EVALUATION OF TRACKING AND RECOGNITION METHODS

Ing. Jaroslav KADLEC, Doctoral Degreee Programme (2)
Dept. of Computer Graphics, FIT, BUT
E-mail: kadlecj@fit.vutbr.cz

Ing. Stanislav SUMEC, Doctoral Degreee Programme (4)
Dept. of Computer Graphics, FIT, BUT
E-mail: sumec@fit.vutbr.cz

Ing. Igor POTÚČEK, Doctoral Degreee Programme (3)
Dept. of Computer Graphics, FIT, BUT
E-mail: potucek@fit.vutbr.cz

Supervised by Dr. Pavel Zemčík

ABSTRACT

This paper describes possibilities and features of manual annotation for evaluation of audio-video tracking and recognition methods. The "Event editor" software tool which was developed for the annotation purposes is also presented.

1 INTRODUCTION

When an algorithm or application is being developed, the crucial requirement is to measure its performance and quality (correctness) of its results. The purpose of this paper is to present a software tool for annotation of the video sequences that is created in order to provide "ground truth" for evaluation of the algorithms for head detection, face position measurement, and face feature parameterization.

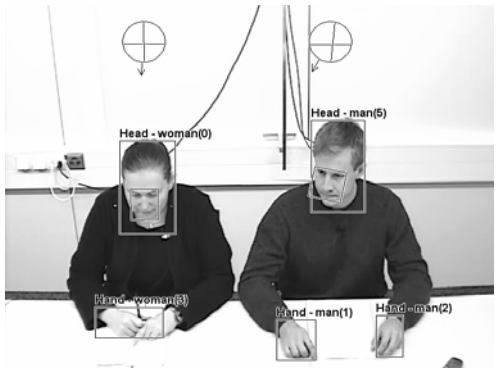

Fig. 1: *Recognized human body parts*

The above mentioned algorithms produce information about head size, position, rotation, etc. (fig.1). These algorithms require evaluation in roder to verify the accuracy, robustness, etc. This paper also discusses the annotation process and the way of using the obtained annotation data.

2 ANNOTATIONS

Two options exist, how to determine whether the algorithm fulfills the specified criteria. First option is to manually feed the algorithm with the images and set true or false values depending on whether the algorithm succeeds or fails. This option is labour-intensive and each future test requires lots of time. The second option is to manually annotate the available data and compare the algorithm generated data with the manually annotated data. This results in much faster testing.

Although creation of annotated data is time consuming process, it can boost future algorithm testing and even development and testing of other algorithms working with the same data. That is why the annotation scheme must be specified precisely.

2.1 ANNOTATION SCHEME

The annotation scheme describes the annotated data. Basically, it contains and defines information that should be annotated. In the presented case, bounding rectangle coordinates are usually used especially for head or hands annotation purposes The bounding rectangles defines the position of the body parts in the image. Other important information can be the visibility of the object of interest. The annotation, however, is not as simple as it might seem. Various problems occur, for example: “How to treat the partially covered head (face)?”, “What is the partially covered face?”, etc. All these bits of information are required for proper annotation and the annotation scheme must contain and resolve the highest possible proportion of the “complications” that can occur during the annotation process. It is imperative that a proper annotation scheme must be prepared before the annotation starts at all. The absence of the information, how to treat the singular cases, can lead into misclassification and wrong results of the developed algorithms.

The following example shows a part of an annotation scheme designed for annotation of the head position and head movement:

Head movement:

- *nodding*
- *shaking*
- *turning left*
- *turning right*
- *turning up*
- *turning down*
- *default: nogesture*

Annotation details:

- *nogesture: this entry is used whenever no other label can be used.*
- *nodding / shaking: beginning until end of the movement of the head*
- *turning left / right / up / down: from the view of the person; only the time of the movement of the head. Annotated is the movement of the head into the specified direction (e.g. from looking left to looking ahead » turning right; from looking ahead to looking right » turning right)*

The above scheme defines exactly what to annotate when the head is moving. The default “nogesture” represents situation, when the head does not move. The annotation details describe the movement key-words and their meaning. Such explanation is important as turning can be represented in several ways. All turning, nodding, and “nogesture” are state values which mean the head movement at all times.

