}$ corresponds to an edge ($R = R_{v1,v2} = R_{v2,v1}$, $v1, v2 \in V$), and the assigned edge probability is considered as observed evidence for that variable. A combination graph is considered as instantiation $I : \mathcal{R} \to \{0,1\}$ of all variables according to

$$I(R_{v1,v2}) = \begin{cases} 1 & if\ (v1,v2) \in E_c, \\ 0 & otherwise. \end{cases} \tag{5.6}$$

Then it is possible to calculate the posterior probability of every variable:

$$P\big(R_{v1,v2} = I(R_{v1,v2})\big) = \begin{cases} p(v1,v2) & if\ (v1,v2) \in E_c, \\ 1 - p(v1,v2) & otherwise. \end{cases} \tag{5.7}$$

The posterior probability of an instantiation is a measure of how likely the instantiation is and thus we choose it as the score. All variables are independent from each other, and therefore we can calculate the score according to equation 5.8. Because of this calculation method, the probabilities 0 and 1 should never be mapped to an edge in the first layer: then a possibility of 0 would be assigned to some combination graphs. Each of these graphs would then be considered as *impossible*, only because one of the edges has a probability of 0 or 1. However, we want an ordering of all graphs, which means that the possibility can get close to 0, but should never be equal 0.

$$\begin{aligned} score = w(V,p,E_c) &= \prod_{\{v1,v2\}\subset V} P\big(R_{v1,v2} = I(R_{v1,v2})\big) \\ &= \prod_{\substack{\{v1,v2\}\subset V: \\ (v1,v2)\in E_c}} p(v1,v2) \prod_{\substack{\{v1,v2\}\subset V: \\ (v1,v2)\notin E_c}} \big(1 - p(v1,v2)\big) \end{aligned} \tag{5.8}$$

The next step is to find the combination graph with the best score. This is a typical optimization problem and different algorithms can be used to solve it. One method would be to calculate the score for all possible combination graphs. However, the complexity is high:

**Proposition 1.** *The number of all possible combination graphs is in* $\mathcal{O}\left(B(p) * \frac{o!}{(o-p)!} * 6^p\right)$, *where* $p =$ *number of persons,* $o =$ *number of objects and* $B(n)$ *is the* $n^{th}$ *Bell number.*

*Proof.* According to assumption 1, each person belongs to exactly one template graph, but any partition of the persons is possible. The number of partitions of a set of p persons is $B(p)$. Each partition consists of at least 1 set (all persons in one set) and at most p sets (each person in a separate set). We are interested in the upper bound, therefore we consider the worst case of p sets. Each set can contain one of the objects or no object according to assumption 2. Every object can be at most in one of the person sets. First we decide how many of the p sets contain an object. There are $\binom{p}{0}$ possibilities to choose no object, $\binom{p}{1}$ possibilities to choose 1 object,... $\binom{p}{p}$ possibilities to choose p objects (the sum goes to $\min(p,o)$ to cover the case $p > o$). There are $\frac{o!}{(o-k)!}$ possibilities to choose k objects for the k sets that get objects. Each set without an object has at most 2 possibilities for the graph structure: discussion or monologue. Each set with object has at most 3 possibilities for the graph structure: separate object interaction, cooperative object interaction or presentation. That leaves us with the upper bound for the number of possible combination graphs $N(p, o)$:

$$\begin{aligned}
N(p,o) &= B(p) * \sum_{k=0}^{\min(p,o)} \left( \binom{p}{k} * \underbrace{\frac{o!}{(o-k)!}}_{\leq \frac{o!}{(o-p)!}} * \underbrace{3^k * 2^{p-k}}_{\leq 3^p} \right) \\
&\leq B(p) * \frac{o!}{(o-p)!} * 3^p * \sum_{k=0}^{\min(p,o)} \binom{p}{k} \\
&\leq B(p) * 3^p * \frac{o!}{(o-p)!} * 2^{\min(p,o)} \in \mathcal{O}\left( B(p) * \frac{o!}{(o-p)!} * 6^p \right)
\end{aligned} \tag{5.9}$$

□

Until now, we have only considered the number of possible combination graphs. The score calculation also takes time: $\frac{(p+o)*(p+o-1)}{2} \in \mathcal{O}((p+o)^2)$ multiplications are needed for each combination graph. Some optimizations can be made to reduce the total number of multiplications, but it is still growing very fast. Obviously, calculating all possible scores is not possible for a large number of persons or objects. Hence we use a branch and bound technique for searching. One characteristic of the template graphs is, that any node can be removed and it still remains a (smaller) valid template graph. Therefore nodes can be kept separate if all adjacent edges are below 0.5, i.e. they are excluded from the search. If they would be in a group with other nodes, then the same graph without that node in the group always gets a higher score. Also, all branches are pruned, where the score gets below the best score so far. The score is calculated by multiplying probabilities, so it is always descending.

The combination graph with the best score is the desired classification result. Each template graph in the combination graph represents one group of persons and the graph type defines the recognized activity.

# 6

# System Evaluation

As described in chapter 5, the system consists of two parts: the interaction layer and the group-activity layer. Several classification runs have been conducted with different parameters: window size, person-person threshold and person-object threshold (for their explanation, see section 5.2). All parameters belong to the interaction layer. At the group-activity layer, the proposed score calculation method is fixed and has no parameters. First, the interaction layer is evaluated separately. The results of the group-activity layer depend on the results of the interaction layer, therefore it is then evaluated for different first layer parameters.

The evaluation is done with a 10 minutes meeting sequence from the data set described in section 4.1. The ground-truth is labeled by hand, which means that it represents the individual view of the annotator. However, both video and audio were used for labeling and therefore the situations were quite obvious with the help of context information (e.g. people saying the name of the person they are addressing, question/answer patterns). Ambiguous situations usually occurred when two people were interacting (e.g. one giving a pen to the other) and the other people were watching them. Then it is not clear whether the other people belong to the group or not. However, these situations only happened for a short time, usually in the breaks of a discussion/presentation or before the meeting started.

## 6.1 Interaction Layer

First, we explain in section 6.1.1 how the interaction layer is evaluated and then the results are shown in section 6.1.2.

### 6.1.1 Evaluation Method

The task of the interaction layer is to estimate a probability value for all possible person-person and person-object interactions. We evaluate every interaction result of the system by first turning it into a hard decisions: if the probability is above 0.5 it is classified as interaction and no interaction otherwise. Then the decision is compared to the ground-truth: if both entities (persons or objects) are not in the same group, then there is no interaction between them. If they are in the same group, then the interaction depends on the group-activity (see table 5.2). Person-person and person-object interactions are evaluated separately, because their estimation is done independently and their parameters do not effect each other (the person-person interaction estimation is independent of the person-object threshold and vice versa).

