2010 IEEE International Conference on Robotics and Automation
Anchorage Convention District
May 3-8, 2010, Anchorage, Alaska, USA

Hierarchical, Knowledge-oriented Opto-acoustic Scene Analysis for Humanoid Robots and Man-machine Interaction

T. Machmer, A. Swerdlow, B. Kühn
Karlsruhe Institute of Technology (KIT)
Institute for Anthropomatics (IFA)
Adenauerring 4, Geb. 50.21
76131 Karlsruhe, Germany
{swerdlow, kuehn}@kit.edu

K. Kroschel
Fraunhofer Institute of Optronics, System
Technologies and Image Exploitation (IOSB)
Fraunhoferstr. 1
76131 Karlsruhe, Germany

Abstract— **The opto-acoustic scene analysis is an extremely important as well as a challenging task for a humanoid robot. By the opto-acoustic scene analysis, the guided and autonomous exploration of the environment by means of acoustic and/or visual perception is meant. On the one hand, the perception ability is necessary to interact with humans in a humanoid way. On the other hand, the proximity of the robot has to be analyzed continuously, in order to enable the robot to fulfill its everyday tasks. Thereby, the greatest challenge lies in the wide variety of different perception tasks, e.g. detection, tracking, and identification of persons and different types of objects. This leads to the need of adapted, both, task- and context-dependent perception modules with specific requirements and abilities.**

Taking these considerations into account, the paper presents a hierarchical, knowledge-oriented concept of a framework for the opto-acoustic scene analysis. The focus of the work is put on formal conditions on one side and the practical realization of a real-time system on the other side. The proposed framework is modular structured and consists of a number of specialized perception modules. To reflect the knowledge-based structure of the framework, an object-oriented environment model is used for continuous inserting, updating and removing the information about the proximity of the robot.

Besides the task of analyzing the scene with the reference to already known objects (and persons[1]), the proposed concept enables the robot to explore a (partially) unknown environment, with the focus on the creation of multimodal signatures for unknown objects and persons. These signatures are used to build an unique representation of the explored objects and enable the robot to recognize them at a later time.

Index Terms— **Opto-acoustic scene analysis, knowledge-oriented exploration of known and unknown objects.**

I. INTRODUCTION

The opto-acoustic scene analysis is an extremely important as well as a challenging task for a humanoid robot. Related research work is done in every project, which has to handle such kind of tasks, for example OpenHRP (Open Architecture Humanoid Robotics Platform) [5] or ASIMO (Advanced Step in Innovative Mobility) [7], to name just a few.

Since the perception ability is necessary for the robot to interact with humans in a humanoid way and to fulfill its everyday tasks, the entire proximity of the robot has to be continuously analyzed and explored. However, the process of the exploration cannot be restricted to few independent tasks like "follow the person" or "recognize the object". It consists rather of a wide variety of different, partly cross-linked perception tasks (for example, "identify the blue object on the left table"). Therefore, both, task- and context-dependent perception modules are required. These modules have specialized perception abilities (e.g. for detection, tracking, or identification of persons and different types of objects). Furthermore, each module can have specific requirements, which have to be fulfilled prior to its execution. For example, a module for the estimation of the height of a person does not require unprocessed video stream, but the output data of the tracking module with 3-d positions of the detected object as well as the knowledge from the environment model that the detected object is really a person.

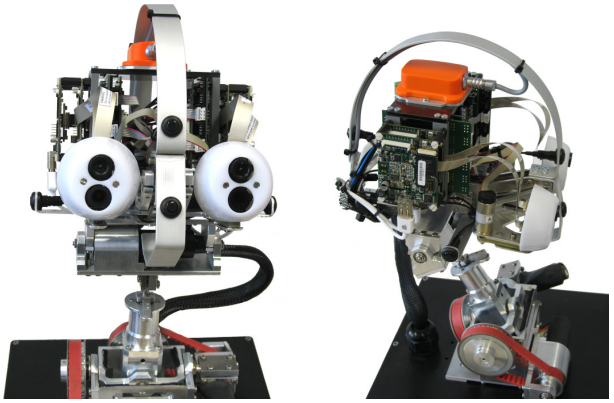
The multimodal scene analysis intends to integrate all perception modules in a consistent and modular framework applicable on a humanoid robot. In doing so, among others, two goals are to be considered: the choice of the most appropriate module for a current task as well as the consideration of possible dependencies of perception modules between each other. As a logical consequence, in order to coordinate the variety of perception modules and their abilities, a perception-oriented framework becomes obligatory.

To respond to the above mentioned challenges, the framework for the opto-acoustic scene analysis has been developed in our research group in the last years. This paper describes the proposed framework and introduces an experimental, real-time capable realization in form of the OPASCA (OPto-Acoustic SCene Analysis) system [8].

The proposed framework and implementation has been developed within the scope of the Karlsruhe humanoid research project SFB 588 [12] and is applicable on the Karlsruhe Humanoid Head [1], shown in Fig. 1.

