A Framework for the Acquisition of Multimodal Human-Robot Interaction Data Sets with a Whole-System Perspective

Johannes Wienke, David Klotz, Sebastian Wrede

Research Institute for Cognition and Robotics (CoR-Lab), Bielefeld University
Universitätsstraße 25, 33615 Bielefeld, Germany
{jwienke,dklotz,swrede}@cor-lab.uni-bielefeld.de

Abstract

In this work we present a conceptual framework for the creation of multimodal data sets which combine human-robot interaction with system-level data from the robot platform. The framework is based on the assumption that perception, interaction modeling and system integration need to be treated jointly in order to improve human-robot interaction capabilities of current robots. To demonstrate the feasibility of the framework, we describe how it has been realized for the recording of a data set with the humanoid robot NAO.

1. Introduction

Improving capabilities of robots for interacting with people is a challenging task which needs to be addressed from various perspectives. Besides models of human-human interaction which guide the development of computational models and realizing software, sufficient perception abilities for people and the scene are required. To cope with the challenges and use opportunities of a fully integrated robot system, these challenges cannot be solved separately. Data sets incorporating such a *whole-system perspective* are required to develop integrated solutions with repeatable and realistic conditions, e.g. for benchmarking (Lohse and others, 2009). Creating such data sets is a complex and time-consuming task. In this work we present a conceptual framework for the creation of these data sets. Moreover, we demonstrate the suitability of the framework by explaining how a real data set involving the humanoid robot NAO was created. This includes the description of chosen technical solutions and lessons learned. We begin with a description of the scenario for later references and examples.

2. Scenario and Required Data

The scenario is part of the activities in a collaborative research project[1], which tries to improve the abilities of a robot interacting with a group of people through audio-visual integration. Here, the setting of a small vernissage where which visitors are guided by the humanoid robot NAO was chosen. It is inspired by (Pitsch and others, 2011). More detailed, naive participants entered a recording room in pairs and were greeted. Afterwards, the robot presented several paintings in the room using speech and matching gestures. These explanations included pauses intended to elicit comments by the visitors and also gave them the chance to tell the robot if they wanted to hear further explanations at specific points. After the explanations, the robot proceeded with a quiz asking several questions about the paintings and more general topics. During the recordings, speech and movements of the robot were remotely controlled by a human operator. This fact was unknown to the participants.

[1]HUMAVIPS, cf. `http://www.humavips.eu`

In order to address the research questions like addressee detection and visual focus of attention (VFOA), several requirements existed. First, absolute positions and orientation of each participant (specifically the head), the robot and all paintings needed to be known for being able to analyze the VFOA. For this task and for being able to detect addressees, annotations based on the orientation of heads and facial reactions were required. Hence, close video recordings of the faces are required. The same is true for spoken words of all participants. Apart from such external cues, internal sensory and status information of the robot are necessary in order to develop algorithms for a robotic platform that are able to cope with real environment restrictions. For instance, these information include CPU load, kinematics or odometry. Having these information available retains the ability of integrating them into developed algorithms.

3. Challenges in Creating Data Sets

One of the primary issues when recording multimodal data sets is the *synchronization* of all modalities. For cameras or audio streams this could be done in a post-processing phase. However, synchronization is much more complicated with modalities which are less intuitive to observe for humans like robot internal states. Also it induces additional effort required in the post-processing phase. Hence, one challenge is the *reduction of required post-processing* already through the recording setup. This influences the choice of devices. Moreover, it requires *automation* and *validation*

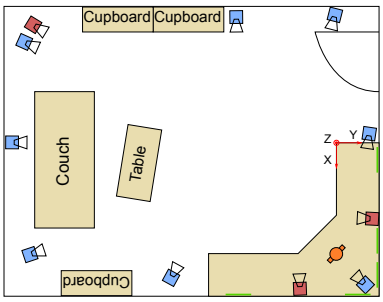

Figure 1: Qualitative overview of the recording room. Orange: NAO, cameras: blue – Vicon, red – HD, green lines: paintings, red: Vicon coordinate system

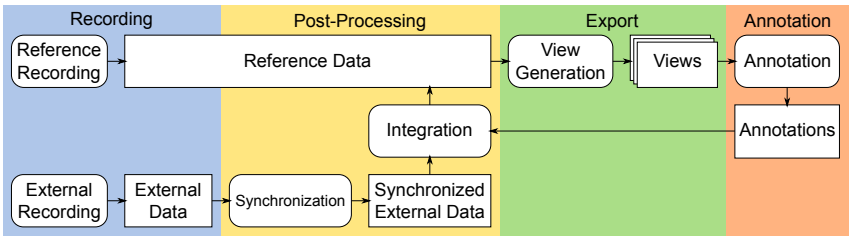

Figure 2: Schematic overview of the proposed framework. Rounded boxes indicate performed activities, squares represent generated data. Different workflow phases while creating the data set are indicated through the background colors.

possibilities during the recording time in order to prevent errors in the recorded data. This also concerns the *calibration* of recording devices. By e.g. calibrating all cameras with respect to a motion capturing system unplanned opportunities to use the data set are preserved.