As evaluation result, a confusion matrix is calculated. The dimension is 2 × 2, because there exist two classes: interaction and no interaction. The interactions are evaluated per frame and per entity pair. That means that frames with more persons/objects contain more cases. A frame with $p$ persons and $o$ objects contains $\frac{p*(p-1)}{2}$ person-person interactions and $p*o$ person-object interactions. Two performance measures are then calculated from the confusion matrix: the recognition rate and the F1-measure. The recognition rate is the proportion of the correctly classified cases to all cases, the F1-measure is the harmonic mean of precision and recall:

$$\text{recognition rate} = \frac{\text{number of correctly classified cases}}{\text{number of all cases}} \tag{6.1}$$

$$\text{precision} = \frac{\text{true positive}}{\text{true positive} + \text{false positive}} \tag{6.2}$$

$$\text{recall} = \frac{\text{true positive}}{\text{true positive} + \text{false negative}} \tag{6.3}$$

$$\text{F1-measure} = 2 * \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}} \tag{6.4}$$

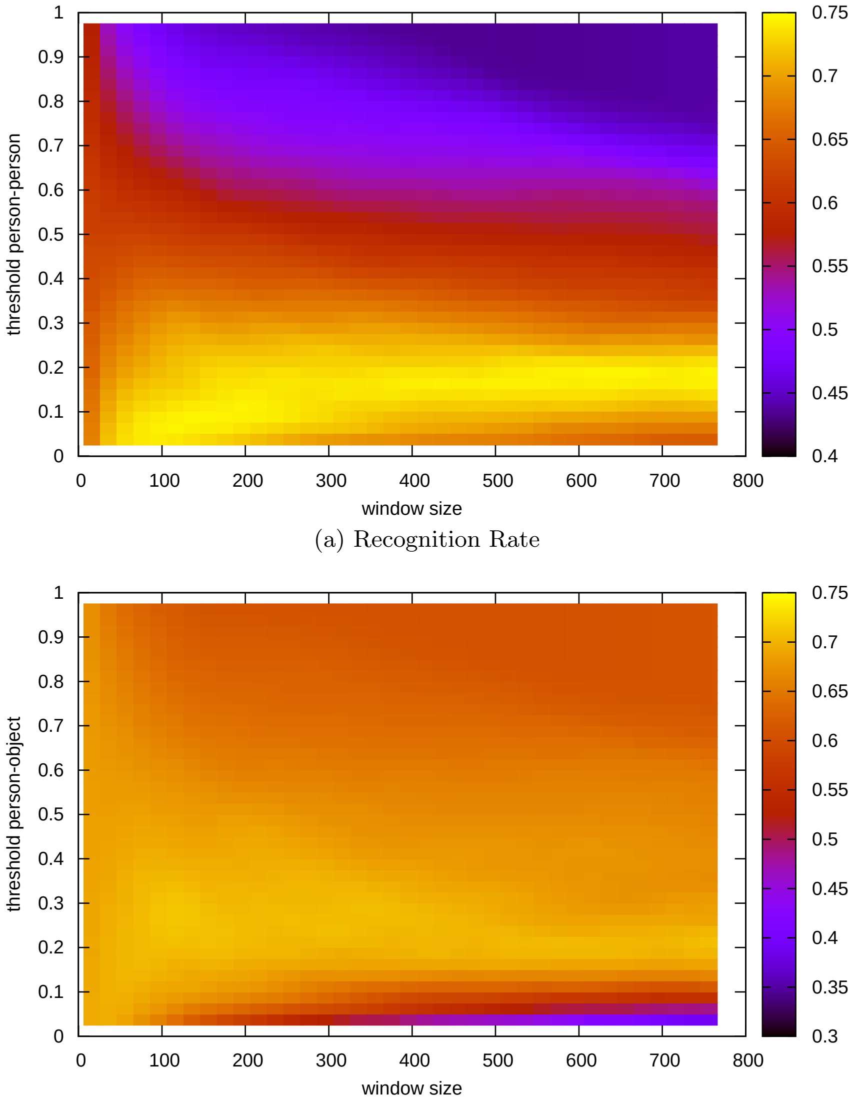
Precision, recall and F1-measure are calculated regarding one class, that means we have to decide whether 'interaction' or 'no interaction' is considered as positive and vice versa. We decided to use the F1-measure to be able to detect if either recall or precision values are bad although the overall recognition rate is good.

### 6.1.2 Recognition Results

First, the recognition rate and the F1-measure are shown in heat maps for a part of the parameter space. The person-person and person-object threshold (see section 5.2) is evaluated from 0.025 to 0.975 with a step size of 0.025. The window size is evaluated from 6 to 766 frames with a step size of 20 (i.e. 0.4 to 51.1 seconds, because the frame rate is 15 fps). Finally, the confusion matrices for good parameter values are shown.

### Person-Person Interaction

The person-person interaction recognition depends on the window size and the person-person threshold. The results are shown in figure 6.1. The F1-Measure is calculated regarding the case 'no interaction' (i.e. 'no interaction' is the positive case, see equation 6.4), because that case occurs less than 'interaction'. The maximum of the F1-Measure graph is not as clear as the maximum of the recognition rate, but both maxima correspond to each other. Good results are achieved with a window size of at least 100 frames and a person-person threshold close to 0. If the window size increases, then the person-person threshold should be increased as well until 0.15, where it stays almost constant for bigger window sizes. In table 6.1 are some results shown for selected parameter values. We were able to achieve recognition rates over 70% and F1-Measures over 60%.



(a) Recognition Rate

(b) F1-Measure

Figure 6.1: Person-person interaction results for variable parameter values

| window size | 120 | 120 | 400 | 400 |
|---|---|---|---|---|
| pers.-pers. thresh. | 0.0001 | 0.875 | 0.05 | 0.375 |
| recognition rate | 72.2% | 48.5% | 70.3% | 64.6% |
| F1-measure | 59.1% | 62.6% | 54.5% | 68.6% |
| conf. matrix | $\begin{bmatrix} 11218 & 12851 \\ 2673 & 29056 \end{bmatrix}$ | $\begin{bmatrix} 24055 & 14 \\ 28703 & 3026 \end{bmatrix}$ | $\begin{bmatrix} 9383 & 13459 \\ 2203 & 27645 \end{bmatrix}$ | $\begin{bmatrix} 20422 & 2420 \\ 16239 & 13609 \end{bmatrix}$ |

Table 6.1: **Person-person interaction results for selected parameter values.** First row of the confusion matrix is true ’no interaction’ and second row is true ’interaction’, the columns represent respectively the hypothesis cases.