This paper is organized as follows: In Section II, the proposed hierarchical approach for the object exploration is introduced. Hereby, the object-oriented modeling of collected information and handling of new and known objects are described. Some important aspects regarding the exploration of unknown objects are given separately in Section III. Section IV presents the system architecture of the developed framework with focus on both, data flow and knowledge orientation of different kinds of perception modules, as well as their handling within the framework. In Section

[1]It should be noticed, that in some cases, persons and different kinds of everyday objects like kitchen appliances, cups, books etc. are summarized as "objects".



(a) front view (b) lateral view

Fig. 1. Head of the humanoid robot ARMAR-III.

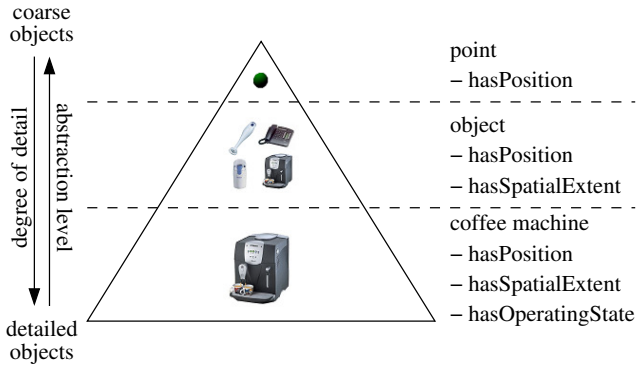


Fig. 2. Abstraction levels of the object COFFEE MACHINE in the environment model.

V, an exemplary real-time implementation of the proposed framework in form of the OPASCA system for the opto-acoustic scene analysis is described. Finally, in Section VI, a concluding summary is given and future work is described.

II. HIERARCHICAL EXPLORATION APPROACH

In this section, our approach for the hierarchical exploration of objects in the environment of the robot is presented. Hereby, the object-oriented modeling of collected information as well as the phases of the object exploration are described.

A. Object-oriented environment model

Like a human being, the humanoid robot needs a memory capability, which is constantly used for acquiring, saving, and recalling information about the robot's environment. Since every human is able to classify the world into objects and relationships and also tag them, e.g. by adding attributes, the robot's environment model needs similar capabilities. Thereby, the information may be provided by sensors of the robot, but may also result from knowledge generated earlier, or from the completely predefined a-priori knowledge.

Within the scope of the humanoid research project SFB 588, an approach for such an environment model was formally proposed in [4]. Taking those considerations into account, an environment model was developed and implemented. One of the core properties of the environment model is the object-oriented knowledge handling, which allows to set up a hierarchical representation of an object on different abstraction levels. These levels are defined by the degree of detail expressed by the attributes of the object. The more detailed they are, the lower the abstraction level is. Exemplary, the corresponding relations are shown for the object COFFEE MACHINE in Fig. 2.

New information, which is acquired by the sensors of the robot or is given by a human guide, may result in adding new attributes to already defined objects or in changing values of them. As a consequence of generating a new attribute, the abstraction level may change. The idea of different abstraction levels can be visualized by a class hierarchy based on inheritance, similar to object-oriented programming languages. All objects within the environment model are represented as instances of the basic class POINT. Depending on their associated characteristics, the individual instances can be specialized, e.g. in form of the class OBJECT. It inherits the spatial position attribute *hasPosition* from the POINT, but in addition, it might have some class specific attributes like *hasSpatialExtent* or *hasOperatingState*.

However, the availability of attributes always depends on the specialization of the class POINT. While the attribute *hasHeight* can be assigned to the class POINT specialized as a PERSON, the knowledge property *hasFillingLevel* can be used to specify the instance of the class POINT specialized as CUP.

Another important ability of the proposed environment model is the fact that each information attribute in the model contains a statement about its uncertainty, which quantifies the quality of the information. For example, the information provided by the sensors of the robot contains general measurement uncertainties, in contrast to predefined a-priori knowledge, which is usually very reliable. In so doing, each information in the model is accompanied by an uncertainty statement in a probability notion interpreted as a degree-of-belief (DoB). A detailed description of the DoB theory can be found in [10].

B. Object exploration

The exploration of the robot's environment is based on the perception of known and unknown objects, of which it consists. As soon as an object is detected by the sensors of the robot, the object exploration starts automatically. Subsequently, the corresponding information will be added to the environment model (see also [4]).

The life cycle of an object in the environment model is roughly characterized by three phases:

1) Detection and instantiation: In the case that new information gathered by the sensors of the robot cannot be associated with an already existing object in the environment model, a new object is instantiated in the environment model of the robot.