From the whole-system perspective an important requirement is that the recorded data allows a smooth *application in the integrated system*. This means that developed system components can be used without changing their interfaces. While this restricts the recording tools and formats it is still important that the data set can be used without the system integration. Thus *export* facilities to common formats are necessary, e.g. for video.

Finally, for the annotation phase, *established tools* should be reusable and benefits for the annotation should be gained from the system modalities. To efficiently evaluate the system based on the data set the *availability of annotations for integrated components* is essential.

4. The Whole-System Framework

In order to address the aforementioned challenges we propose to directly use the communication system of the robot (i.e. the middleware) as the primary tool and format for data set recordings (data recorded in this way will be called *reference data*). By providing record and replay solutions tightly integrated to the middleware layer, system-internal data can be captured easily and in the native format of the system, as required for whole-system analysis. Moreover, we propose that additional recording devices are either captured directly with this system or their data is later integrated into it. This integration is also the proposed method for secondary data like annotations. Replaying this data enables the *application of the data set in the integrated system* while ensuring the *availability of annotations* without depending on a concrete annotation tool. This means unchanged system component can be used online on the data set with their usual inputs like audio and vision and also have the annotations available in the architecture. The *synchronized replay* in this case is a generic problem and needs to be solved only once, hence reducing parts of the *post-processing effort*. Fig. 2 visualizes the proposed process of data management.

According to this process, external recording devices should be selected in a way that they can be recorded with the system infrastructure. This is for instance the case with network cameras, which can be recorded directly using the middleware layer of the system. In cases where this is not possible, the implementation of the framework in Section 5. demonstrates how to *automate* the *synchronization* and conversion of the external data to the reference data.

To address the requirement of *exporting* parts of the data set to common formats we propose a *view-based approach* on the reference data. Views are selective immutable exports of (parts of) the data sets. An exemplary use case of this approach can be the generation of a project for an annotation tool.

5. Applying the Whole-System Approach for a Multimodal HRI Data Set

We will now describe how the framework has been instantiated to record the scenario introduced in Section 2.. Experimental robotics applications in the project are realized on the humanoid robot NAO[2]. Integration is performed using an event-based middleware termed RSB (Wienke and Wrede, 2011) which allows full introspection of the internals of the robotics system. All information in the system is continuously sent over a logically unified bus, which can be composed of different transport layers (e.g., network or in-process communication) within events as the basic unit of communication. Each event contains data like sensor reading and a set of meta data including accurate timing information. Based on the introspection support, RSB includes a mechanism (RSBag) to record and replay the events stream with original timing and hence realizes the acquisition of the reference data. All internal data from NAO as well as the control commands for remote operation have been recorded using this mechanisms without needing modifications of the system. As RSB and the record and replay mechanism are generic, this architecture can be reused for other data set acquisition activities.

Besides this system-level data we utilized a Vicon motion capturing system[3] to acquire ground truth position data of participants and the robot, installed 3 HD cameras in the recording room to provide views for the annotation of addressees and VFOA, and equipped each participant with a close talk wireless microphone. Due to restricted interfaces, the Vicon system and the HD cameras could not be recorded based on the RSB system, so they form a test case for the later integration into the reference data during the post-processing. In contrast, the close talk microphones were attached to one of the recording computers and hence could be recorded inside the RSB architecture. As the internal clocks of all computers in the distributed system were synchronized using NTP, all data recorded using RSB is synchronized without manual work. An overview of the

[2] http://aldebaran-robotics.com
[3] http://www.vicon.com

Type	Specification
NAO video	Monocular uncompressed frames, VGA, variable frame rate (∼15 fps mean), YUV422 color mode.
NAO audio	4 channels, 48000 Hz, 16 bit signed.
NAO odometry	est. 2D location of robot body
NAO proprioception	Joint angles, stiffness, last command value, temperature
NAO system	CPU, memory, battery, modules
Demo system and control	Wizard commands, internal events for speech and gesture production
close talk mics	4 channels, 44100 Hz, 24 bit signed
Vicon	6D pose for people and NAO, 100 Hz
External HD Cameras	3 perspectives, 1920×1080 pixels, 25 Hz. 5.1 channel sound, 48000 Hz.