### Person-Object Interaction

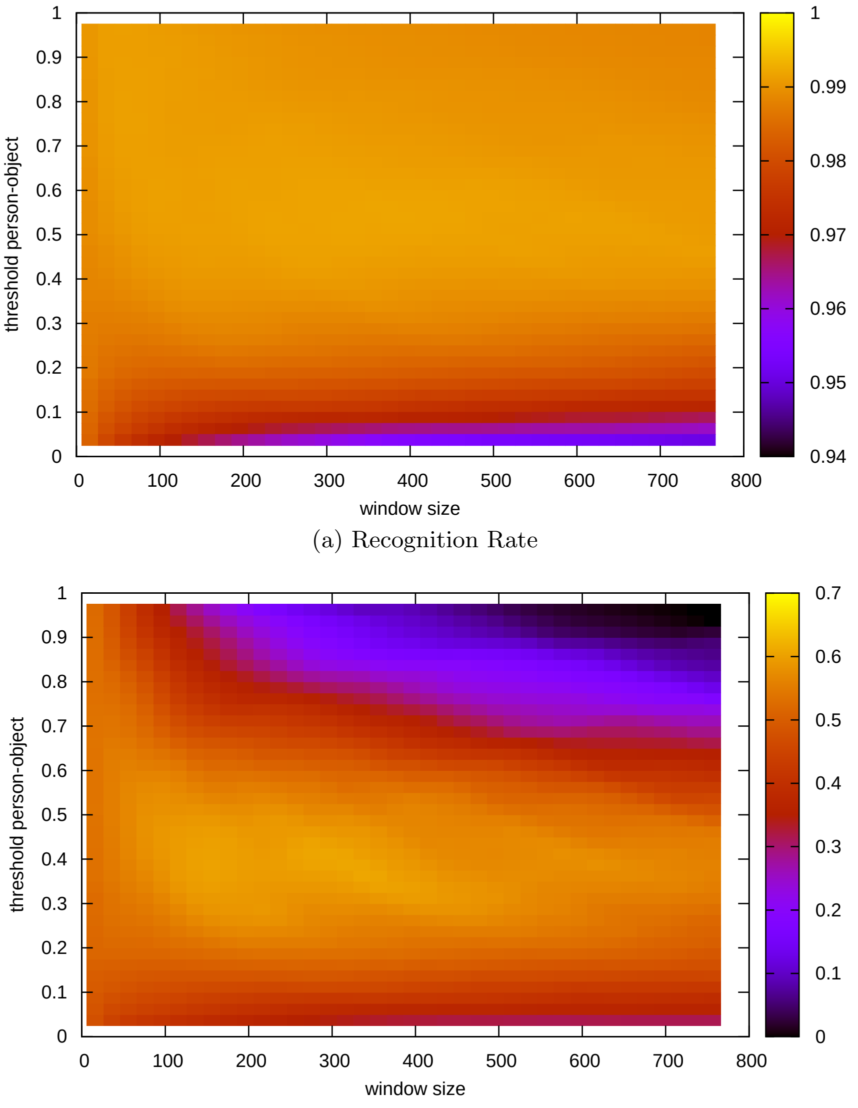
The person-object interaction recognition depends on the window size and the person-object threshold, results are shown in figure 6.2. The F1-Measure of class ’interaction’ is calculated, because it occurs less than ’no interaction’ and true positives of ’no interaction’ would dominate the F1-Measure. According to the recognition rate, all person-object thresholds above 0.2 seems fine. However, when taking the F1-Measure into account, we can see that the results gets worse when choosing too high values. This is due to a bad precision of ’interaction’. However, the precision has almost no influence on the recognition rate, because it is dominated by the large number of truly recognized ’no interaction’. In figure 6.2b, the F1-Measure is shown. We can see that the optimum values for the person-object threshold are between 0.2 and 0.3 for window sizes greater than 100.

| window size | 120 | 120 | 400 | 400 |
|---|---|---|---|---|
| pers.-obj. thresh. | 0.375 | 0.875 | 0.25 | 0.75 |
| recognition rate | 99.0% | 99.1% | 98.6% | 99.0% |
| F1-measure | 59.8% | 36.3% | 54.9% | 29.7% |
| conf. matrix | $\begin{bmatrix} 1106980 & 7018 \\ 4050 & 8218 \end{bmatrix}$ | $\begin{bmatrix} 1113180 & 818 \\ 9364 & 2904 \end{bmatrix}$ | $\begin{bmatrix} 1054648 & 12556 \\ 2788 & 9353 \end{bmatrix}$ | $\begin{bmatrix} 1066698 & 506 \\ 9933 & 2208 \end{bmatrix}$ |

Table 6.2: **Person-object interaction results for selected parameter values.** First row of the confusion matrix is true ’no interaction’ and second row is true ’interaction’, the columns represent respectively the hypothesis cases.

## 6.2 Group-Activity Layer

The group-activity layer depends on the interaction layer, therefore the higher-level results of the same runs as in the interaction layer are shown. This layer itself has no parameters, but the evaluation is done for different parameters of the interaction layer.



(a) Recognition Rate

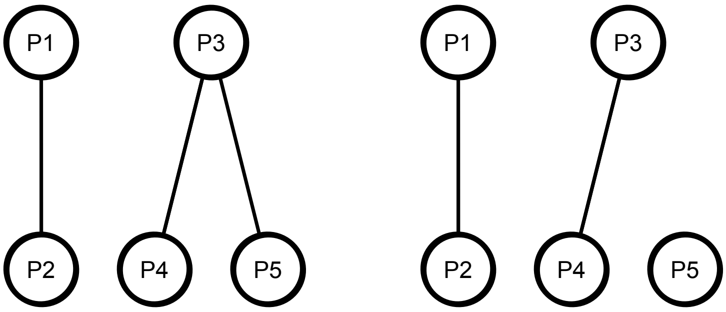
(b) F1-Measure

Figure 6.2: Person-object interaction results for variable parameter values

### 6.2.1 Evaluation Method

The evaluation of the group-activity layer is non-trivial, because two classification tasks are done for every-frame: first, there is the separation of the persons and objects in clusters and second, there is the group-activity type associated to every cluster. Therefore, the recognition engine is evaluated separately for clustering and group-activity type.