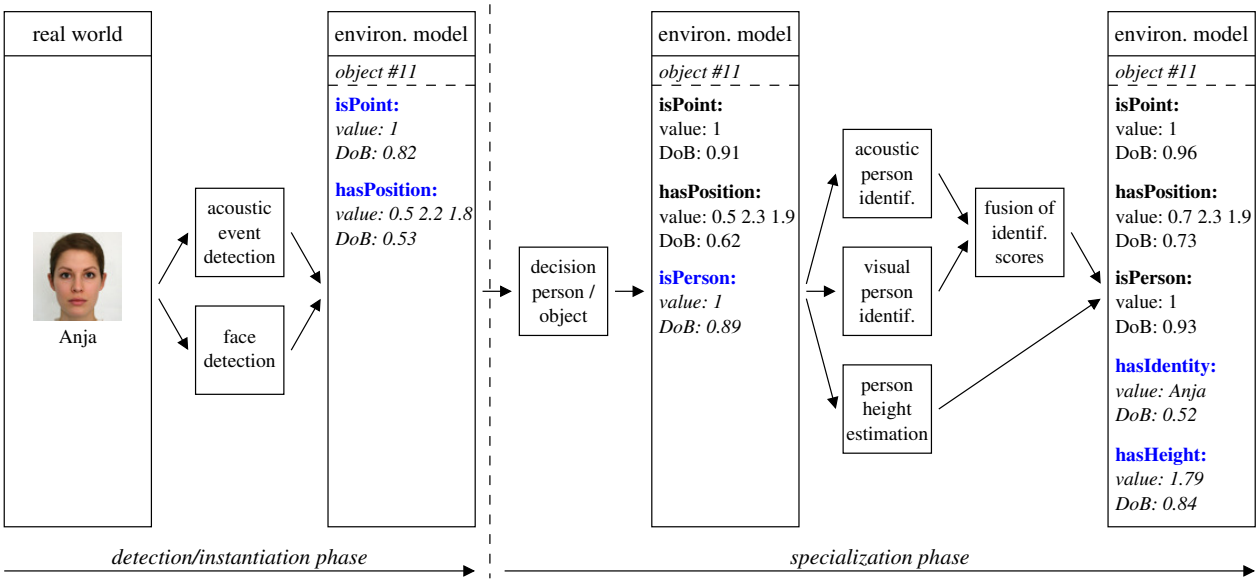


Fig. 3. Example for the typical exploration of an object.

2) Specialization: If the new information is associated to an existing object in the model, it will be fused with the information, which is already available in the model for this object. Thereby, two cases are to be distinguished: the new information can be either used to update an existing attribute, or it results in a new attribute, which is added to the object, changing in this way its abstraction level. In any case, the new information manipulates the DoB for the existence of the object in the model. This phase of the object exploration is continuously repeated as long as new sensor data are available.

3) Deletion: When objects are propagated over the time, the DoB will decrease for the existence of those objects, which were not updated with new information. In the case that the DoB for the existence of an object drops under a predefined threshold, the instance of this object is deleted from the environment model.

An example for the typical exploration of an object is shown in Fig. 3. There, the process is initiated by detections of an acoustic event and/or a face of a person in the real world. Thereby, an instance of the basic class POINT is created in the environment model, with two mandatory knowledge properties *isPoint* and *hasPosition* (*detection/instantiation phase*). In the subsequent *specialization phase*, at first, the decision for a PERSON class is made, which leads to the corresponding specialization and adding the attribute *isPerson*. Based on this knowledge, person-oriented algorithms (acoustic and visual person identification, estimation of the height) are applied in the next step and result in a further specialization (*hasIdentity*, *hasHeight*) of the object instance in the environment model.

III. EXPLORATION OF UNKNOWN OBJECTS

Besides the task of analyzing the scene with the reference to already known objects, there is a need for the robot to be able to explore a (partially) unknown environment. The necessity of such exploration is given every time, when the robot leaves its usual proximity (e.g. kitchen) and enters a new one (e.g. living room). But already the appearance of a new object inside the known environment leads to the need of updating the robot's knowledge with new information. For this purpose, an exploration process is needed, which is particularly focused on the collection of new information about both, known and unknown objects. For example, such kind of information could be color histograms for the characterization of a new object, or the information about the height of an already known person.

A. Multimodal signatures

Like a human being, the humanoid robot can continuously acquire new information about its environment and use it to update the stored knowledge. This allows the robot to gather more and more information about the objects in its proximity in a natural and humanoid way, particularly without supervised learning. It is rather a question of time, how long it will take to explore all unknown objects just carrying out the everyday tasks, for which the perception units of the robot are employed.

Having this in mind, it is necessary to know how the acquired knowledge can be transformed to a representation, which is geared to be stored in the environment model. Such kind of representation is given by unique *signatures* of the object, which are generated from its typical characteristics. Due to the fact that both, many objects and persons can be perceived by more than one modality of the robot, the corresponding signatures are typically multimodal. For example, a signature for a coffee machine consists of corresponding acoustic and visual features. However, the visual features could comprise of color histograms, dimensions, and shape information. Taking all this information together, an unique

multimodal signature can be generated and stored for the object coffee machine. Exactly these signatures are used in combination with *Classifier* modules, described in Section IV-A, to recognize the detected object at a later time.