Table 1: Recorded data. Italic: recorded using RSB.

recording room and the placement of recording devices is depicted in Fig. 1. Table 1 lists all recorded data streams.
Besides the actual recording of the scenario with different participants, *calibration* sequences have been recorded. From these runs a special Vicon marker has to be extracted at certain points in time to find out the locations of paintings and several other objects in the scene. Even though it would be possible to extract these positions manually, we increased the *automation* in the post-processing phase by presenting a clapperboard which was markered for the Vicon system each time the marker for measuring positions was finally placed at the desired position.
Besides this calibration aspect, we recorded a checkerboard pattern for all cameras (including NAO) so that distortions can be calibrated. Moreover, a special Vicon subject with 4 tracked markers has been presented to the HD cameras and the Vicon at once. Hence, the location of each HD camera in the Vicon coordinate system can be computed.

5.1. Post-Processing

As previously mentioned, the Vicon system and the HD cameras could not be recorded directly using RSB. For Vicon, in principle, an online API exists so that special software can be written which receives the Vicon measurements and sends them using RSB. However, using this approach no tracking errors can be corrected afterwards using the Vicon Nexus software. Moreover, the online API up to our knowledge lacks accurate timing information. For these reasons we decided to use the internal recording capabilities of Vicon which allows manual error correction afterwards but also requires a manual processing, export and synchronization with the remaining recordings. The HD cameras were used as no cameras with Ethernet connection were available for the recording and the high resolution is a requirement for the annotation.
To synchronize the videos from the HD cameras we calculate the cross-correlation peak of the cameras' audio channels with a reference audio channel recorded in RSB where exact timing information are available. For this purpose we used the sound recorded by NAO's microphones as a good cross-correlation can be expected because NAO and the external cameras were always in the same room. This was not the case for the close talk microphones which were carried around by participants and could be muted. Praat (Boersma and Weenink, 2011) was used to realize the cross-correlation calculation. Based on the correlation peak we deduced the offset of the external videos with respect to the audio from NAO, which in turn allowed us to compute the start time of the external videos in the RSB time frame.
To generate synchronized results from the Vicon system several steps were necessary. First, each recorded trial needs to be processed in Vicon which might involve manual labeling in situations where the Vicon system could not track its artificial markers sufficiently well. After this processing, an export of the tracking results was performed using the easy to parse CSV (comma-separated values) export format. Vicon uses a fixed frame rate, in our case 100 Hz. Hence, it is sufficient to know the RSBag time of one Vicon frame per trial. For this purpose we implemented a clapperboard detection which extracts the Vicon frame for the moment the clapperboard was shut. The detection is based on a sliding window approach on the Euclidean distance of the two clapperboard parts. Unfortunately, we did not find an automatic way to relate this Vicon frame to RSBag, as no easy detection of the clapperboard in the recorded modalities using this system was possible. Hence, we manually searched for the time of the clapperboard in an export of the audio recordings using the open source audio editor Audacity[4] which allows to display the exact sample count for the cursor. This provides a high precision and simplifies the calculation of the RSBag timestamp. The clapperboard detection was also used to automatically obtain the positions of paintings and other interesting locations from the calibration runs.
Both, Vicon exports to CSV and HD video can be integrated into the RSBag files by creating RSB events containing the data with the correct timestamps provided by the aforementioned synchronization procedures.

5.2. Annotation and View-based Access

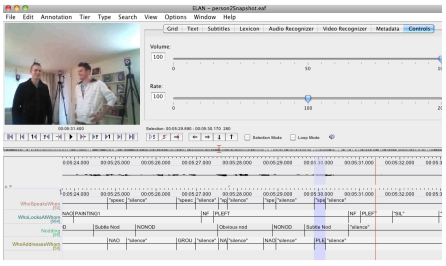

Figure 3: Export of one synchronized trial to an ELAN project with added annotations.

We decided to use the well-established annotation tool ELAN (Wittenburg and others, 2006) for the data set. As ELAN is not capable of processing the RSBag format of the data set, we needed to provide an ELAN-view on it. To *automate* the creation of this task, we have developed a script which creates a view on the data set to enable the annotation in ELAN. It uses the synchronized data created during the post-processing, converts video and audio to file

[4] `http://audacity.sourceforge.net/`

formats compatible with ELAN using ffmpeg[5], and automatically creates a project file to load in ELAN (cf. Fig. 3). The annotations created in ELAN will be integrated into the RSBag files to ensure the *availability of annotations for integrated components*.

6. Related Data Set Acquisition Approaches

To the best of our knowledge, no work exists which provides a conceptual framework on how to capture data sets which contain HRI data in a whole-system manner, especially with the aspect of direct replay possibilities in the system architecture.