2.2 ANNOTATION WORK

An annotator is a person that performs the annotation. Several people are usually working on the annotation because is the annotation process in highly time consuming. The annotators are required to do the job as precisely as possible because any mistakes in the annotation can lead in false comparison of algorithm generated data and, in the worst case, if the data is used for machine learning, the learning process canbe adversely affected.

Several annotators can be assigned to annotation of the the same data and can be doing the same annotation. The reason is the that the inaccuracies, that can be made by some of the annotators, can be eliminated this way. Three or more annotations of the same data can be merged and compared for optimal data annotation. An example of annotation can be head position annotation where annotators are tagging two points in every video frame. Precise point designation can last few seconds. Two point tagging lasts around 5 seconds for each frame. An annotator spends almost two minutes on one second of 25 fps video annotation, which is (in time per one hour of video) extremely large figure and that is why other approach should be suggested. A good source of the speed increase in annotation is the “frame skipping” technique - tagging for example only the key frames. (Key frame is defined as video image in specified time interval – e.g. each tenth frame.) The key frame can also seen as each video frame where with a major change comparing to the predeceding frame. The frames in between the key frames are interpolated by some linear or other suitable function. This approach shortens annotation time but increases inaccuracy in frames between key frames. These inaccuracies are, however, not necessarily a problem in all applications and that is why the key frame annotation (“frame skipping”) is frequently used.

All annotators having a properly designed annotation scheme can work separately with no need of further consulting. However, frequent meetings during first time annotations are required because new tags can be defined just as the need occurs based on the annotated data and annotation scheme can be updated to support all variations.

2.3 ANNOTATION OUTPUT

The easily readable annotation output is often required and that is why various text file formats have been developed. Most useful and also well known and standardized format is the XML file format. This self descriptive format contains all the information about the annotated tags, their parameters, and their structures. Many tools supporting this format exist. The annotated data can be clearly understood and easily parsed by generally available XML parsing tools. The following example is an XML file with head position annotation. The file contains only head position annotation which is represented by number 1. The XML file also specifies a file from which the annotations were done and whether the annotation was done frame by frame or second by second. Finally, the parameters describing head position and other properties, are included.

```
<AVEvents>
  <Events><Name ID="1">Head position</Name></Events>
  <File>
    <Source Camera="1">heads.avi</Source>
    <TimeFormat>Frames</TimeFormat>
    <Event>
      <ID>1</ID>
      <Time>1</Time>
      <Parameters Object="1" CenterX="184" CenterY="262" Person="A"/>
    </Event>
  </File>
</AVEvents>
```

2.4 EVENT EDITOR

A tool called "Event editor" was developed and used for annotating of the audio and the video files can be seen in fig.2. This software tool is able to show and play several sources simultaneously. The records acquired with several cameras and microphones can be annotated simultaneously. This is a big advantage comparing to other available annotation tools, which are capable to use only one source at the time.

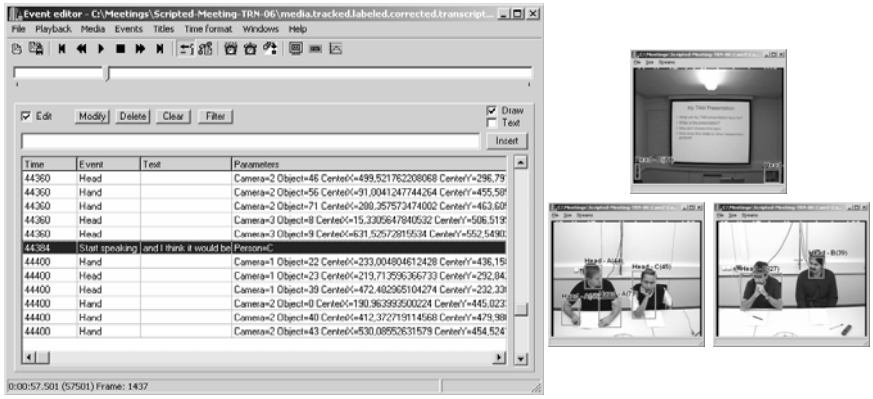

Fig. 2: *Main window and source windows with visual events*

The annotation scheme is represented with simple events in the presented tool. The events describe what happens in the given time e.g. a change of some state. Also the shapes, which are visible in the source video, can be represented with events because the events can hold additional parameters as coordinates. Similar type of the events, e.g. events representing different values of the same state, may be grouped into named groups which enable to use more complex annotation schemes. An example of representation of certain annotation schema can be seen in fig. 3. Two groups of events intended for presentation of various states of a person being annotated are used in this scheme.