A simple approach would be to compare for each frame if ground-truth and hypothesis match entirely. However, this approach overlooks partial matches of the system (see figure 6.3). Although the discussion of P1 and P2 is recognized correctly, the frame is counted as false because the discussion between P3, P4 and P5 is not recognized. A second disadvantage of the simple approach is the ignorance of complexity. A frame with five persons is harder to classify correctly than a frame with only one person, because the number of possible classifications is higher. To overcome these disadvantages, the evaluation method proposed here is centered on a single person: a frame sequence is



(a) Ground-Truth (b) Hypothesis

Figure 6.3: Partial Match

evaluated from the relative view of each involved person. The final result is then the average over all individual person results, weighted by the number of frames the person was present. In the example above (figure 6.3), the individual classification for P1 and P2 is correct (because they are together with the right persons, doing the right action). The result for P3, P4 and P5 is false, therefore the overall error rate is $3/5 = 0.6$(3 out of 5 are classified wrong). The person centered method also leads to a weighting regarding the number of persons: a sequence of 1 frame with 5 persons is weighted equal to a sequence of 5 frames with 1 person. An example sequence with the individual views of the persons is shown in table 6.3.

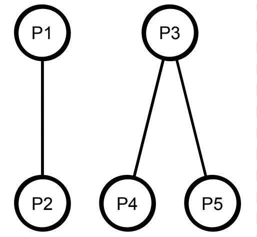
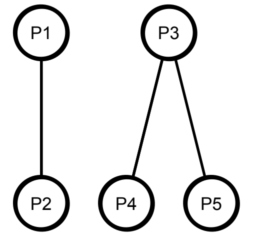
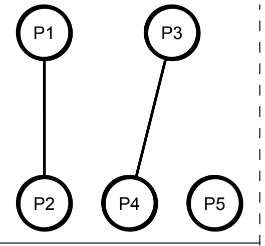
| | | Frame 1 | Frame 2 | Frame 3 |
|---|---|---|---|---|
| | |  |  |  |
| view P1 | cluster<br>action | {P2}<br>discussion | {P2}<br>discussion | {P2}<br>discussion |
| view P2 | cluster<br>action | {P1}<br>discussion | {P1}<br>discussion | {P1}<br>discussion |
| view P3 | cluster<br>action | {P4,P5}<br>monologue P3 | {P4,P5}<br>monologue P3 | {P4}<br>discussion |
| view P4 | cluster<br>action | {P3,P5}<br>monologue P3 | {P3,P5}<br>monologue P3 | {P3}<br>discussion |
| view P5 | cluster<br>action | {P3,P4}<br>monologue P3 | {P3,P4}<br>monologue P3 | {}<br>no action |

Table 6.3: Individual view sequences

For the error rate calculation we use two different methods, like in [19]: the word error rate (WER) and frame recognition rate (FRR). The WER is the same as the one used by speech recognition systems. When comparing a hypothesis with a ground-truth sequence of words, the WER is calculated from the sum of insertions (I), deletions (D) and substitutions (S) divided by the length of the ground-truth ($WER = \frac{I+D+S}{length}$). The sequence of words is derived from the sequence of frames by deleting repetitions: 'aaabbbbbcca' becomes 'abca'. Therefore the WER takes only the ordering of group-activities (or clusters) into account, the time-alignment is lost. According to [19], the WER is a good measure, because the ordering is more important than the time-alignment.

To evaluate the time-alignment as well, we also calculate the FRR. For the sequence of group-activity types, this is the number of correctly classified frames (C) divided by the length of the sequence ($FRR = \frac{C}{length}$). For the sequence of clusters, we also want to take partly recognized clusters into account. To explain that, see again figure 6.3, from view P3: the true cluster of P3 is {P4, P5}, but recognized is {P4}. However, the recognition engine is not completely wrong, because P4 is correct. Therefore we define the error between the true cluster set (TC) and the hypothesis cluster set (HC) as the sum of deleted elements (DE) and inserted elements (IE) divided by the maximum number of elements:

$$error(TC, HC) = \frac{DE + IE}{max(|TC|, |HC|)} = 1 - \frac{TC \cap HC}{max(|TC|, |HC|)} \tag{6.5}$$

The error is 0, only if TC and HC are equal. And the error is 1 if TC and HC have no elements in common. The FRR for a cluster sequence is then calculated in the following way:

$$FRR = \sum_{frames} 1 - error(TC, HC) \tag{6.6}$$

When calculating the FRR or WER for a distinct person, then the data is first split to sequences where the person is present. The final FRR/WER is then the weighted sum (regarding length) of all these sequences:

$$FRR(overall) = \sum_{persons} \sum_{sequences} FRR(person, sequence) \tag{6.7}$$

The WER is calculated respectively in the same way. If all sequences for all persons have been classified correctly, then the FRR is 1 and the WER is 0. Therefore higher values are better for the FRR and lower values are better for the WER. FRR and WER are calculated separately for person clustering, object clustering and activity recognition.