A closely related scenario with comparable modalities has been presented in (Green and others, 2006). However, it only covers a single sensor from the system and neglects the remaining communication in the system, especially the commands for remote control of the robot. Also, no generic approach for recording has been presented.

Regarding data sets without robotics involved, (Luz et al., 2006) describes the acquisition of computer-mediated meetings through a distributed system. Their system streams data over the network using RTP and also contains internal data of a collaborative editor software. However, there is no unique recording format. (Roggen and others, 2010) describes the implementation of an acquisition system for corpora based on a largely distributed sensor system. Their recording system provides no intrinsic mechanism for synchronized recording. Instead manual inspection is proposed for this task. A recording architecture for meeting corpora is presented in (Banerjee and others, 2004) with comparable aims for extensibility as in our approach. The presented system uses NTP for time synchronization and has a comparable notion of timed events. Besides the recording aspects, an approach to collaboratively aggregate more data in data sets is presented using a central server where the data is uploaded. The collaboration idea is further devised in (Chervenak and others, 2000). Our current approach does not cover such a level of dissemination and collaboration, even though this is highly required to facilitate research.

7. Conclusion and Outlook

The applicability of our devised generic framework for data set acquisition has been demonstrated to a large extend on a concrete implementation for the vernissage scenario. However, further validation of the concept with respect to the annotation as incremental addition of other data needs to be performed. Ultimately, the integration of our concept with the ideas presented in (Chervenak and others, 2000) is required to cover the whole workflow from recording to dissemination.

For the described implementation several optimizations are possible. The current integration of the Vicon system needs to be improved to provide further automation and reduce the post-processing effort. External cameras could be replaced with networked cameras. However, the current integration solutions are not tailored to the specific scenario and can be reused for other recordings. In the future we will continue to evaluate the application of further views for the export of the data set. A prototype we have developed indicates that with a web-based interface potential users of the data set can easily browse the contents and request an appropriate synchronized view from the web-server without needing to install specialized tools.

Summing up, the framework provides a structured approach for the acquisition of data sets which include system level information. As the situational context is partially determined by the system, the frameworks helps in generating a better view on these context aspects in data sets.

[5] `http://ffmpeg.org`

8. Acknowledgments

This work was funded by the European FP7 project HUMAVIPS, theme ICT-2009.2.1, grant no. 247525. Thanks to Raphaela Gehle, Michael Götting, Dinesh Jayagopi, Stefan Krüger, Phillip Lücking, Jan Moringen, Karola Pitsch, Lars Schillingmann, Jens-Christian Seele, Samira Sheikhi, and all participants.

9. References

Satanjeev Banerjee et al. 2004. Creating multi-modal, user-centric records of meetings with the carnegie mellon meeting recorder architecture. In *Proc. of ICASSP'04, the Int. Conf. on Acoustics, Speech, and Signal Processing, Meeting Recognition Workshop*, Montreal, Canada.

Paul Boersma and David Weenink. 2011. Praat: doing phonetics by computer (version 5.3.03). `http://www.praat.org`. Computer program.

Ann Chervenak et al. 2000. The data grid: Towards an architecture for the distributed management and analysis of large scientific datasets. *Journal of Network and Computer Applications*, 23(3).

Anders Green et al. 2006. Developing a contextualized multimodal corpus for human-robot interaction. In *Proc. of the 5th Int. Conf. on Language Resources and Evaluation (LREC 2006)*, Genoa, Italy.

Manja Lohse et al. 2009. Systemic Interaction Analysis (SInA) in HRI. In *Proc. Int. Conf. Human-Robot Interaction*.

Saturnino Luz, Matt-Mouley Bouamrane, and Masood Masoodian. 2006. Gathering a corpus of multimodal computer-mediated meetings with focus on text and audio interaction. In *Proc. 5th Int. Conf. on Language Resources and Evaluation (LREC 2006)*, Genoa, Italy.

Karola Pitsch et al. 2011. Attitude of german museum visitors towards an interactive art guide robot. In *Proc. of the 6th int. conf. on Human-robot interaction*, HRI '11, pages 227–228, New York, NY, USA. ACM.

Daniel Roggen et al. 2010. Collecting complex activity datasets in highly rich networked sensor environments. In *2010 Seventh Int. Conf. on Networked Sensing Systems (INSS)*, number 00. IEEE.

Johannes Wienke and Sebastian Wrede. 2011. A middleware for collaborative research in experimental robotics. In *2011 IEEE/SICE Int. Symposium on System Integration, SII2011*, Kyoto, Japan. IEEE, IEEE.

P. Wittenburg et al. 2006. Elan: a professional framework for multimodality research. In *Proc. of LREC 2006, Fifth Int. Conf. on Language Resources and Evaluation*.