B. Assisted exploration

The generation of multimodal signatures is not a trivial task. Due to the fact, that an object is not mandatory observable all the time by every modality, several unimodal signatures can be created for one and the same object, which should be fused to a multimodal signature. Besides this, the uncertainty of the *Classifier* modules could lead to the generation of a completely new signature for an already known object. As a consequence, several signatures would characterize one and the same object.

However, for the robot, it is very difficult to resolve this situation on its own. Therefore, the assistance by a human guide can be helpful. This person could advise the robot to fuse redundant signatures or correct the wrong ones, on the one hand, but also tag autonomously generated signatures with labels, which can be easily interpreted by humans (e.g. "Person Anja" instead of "object #11"), on the other hand.

IV. ARCHITECTURE OF THE FRAMEWORK

To realize the hierarchical exploration described above, this section proposes a modular, data flow-oriented concept. In the following, the structure of the framework as well as dependencies between different stages of the process of the object exploration are described.

A. Module categories

Modules within the framework are organized in categories, which represent various types of tasks. Thereby, seven categories can be characterized: *Initiator*, *Classifier*, *Fusion*, *Attribute*, *Tracking*, *View*, and all further modules, summarized in the *General* category.

a) Initiator: Modules of the *Initiator* category provide the ability to create an instance of an object in the environment model. This initial object is always of type POINT and have two mandatory properties *isPoint* and *hasPosition*. In other words, the class POINT represents the existence of an object instance in the environment model. However, alternatively new sensor data can be associated to previously created objects.

b) Classifier: The main task of the *Classifier* modules is to perform the specialization of the basic class POINT. The particular characteristic of these modules is their nesting ability. It is possible to execute one classification process after another one or to activate more than one module in parallel. Thus, the class POINT is specialized in a repetitive process. For example, the class POINT can be specialized as an OBJECT in the first step, and as a CUP in the next step. Usually, classifier modules operate on various statistical models which build the basis of the classification process. For example, Gaussian Mixture Models (GMMs) can be used to distinguish between different kitchen appliances based on their characteristic acoustic signatures [11].

c) Fusion: The *Fusion* category consists of modules, which are specialized on the consolidation of the corresponding classifier results. In this way, the results of two or more classifiers are combined to one decision, e.g. outputs of the modules for acoustic person identification and face identification to the global identity of a person ([10], [6]).

d) Attribute: Modules of the *Attribute* category represent the further specialization of a class and can extend it with various attributes. As mentioned before, the availability of attributes depends on the specialization of the class POINT. While the class POINT specialized as PERSON can be characterized by the attribute *hasHeight*, the attribute *hasFillingLevel* could be a typical property for an object specialized as CUP.

e) Tracking: The *Tracking* category consists of special modules for the attribute *hasPosition*. The peculiarity of this attribute is that it has to be generated and updated by only one of the Tracking modules exclusively. For example, while a cup is tracked by a generic median based tracker, a particle filter based approach can be used for a person.

f) View: Modules of the *View* category are not mandatory for the execution of the system. They rather provide various representation possibilities of the knowledge in the environment model to the outside world. For example, a GUI module can be integrated to give a human user an overview about the acquired and stored knowledge, but also about the available statistical models for the classification task.

g) General: Finally, modules in the *General* category provide global functionalities like data acquisition or self-localization.

Regarding the life cycle of an object in the environment model described in Section II-B, the above module categories can be assigned to the three phases of the object exploration. While modules in the *Initiator* category provide the abilities for the *detection and instantiation phase*, all other modules - excepting the *View* and *General* categories - belong to the *specialization phase*. Especially, the *General* modules cannot be assigned to a specific phase due to their generality.

B. Reflecting the principle of hierarchical object exploration

The idea of the hierarchical object exploration given in Section II ensures that only perception tasks are executed which make sense on the current representation level of the object to be explored. In particular, that means that modules for analyzing the object at the lower abstraction level are always executed subsequent to modules which are operating on the higher abstraction level. For example, a module which is specialized on the face identification task cannot be executed until the current object has been tagged as a PERSON (e.g. based on acoustic sensor data) with a degree-of-belief (DoB) above a certain threshold.

C. Module dependencies

Taking into account the above mentioned considerations, two kinds of dependencies between the modules can be derived for the system architecture – the data dependence and the knowledge dependence.

The *data dependence* reflects the data flow-oriented aspect of the system. Thereby, the acquired sensor data have to be processed over several stages within the system, following a defined processing chain. This chain is implicitly determined by available modules within the framework as well as their perception abilities. In doing so, the system has to ensure that prior to the execution of a module, all required input data for this module were generated as output of one or more previously executed modules. For example, the module for the acoustic person identification requires acoustic features, which are generated by the module for the acoustic event detection. An example for the data dependence is given by the graph in Fig. 4 and represents the data flow between the modules.