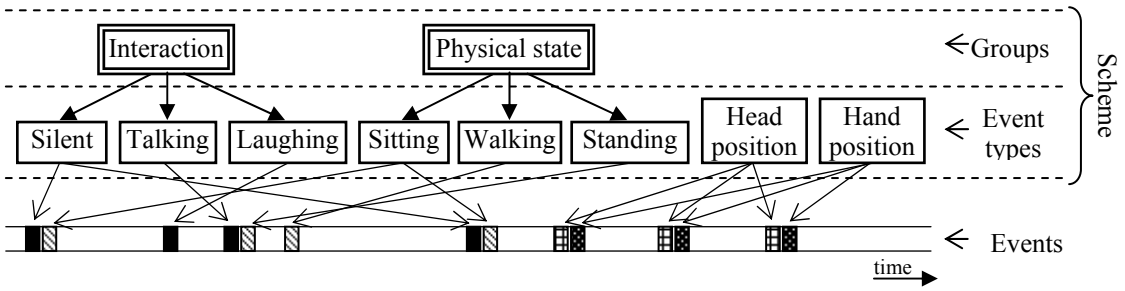

Fig. 3: *Example of annotation with annotation schema*

The scheme contains also two other events (not belong to any group), which describe positions of person's body parts. The events are inserted into the annotation using the hot keys, which can be assigned to the defined events and with mouse in case of the visual events. The standard editing features, such as modifying, deleting, filtering etc. are implemented, too.

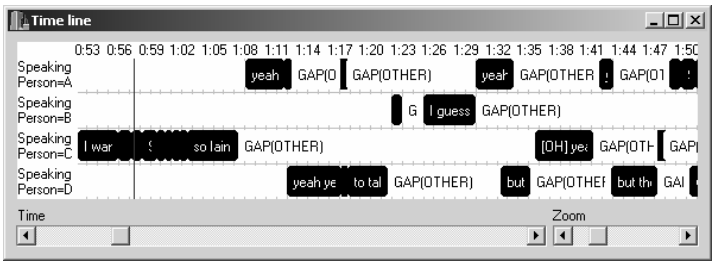

Fig. 4: *Timeline representation*

An important factor for annotators is a form of presentation of the annotated data. Inserted events are primarily displayed in a list; a timeline representation (see fig. 4) of the state events is also available, and events describing continuous values can be displayed in a graph. Finally, visual events are directly drawn into a window with the picture of the source video. An output is stored into XML file that can be easily transformed into the desired format.

3 CONCLUSION

Annotations present a complex problem and it should not be ignored because a proper annotation can speed up algorithm development. To speed up annotation process, more than three annotators are highly recommended for one annotated file in order to achieve precise annotation (based on the size of annotated data). An XML output is advised as it is simple both to understand and parse. The acquired annotations will serve for accuracy and reliability evaluation of our developed algorithms.

ACKNOWLEDGEMENT

This work was partially supported by EC IST project Augmented Multi-party Interaction (AMI), No.~506811.

REFERENCES

[1] Rienks, R. J., Reidsma, D.: Meeting Annotation: A Framework for Corpus Based Research on Human-Human Interaction, MLMI'04, Martigny, Switzerland, 2004

[2] Carletta, J., Kilgour, J., O'Donnell, T., Evert, S., and Voormann, H.: The NITE Object Model Library for Handling Structured Linguistic Annotation on Multimodal Data Sets, Proceedings of the EACL Workshop on Language Technology and the Semantic, 2003

[3] Shriberg, E., Dhillon, R., Bhagat, S., Ang, J., Carvey, H.: The ICSI Meeting Recorder Dialog Act (MRDA) Corpus, Proceedings of the HLT-NAACL SIGDIAL Workshop, April 2004, Boston

[4] Reiter, S.: Annotation scheme for gestures and individual actions, Munich Technical University, AMI-WP3 document, November, 2004