### 6.2.2 Recognition Results

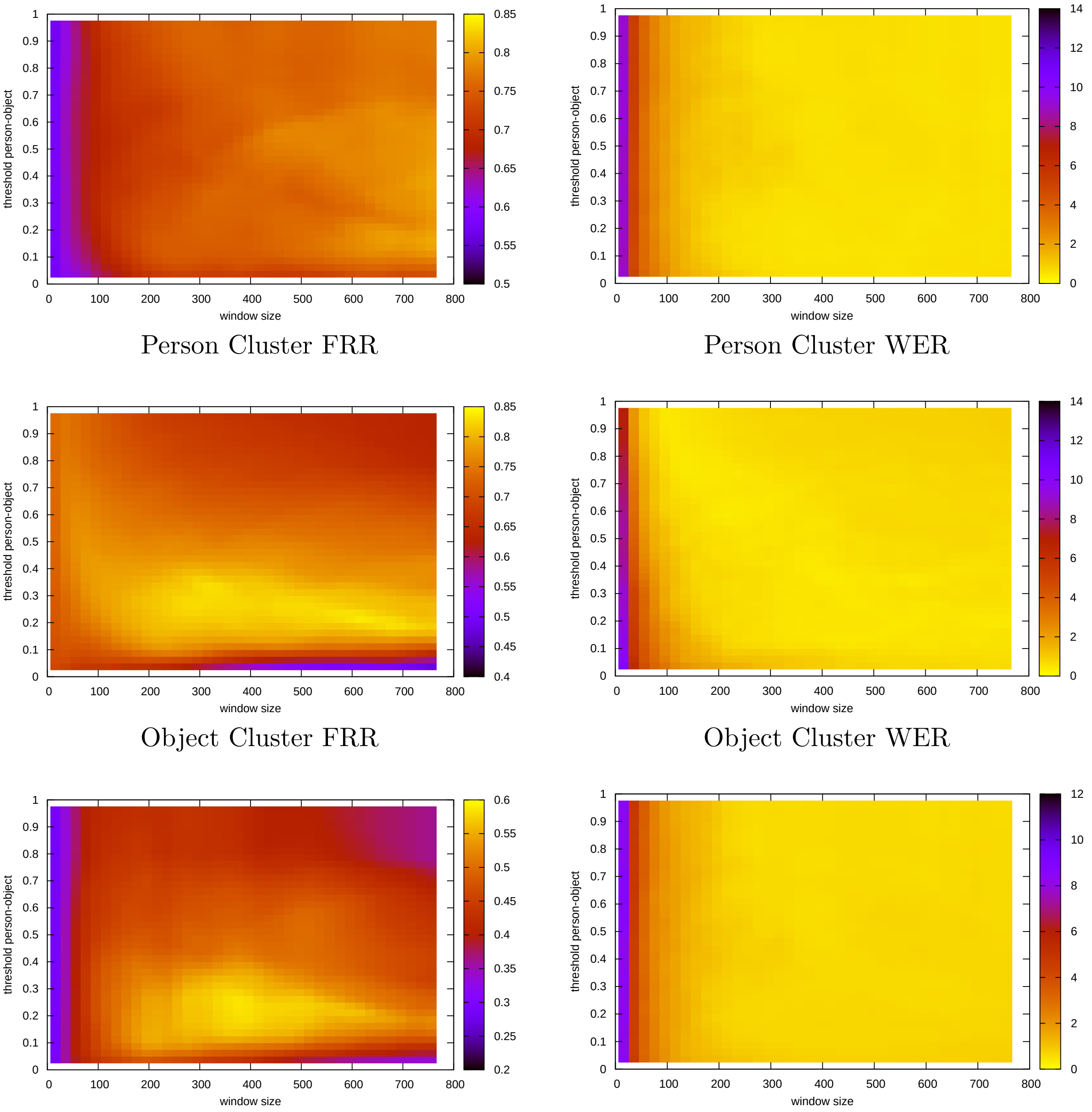
Like explained in section 6.2.1, there are 3 different recognition tasks evaluated: person clustering, object clustering and activity recognition. For each of these 3 tasks, two different measures are calculated: the FRR and the WER. The results depend on the 3 parameters of the interaction layer: the person-person threshold, the person-object threshold and the window size. For optimizing the recognition results, a systematic parameter evaluation is done in the following part. In each of the evaluation runs (see figures 6.4, 6.5 and 6.6), one of the parameters is kept constant at a good value and the results are shown for different values of the other two parameters.

Evaluation results for person-object threshold and window size are shown in figure 6.4. The graphs of person cluster FRR and WER show no clear maximum. The results are almost independent from the person-object threshold. This is expected, because the grouping of the persons mainly depends on the person-person threshold. The recognition rate increases with bigger window sizes between 0 and 200 frames. The object cluster FRR and activity FRR graphs show good results for a window size of 300 to 500 and a person-object threshold of 0.2 to 0.3. The maxima of the object cluster WER graph have slightly higher person-object thresholds than the maxima of the object cluster FRR graph. That means that objects must be for more time the visual focus target until an interaction is assumed. That makes sense, because the time-alignment does not count for the WER and thus it is more important to be sure that an interaction happens than a fast adaptation of the system regarding new interactions. The activity WER graph shows the same behaviour like the person cluster graphs, there is no clear maximum and the WER decreases with bigger window sizes.

Evaluation results for person-person threshold and window size are shown in figure 6.5. All graphs show that the recognition rates do not depend on the window size, as long as it is bigger than 200 (and window sizes below 200 lead to poor results). The graphs of the person cluster FRR, object cluster FRR and activity FRR show the same tendency: low person-person threshold values around 0.1 are the optimum. At first glance, it is surprising that the object cluster FRR highly depends on the person-person threshold. However, this is due to the fact that we assign an object to a *group* of persons. That means, if the group is not detected correctly, the assignment of the object must be wrong for at least a part of the group (either the correct persons in the group or the wrongly added persons). All WER graphs have no clear maximum, we can only say for sure that window sizes below 200 lead to poor results.

Evaluation results for the person-person and person-object threshold are shown in figure 6.6. The person cluster FRR depends mainly on the person-person threshold, values between 0 and 0.1 show good results. For person-object thresholds below 0.3, the object cluster FRR depends on both, person-person and person-object threshold. Above 0.3, there is no more dependence on the person-person threshold and the results for values between 0.3 and 0.6 are good. However, the maximum for the person cluster FRR is reached for person-person values between 0 and 0.1 and person-object values between 0.2 and 0.3. The object cluster WER shows almost no dependence on the person-person threshold. Also there is no clear maximum for the person-object threshold, good values are between 0.2 and 0.5. The person cluster WER and the activity FRR have a common unambiguous maximum at a person-person threshold of 0.05 and a person-object threshold of 0.2. This maximum is consistent with the person cluster FRR, the object cluster FRR and the object cluster WER graph. However, the activity WER graph has another maximum at a person-person threshold from 0.35 to 0.5 and a person-object threshold above 0.6. That means that there are two possible ways to optimize the system: either the system is optimized regarding the time alignment *or* the ordering of the recognized activities. There is no good trade-off between these two, because the values between the two maxima all lead to poor results. In all three runs (see figures 6.4, 6.5 and 6.6), the activity FRR graph seems like a combination of the person cluster FRR and the object cluster FRR graph. This is due to the dependence of the activity recognition on person and object clustering.