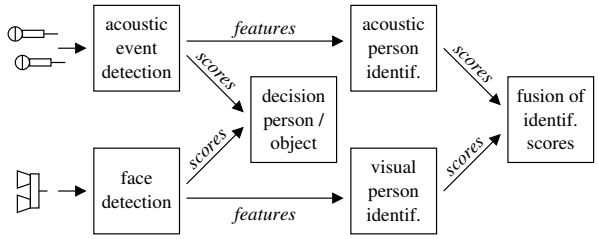


Fig. 4. Example for data dependencies between modules for the exploration of a person.

While the data dependencies are focused on the consistent data processing flow (determination of the order/sequence of the processing chain), from the knowledge-oriented point of view, additional dependencies related to the abstraction level have to be considered. This is what we call *knowledge dependence*. In particular, the knowledge dependence implies, that the "knowledge flow" (that is, the level of abstraction) starts with the object representation on a high abstraction level (POINT) and ends on a lower level of abstraction (e.g. detailed description of an object). Translated to the module processing chain, this means that modules on a lower abstraction level are always executed subsequent to modules which are operating on the higher abstraction level. Furthermore, the processing chain based on knowledge dependencies takes into account, that modules for a more detailed object description (lower level) are only executed, if the reliability (or degree-of-belief) of the higher level results fulfills minimum requirements (higher than a given threshold). For the above mentioned example in Fig. 4, the detected object has to be specialized as a PERSON prior to the execution of the modules for the acoustic and visual person identification. The corresponding example is shown in Fig. 5.

D. Module execution

Considering the above described dependency kinds, each module defines its own data and knowledge dependencies. In order to execute a module, the system has to guarantee the fulfillment of all corresponding dependencies. This results in a non-random executing order of the modules. It is rather

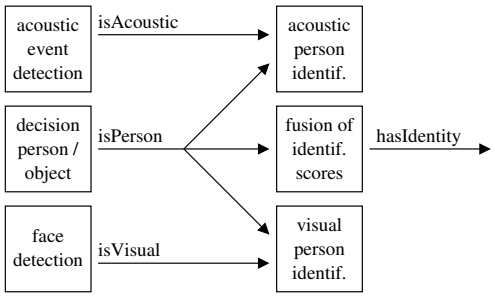


Fig. 5. Example for knowledge dependencies between modules for the exploration of a person.

a question of the appropriate dependencies when a module has to be executed by the system. Based on this constraint, a hierarchical structure of the entire framework can be derived.

Fig. 6 shows the corresponding module executing order, on the basis of the dependency examples given in Fig. 4 (data dependence) and Fig. 5 (knowledge dependence).

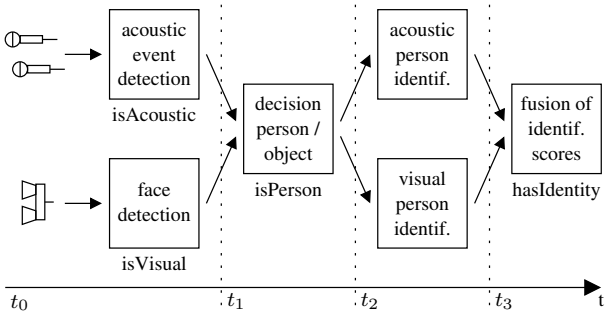


Fig. 6. Example for the module executing order based on data dependencies in Fig. 4 and knowledge dependencies in Fig. 5.

The automatic determination of the module executing order in combination with the modular structure of the framework allows to add and replace modules in a very comfortable way, in particular without changing the system architecture itself.

V. OPASCA SYSTEM

In this section, an exemplary implementation of the proposed concept for the opto-acoustic scene analysis is presented. Taking into account the considerations and requirements for such a framework described in previous sections, we built a real-time system for the opto-acoustic scene analysis in a proof of concept way. The system is called OPASCA (OPto-Acoustic SCene Analysis) [8].

A. Sensor setup

Although the proposed concept is not branded to any special sensor setup, a humanoid sensor arrangement was used in this work (see also Fig. 1). Therefore, a stereo camera with 6 mm focal length as well as a microphone array were mounted on a pan-tilt-unit. Thereby, the microphone array consisted of six condenser Lavaliere microphones. Two of them are placed on the positions of the human ears with

a distance of 19 cm. Two further microphone pairs are placed on the median planes of the head at the vertical level of the nose as well as at the level of the forehead, both with a distance of 23 cm. The distance between both front microphones is 6 cm. Two microphones on the back of the head are mounted with a distance of 4.5 cm.

B. Module overview

As mentioned before, the scene analysis consists of a variety of perception tasks, e.g. detection, tracking, and identification of persons and different types of objects. The framework modules and corresponding tasks, which can be currently addressed by the OPASCA system, are summarized in Table I. By analogy with Section IV-A, the modules in the OPASCA system can be sorted into seven categories. However, this compilation shows only a selection of some exemplary implementations trying to tackle the primary perception tasks of a humanoid robot. When required, further modules can be integrated into the system at any time.