Evaluation results for some good parameter values are shown in table 6.4. Two window sizes were chosen: 120 and 400. The parameters of the runs 1 and 3 are optimized regarding the activity FRR, the other two are optimized regarding the activity WER. The optimized results are comparable. However, the WERs of the first run are all



Person Cluster FRR

Person Cluster WER

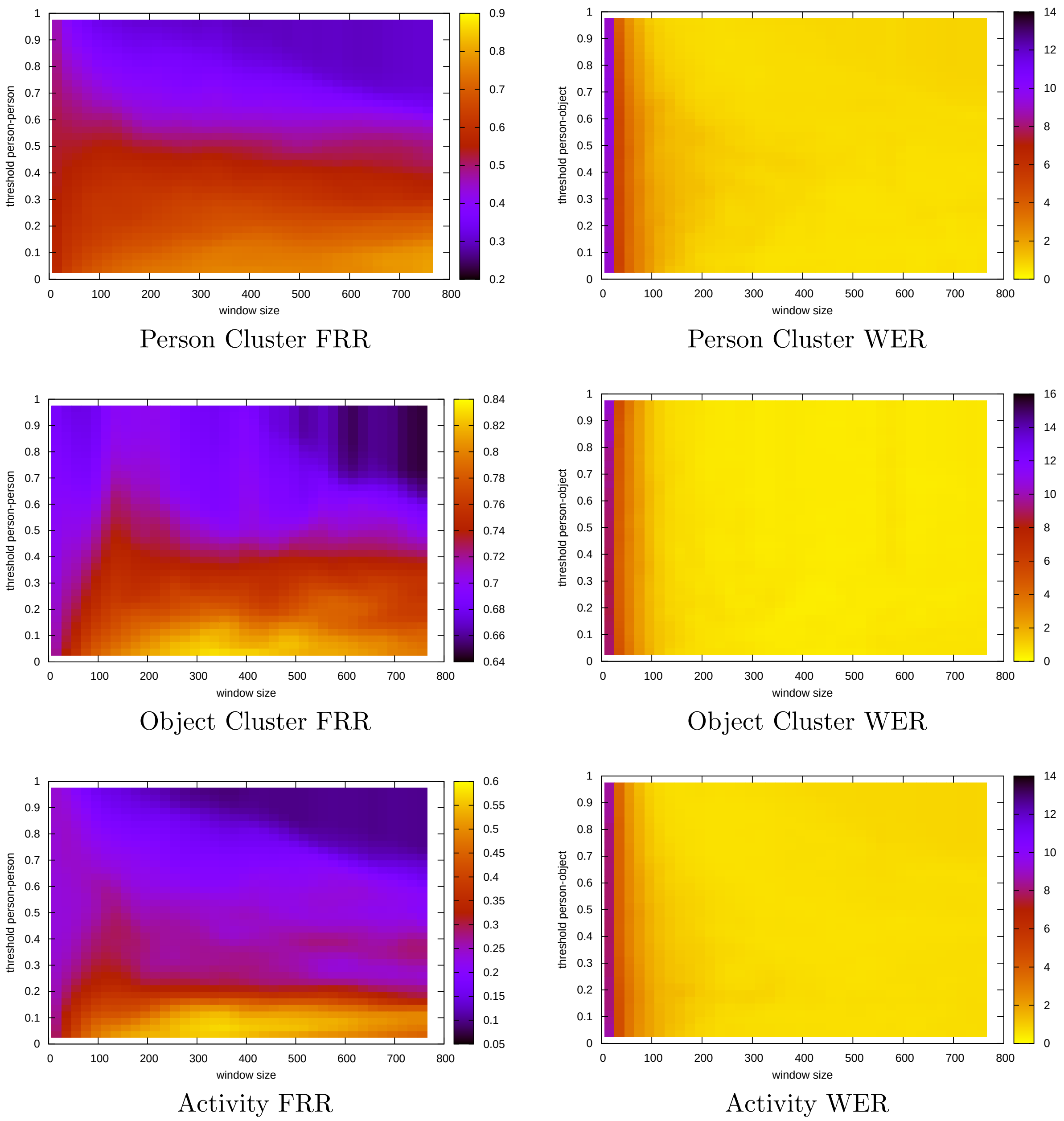
Object Cluster FRR

Object Cluster WER

Activity FRR

Activity WER

Figure 6.4: **Person-object threshold vs. window size.** Group-activity layer results for variable parameter values of person-object threshold and window size are shown. The person-person threshold is kept constant at 0.05.