C. Selected evaluations

In the following, some selected abilities of the OPASCA system are evaluated in real scenarios.

a) Life cycle of an object: Addressing the hierarchical object exploration, described in Section II-B, an exemplary real life cycle of an object is shown in Fig. 7. Thereby, the scenario consists of a person, who appears in the field of view of the robot, speaks to the robot, and finally leaves the field of view. The figure shows the DoB for the attributes of the explored object at different exploration stages.

At first, a new object is instantiated at time t_0. The instantiation is caused by an acoustic event, which has been detected by the *Initiator* module iAcoustics. Simultaneously, the attributes *isPoint*, *hasPosition*, and *isAcoustic* are added by the *Fusion* module fInitiator to the created instance of the object in the environment model.

Few moments later, the object is specialized as a person (by the *Fusion* module fPoint) and the attribute *isPerson* is added at t_1.

As soon as the DoB for *isPerson* reaches a certain threshold, various person-specialized modules are executed. This includes the *Classification* module cPersonAcousticID with the corresponding *Fusion* module fPerson, but also the *Attribute* module aPersonHeight for the estimation of the height. Consequently, two further attributes *hasIdentity* and *hasHeight* are created at t_2.

Few seconds later, the detected person enters the field of view of the robot. This event is noticed by the *Initiator* module iVisualPerson and results in the creation of the corresponding attribute *isVisual* at t_3. Due to the availability of visual data and the classification as person, the *Classification* module cPersonVisualID can be executed. From now on, the classification results of cPersonAcousticID and cPersonVisualID are fused by the *Fusion* module fPerson.

At time t_4, the person stops talking and leaves the field of view of the robot. As a result, there are no new detections, neither acoustic nor visual. Thus, the DoB for the attributes *isVisual*, *isAcoustic*, and *hasPosition* decreases continuously. This entails the decrease of the DoB for *isPoint* at the same time.

Finally, at time t_5, the DoB for *isPoint* reaches the predefined threshold for the deletion. As a result, the instance of the object is deleted from the environment model and the life cycle of this object is over.

b) Multimodal fusion: As an example for the fusion of different information sources, the multimodal identification of a person is shown in Fig. 8 (a). On the ordinate, the mean probability for the correct recognition is given. Additionally, all results are presented for different signal acquisition dura-

TABLE I

OVERVIEW OF INTEGRATED MODULES IN THE OPASCA SYSTEM WITH A SHORT DESCRIPTION OF THEIR SPECIALIZED TASKS

category	module name	description of task
Initiator	iAcoustics	acoustic event detection and localization [2]
	iVisualPerson	face detection and localization [13]
	iSurfaceObject	segmentation of objects on a-priori known surfaces
Classifier	cAcoustics	GMM based acoustic event classification [11]
	cVisualPerson	detection based classification for the class POINT
	cSurfaceObject	detection based classification for the class POINT
	cObjectColor	color histogram based classification of objects
	cObjectShape	shape based classification of objects
	cObjectState	HMM based estimation of operating states of an acoustic observable object (e.g. coffee machine)
	cObjectVolume	volume based classification of objects
	cPersonAcousticID	UBM based acoustic identification of persons [9]
	cPersonVisualID	local DCT features based face identification [3]
Fusion	fInitiator	calculation of the DoB for the class POINT (existence)
	fObject	fusion of classification results for the class OBJECT
	fPerson	fusion of classification results for the class PERSON
	fPoint	fusion of classification results for the class POINT
Attribute	aObjectPose	stereo information based estimation of dimension and pose (length, width, height, rotation angles) of an object
	aPersonGesture	marker based estimation of pointing gestures of a person
	aPersonHeight	3-d position based estimation of the height of a person
Tracking	tDefault	default tracker for classes POINT and OBJECT
	tPerson	particle filter based tracking for the class PERSON
View	vGui	representation of the knowledge in the environment model
General	gDataAcquisition	interface to physical sensors of the robot
	gSelfLocalization	marker based self-localization and pose estimation of the sensor head