Person Cluster FRR

Person Cluster WER

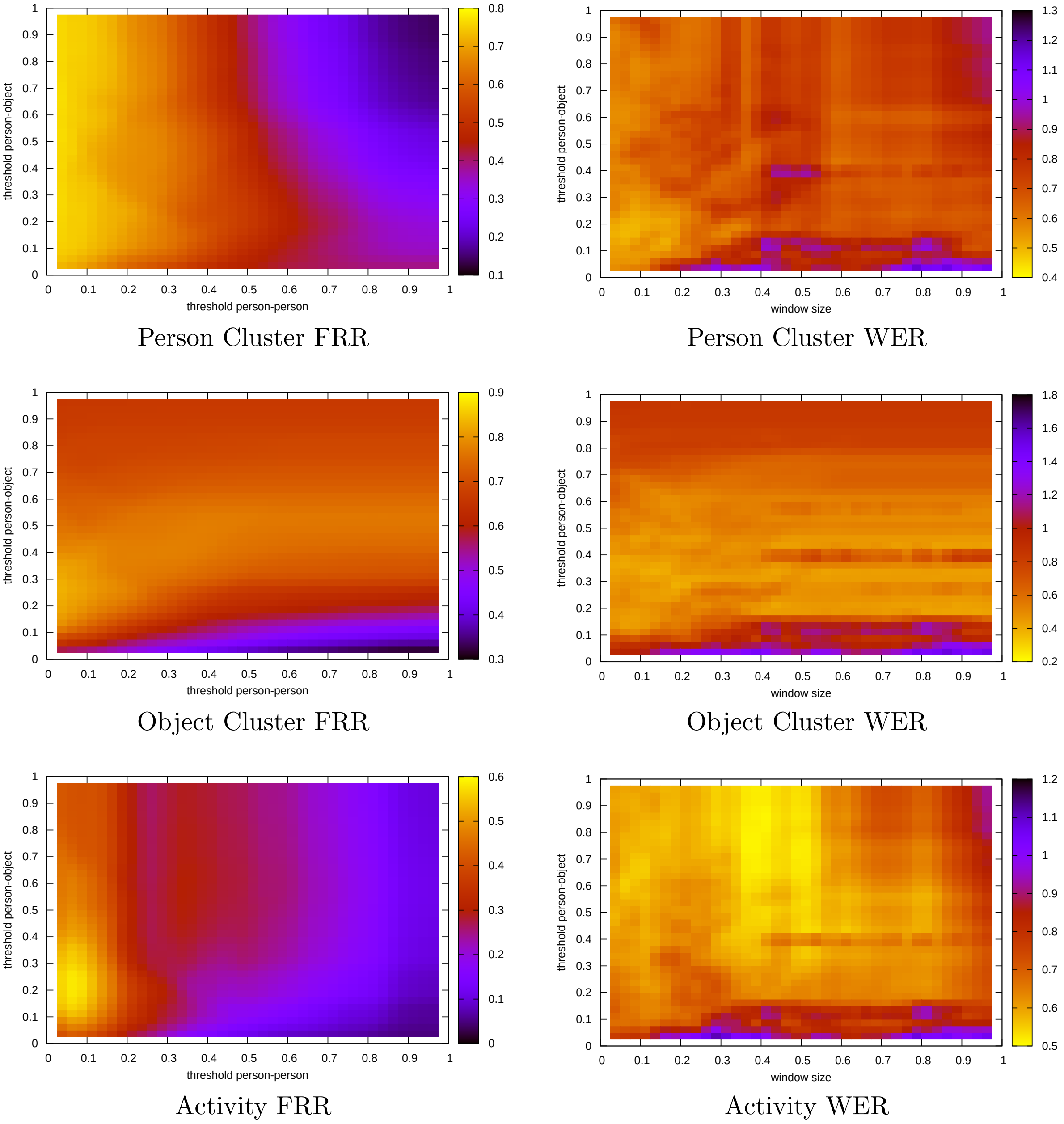
Object Cluster FRR

Object Cluster WER

Activity FRR

Activity WER

Figure 6.5: **Person-person threshold vs. window size.** Group-activity layer results for variable parameter values of person-person threshold and window size are shown. The person-object threshold is kept constant at 0.3.



Person Cluster FRR

Person Cluster WER

Object Cluster FRR

Object Cluster WER

Activity FRR

Activity WER

Figure 6.6: **Person-person threshold vs. person-object threshold.** Group-activity layer results for variable parameter values of person-person and person-object threshold are shown. The window size is kept constant at 400.

significantly higher than the ones of the third run. The number of time-alignment errors is similar, but many more ordering errors are made. This behavior is expected, because systems with lower window sizes are more sensitive than the ones with bigger windows.

| run number | | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| window size | | 120 | 120 | 400 | 400 |
| pers.-pers. thresh. | | 0.0001 | 0.875 | 0.05 | 0.375 |
| pers.-obj. thresh. | | 0.375 | 0.875 | 0.25 | 0.75 |
| pers. cluster | FRR | 72.3% | 24.8% | 75.4% | 55.4% |
| obj. cluster | FRR | 79.7% | 72.2% | 82.3% | 69.6% |
| activity | FRR | **54.3%** | 20.0% | **58.5%** | 28.7% |
| pers. cluster | WER | 114.4% | 67.2% | 56.2% | 61.3% |
| obj. cluster | WER | 95.8% | 39.0% | 53.7% | 63.2% |
| activity | WER | 93.8% | **49.6%** | 64.1% | **50.3%** |

Table 6.4: Group-activity layer results for selected parameter values. The best results are typed in bold letters.

# 7 Conclusion

A two layer system has been developed that gives an answer to the question: 'Who is doing what with whom and when?' The first layer can be implemented in many ways, the only requirement is the ability to calculate a probability of general interaction for person-person and person-object pairs. That also means that any available data could be used to solve the probability calculation task (see also section 7.3). We implemented a heuristic method for the first layer, that calculates the interaction probability based on the time the persons gaze at each other or the person gazes at the object. The second layer searches the best explanation for the earlier calculated probabilities. Valid explanations are composed of predefined template graphs, each of them represents one activity. In the following part, we discuss the results in section 7.1. Then the limitations of the used method are shown in section 7.2 and finally some future perspectives are given in section 7.3.

## 7.1 Discussion of the Results

Frame recognition rates of over 70% have been reached for person and object clustering. The best FRR for activity recognition is lower at around 50%, because it depends on both, person and object clustering. Good WERs below 50% for person/object clustering and about 66% for activity recognition can only be reached with window sizes greater then 200 frames, because otherwise the system is too sensitive to temporary interaction changes in the data. The person-person and person-object threshold have a big influence on the system sensitivity as well: low values lead to a sensitive system that adapts fast to new situations and the results show good frame recognition rates. However, the activity WER is in this case not optimal, because of the sensitivity of the system. For a good activity WER result, the thresholds must be higher, what also leads to worse frame recognition rate results. That means, we have two possibilities for optimizing the system parameters: toward a sensible, fast adapting system with a good time-alignment of the results or a lower sensible, slow adapting system with a good ordering of the results.

## 7.2 Limitations

Our system uses only the general interaction 'paying attention to' for recognizing activities. On the one hand, this simplification leads to a computational tractable problem and generalizes the approach to be applicable in many situations. On the other hand, information is lost and the result is only a coarse description of the situation, because it is generalized as well. For example: assume that our system detects monologue, which is one speaker interacting with each person of the audience. We cannot be sure that the detected speaker is really the speaker, because we only detect general undirected interactions. Another explanation would be, that all persons are talking at the same time to the same person. Admittedly, this situation is somehow unlikely because it is not polite to talk at the same time, but it is not impossible. We really detect that a group of persons is not interacting with each other, but interacting with another person (the assumed speaker). In an office scenario, this is likely a monologue. However, in other scenarios other explanations are likely. For example, the group could be only watching one person. This limitation effects especially groups of only two persons, because we cannot differentiate between monologue and discussion. To overcome this limitation, it would be possible to do further recognition steps, see also section 7.3.

Another limitation is, that the activity recognition is only based on one probability graph and not on several consecutive graphs. That means it is impossible to recognize activities that follow a special interaction pattern over time. An example for such an activity is 'shopping in the supermarket': first, on person interacts with several goods in several shelves (i.e. puts them in the basket) and then finally goes to the exit and interacts (i.e. pays) with the cashier. Our system can detect all consecutive sub-activities, but for detecting the overall activity 'shopping', further recognition steps would be needed.

## 7.3 Future Work

In the future, our system could be improved and extended in several ways. The interaction layer could be improved by learning the probability functions instead of using the predefined heuristics. The annotated data provides the information whether an interaction happens or not. Then we could learn the probability distribution of the VFOA over the time window for 'interaction' and 'no interaction'.

Also different implementations of the interaction layer are possible. The current implementation needs the VFOA as input and we are actually using the annotated VFOA. However, in the future, the system should be able to detect interactions automatically without using annotations made my humans. One possibility to achieve that would be to use automatically detected VFOA target probabilities. A promising system that could provide this data is explained in [17]. Another possibility would be to use statistical models for interaction detection (like HMMs or DBNs), that use the automatically generated information of the SmartControlRoom as feature vector (i.e. tracking, head pose, speech and gestures). Some initial experiments have been conducted with HMMs. For detecting the interaction probability between two persons, we use the feature vector $F_{pers.-pers.} = (d, v_1, v_2, \alpha_1, \alpha_2, s_1, s_2, d', v_1', v_2')$, where $d$ is the distance between the two persons, $v_1$ is the velocity of person 1, $\alpha_1$ is the head pose angle of person 1 relative to the line connecting the two persons, $s_1$ is the speech activity of person 1 (and respectively the same for person 2) and the last three positions are the time derivatives of $d$, $v_1$ and $v_2$. All values are relative to the person positions and thus are independent from the environment. The feature vectors for person-object interaction are the same, but reduced by the attributes of the second person: $F_{pers.-obj.} = (d, v_1, \alpha_1, s_1, d', v_1')$. HMMs for the cases 'interaction' and 'no interaction' are trained separately, therefore we train four HMMs in total: person-person interaction, person-person no interaction, person-object interaction and person-object no interaction. The feature vector contains information of one person-person or person-object pair, therefore each possible pair in the data set provides a training sequence. The sequence is divided into parts with interaction and without interaction according to the annotations (if an interaction happens or not is derived from the interaction structures of the annotated group-activities). The evaluation of the trained HMMs with the same sequences, that have been used for training, showed poor results. That means that the HMMs could not learn the sequences at all. However, with further improvement of the feature selection and the training, satisfying results might be achieved.

One disadvantage of the system is the poor computation performance of the group-activity layer for scenes with a large number of persons. However, the performance of the group-activity layer could be improved by using more sophisticated graph algorithms. For example, the connected components of the binary graph, that is deduced from the probability graph by doing hard decisions, could be detected first. Then each of the connected components could be searched separately. Also probabilistic optimization methods could be used that do not guarantee an optimal solution, but give a huge speedup.

Depending on the actual needed recognition task, the system could be extended in many ways. For example, some additional constraints could be checked: if a two person group-activity is recognized, then it could be checked if one person is speaking significantly more than the other and thus it is a monologue (and a discussion otherwise). Another possibility would be to use a third layer: as soon as the groups are known, each group can be examined by another classifier. For two people interacting, the system build in [10] could be used to recognize what exact kind of interaction is happening. The third layer could also analyze a sequence of template graphs and thus recognize activities with a special interaction pattern over time (like 'shopping' explained in section 7.2).

Finally, we can conclude that it is possible to build a system that is able to cluster several persons and objects into groups. In addition, it is able to assign each group the performed activity, out of the set of predefined basic group-activities. If more detailed information about the activities is needed, the system can be extended easily, because the groups are already known. The system has been evaluated with an office data set, but the same approach might be used for other situations. However, more research is needed with data of other situations to proof that point.

# List of Figures

# List of Tables

# Bibliography

[1] J. Aggarwal and M. Ryoo. Human Activity Analysis: A Review. *ACM Computing Surveys*, 2011.

[2] K. Bernardin, T. Gehrig, and R. Stiefelhagen. Multi-level particle filter fusion of features and cues for audio-visual person tracking. *Multimodal Technologies for Perception of Humans*, pages 70–81, 2009.

[3] P. Dai, H. Di, L. Dong, L. Tao, and G. Xu. Group interaction analysis in dynamic context. *IEEE transactions on systems, man, and cybernetics*, 39(1):34–42, Mar. 2009.

[4] H. Ekenel, Q. Jin, M. Fischer, and R. Stiefelhagen. ISL person identification systems in the clear 2007 evaluations. *Multimodal Technologies for Perception of Humans*, pages 256–265, 2009.

[5] J. Ijsselmuiden and R. Stiefelhagen. *Towards High-Level Human Activity Recognition through Computer Vision and Temporal Logic*, volume 6359 of *Lecture Notes in Computer Science*, pages 426–435. Springer, Karlsruhe, 2010.

[6] Y. Ivanov and A. Bobick. Recognition of visual activities and interactions by stochastic parsing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):852–872, 2002.

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

[8] N. Oliver, B. Rosario, and a.P. Pentland. A Bayesian computer vision system for modeling human interactions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):831–843, 2000.

[9] K. Otsuka, Y. Takemae, and J. Yamato. A probabilistic inference of multiparty-conversation structure based on Markov-switching models of gaze patterns, head directions, and utterances. In *Proceedings of the 7th international conference on Multimodal interfaces*, page 191, New York, New York, USA, 2005. ACM Press.

[10] S. Park and J. Aggarwal. A hierarchical bayesian network for event recognition of human actions and interactions. *Multimedia Systems*, 10(2):164–179, Aug. 2004.

[11] M. Ryoo and J. Aggarwal. Recognition of high-level group activities based on activities of individual members. In *Workshop on Motion and Video Computing*, pages 1–8. IEEE, 2008.

[12] A. Schick, F. van de Camp, J. Ijsselmuiden, and R. Stiefelhagen. Extending touch: towards interaction with large-scale surfaces. In *Proceedings of the International Conference on Interactive Tabletops and Surfaces*, number c, pages 117–124. ACM, 2009.

[13] Y. Sheikh, M. Sheikh, and M. Shah. Exploring the space of a human action. In *Tenth International Conference on Computer Vision*, volume 1, pages 144–149. IEEE, 2005.

[14] H. Soltau, F. Metze, C. Fügen, and A. Waibel. A one-pass decoder based on polymorphic linguistic context assignment. In *Automatic Speech Recognition and Understanding Workshop*, pages 214–217. IEEE, 2001.

[15] R. Vertegaal, R. Slagter, G. van Der Veer, and A. Nijholt. Eye gaze patterns in conversations: there is more to conversational agents than meets the eyes. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 301–308. ACM, 2001.

[16] M. Voit and R. Stiefelhagen. Deducing the visual focus of attention from head pose estimation in dynamic multi-view meeting scenarios. In *Proceedings of the 10th international conference on Multimodal interfaces*, pages 173–180, New York, USA, 2008. ACM.

[17] M. Voit and R. Stiefelhagen. 3D user-perspective, voxel-based estimation of visual focus of attention in dynamic meeting scenarios. In *International Conference on Multimodal Interfaces and the Workshop on Machine Learning for Multimodal Interaction*, page 51. ACM, 2010.

[18] C. Wojek, K. Nickel, and R. Stiefelhagen. Activity recognition and room-level tracking in an office environment. In *International Conference on Multisensor Fusion and Integration for Intelligent Systems*, number section IV, pages 25–30. IEEE, Sept. 2006.

[19] D. Zhang, D. Gatica-Perez, S. Bengio, and I. McCowan. Modeling individual and group actions in meetings with layered HMMs. *IEEE Transactions on Multimedia*, 8(3):509–520, June 2006.