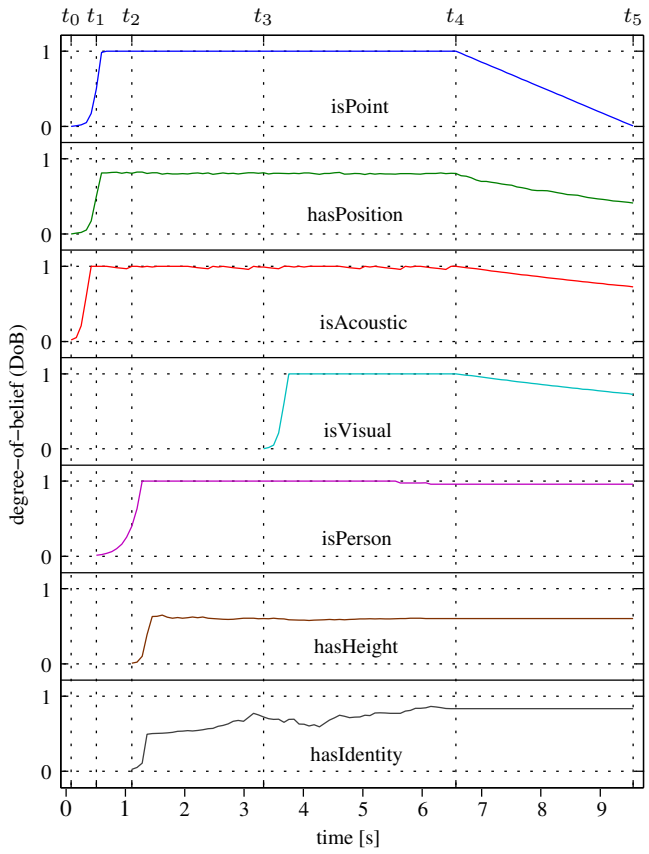


Fig. 7. Example for the typical life cycle of an object, showing the DoBs of attributes in the environment model, which are created during the exploration process by the real-time system.

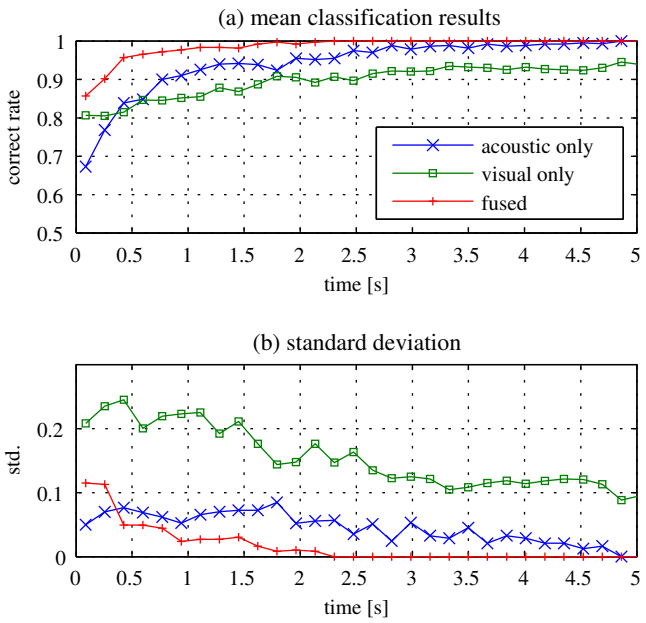


Fig. 8. Example for the fusion of unimodal classification results (acoustic person identification and face identification) for the multimodal person identification.

tions. That means that the classification result is available as soon as a sensor data block of a specific length is acquired. The x-axis shows the corresponding acquisition time. For this setup, a database with six multimodally recorded persons was used.

It can be seen that the unimodal classification results, acoustic only and visual only, are partially significantly worse than the fused multimodal classification result. Additionally, the standard deviation between different persons is given in Fig. 8 (b). The fusion process results in a significant lower standard deviation, which indicates the improved robustness of the classification.

c) Multimodal signatures: As pointed out in Section III-A, the new sensor information acquired by the robot during the exploration process can be used to create and update the unique signatures of objects in the environment model. Fig. 9 (a) shows the mean quality of generated signatures for six persons after the acquisition time, given on the x-axis. The quality of a signature, which is used for the classification in combination with *Classifier* modules, is determined by the acquisition time, which has to be passed, before the corresponding signature is generated.

As it can be seen, the mean correct classification rate increases slightly when more sensor data are used for the generation of a signature, both, acoustic and visual. However, in this case, the standard deviation between different persons, given in Fig. 9 (b), decreases significantly. This can be interpreted as an indication for a more reliable classification.

Nevertheless, there is no need to collect a lot of sensor data at once to get a high quality signature. The ability to update an already existing signature (of a low quality) in the environment model is implicitly provided by the system and the *Classifier* modules. It is rather a question of new matching sensor data: as soon as new data are available, an already existing signature of the corresponding object can be refined.

D. Real-time capability of the system

The OPASCA system operates on a single computer (Intel Core 2 Quad with 2.67 GHz and 4 GB RAM). While a maximum processing rate of 11.7 frames per second can be achieved, the typical frame rate amounts about 9 frames per second. This cycle is sufficient to perform the scene analysis task and to keep the environment model in a state, which continuously reflects the current proximity of the robot.

VI. CONCLUSION AND FUTURE WORK

A. Conclusion

This paper presents the framework for the opto-acoustic scene analysis. The hierarchical exploration approach for the environment of a humanoid robot is described for both, known and unknown objects. Thereby, three different exploration phases are characterized: *detection/instantiation*, *specialization*, and *deletion*. The entire exploration process is based on the *object-oriented environment model*, which is constantly used for acquiring, saving, and recalling information about the environment of the robot. In particular, the environment model provides the ability to store and exchange

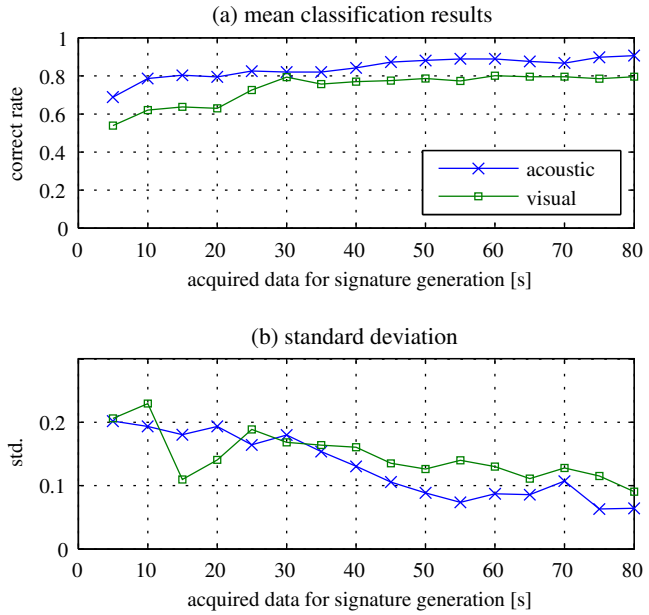


Fig. 9. Example for the unimodal person identification using unique signatures (both, acoustic and visual) of different quality. The quality of a signature is determined by the acquisition time, which has to be waited for, before the corresponding signature is generated.

knowledge generated and required by specialized perception modules.

Additionally, a further type of knowledge can be stored in the environment model. This information is given by unique *multimodal signatures* of explored objects and can be used to recognize these objects at a later time. The exploration process can be supported by a human guide within the *assisted exploration*.

For the realization of the described exploration process, a hierarchical, knowledge-oriented, and modular concept of a framework is proposed. Thereby, the focus is put on formal conditions in form of dependencies between various perception modules. In doing so, two kinds of dependencies are distinguished. The *data dependence* reflects the data flow of the acquired sensor data, which is processed over several stages, following a certain processing chain determined by the available modules within the framework and their specific perception abilities. By the *knowledge dependence*, the knowledge-oriented point of view of the object exploration is considered. Related to the abstraction level of the exploration process, the knowledge flow starts with the object representation on a high abstraction level and ends on a lower one. For both dependence types, each module within the framework defines its own data and knowledge dependencies, which implicitly restrict the execution order of the modules during the exploration process.

Based on the formal proposed framework, the developed highly integrated, real-time system for the opto-acoustic scene analysis (OPASCA system) was presented. An evaluation scenario was exemplary shown in form of a typical life cycle of an object during the exploration process. Additionally, the benefit of the multimodal fusion was demonstrated. A further example related the quality of multimodal signatures depending on the amount of available sensor data.

B. Future Work

Future work will have to address the question of an intelligent knowledge- and task-driven symbolic planning of the entire process of exploration. In contrast to the current design of the framework, which aims to collect the maximum of knowledge at any time, a smart planning of the exploration would only collect the information which is needed to fulfill the current task of the robot. For example, the task to identify the person near the robot does not inevitably require to estimate the height of the person.

Therefore, different knowledge sources (e.g. a-priori knowledge, current situation, input from interaction with an user) as well as general restrictions and limitations (for example, possible skills of the robot, competitive tasks) have to be taken into account.

VII. ACKNOWLEDGMENT

This work has been supported by the German Science Foundation (DFG) within the Sonderforschungsbereich (SFB) 588 "Humanoid Robots - Learning and Cooperating Multimodal Robots".

REFERENCES

[1] T. Asfour, K. Welke, P. Azad, A. Ude, and R. Dillmann. The Karlsruhe Humanoid Head. In *Proceedings of the 8th IEEE-RAS International Conference on Humanoid Robots*, December 2008.

[2] J. H. DiBiase, H. F. Silverman, and M. S. Brandstein. *"Robust localization in reverberant rooms"*, chapter 8, pages 157–180. Springer, Berlin, 2001.

[3] H. K. Ekenel and R. Stiefelhagen. Local appearance-based face recognition using discrete cosine transform. In *FRGC 2.0 Database, Face Recognition Grand Challenge Workshop (FRGC)*, 2006.

[4] I. Gheta, M. Heizmann, and J. Beyerer. Object oriented environment model for autonomous systems. In H. Bostrm, R. Johansson, and J. van Laere, editors, *Proceedings of the second Skövde Workshop on Information Fusion Topics*, pages 9–12. Skövde Studies in Informatics, November 2008.

[5] F. Kanehiro, H. Hirukawa, and S. Kajita. Openhrp: Open architecture humanoid robotics platform. *I. J. Robotic Res.*, 23(2):155–165, 2004.

[6] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas. On combining classifiers. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 20(3):226–239, 1998.

[7] K. Ng-Thow-Hing, R. K. Thrisson, Sarvadevabhatla, and T. L. J. Wormer, V. The cognitive map architecture for facilitating human-robot interaction in humanoid robots. *IEEE Trans. Robotics and Automation*, 2009.

[8] OPASCA research group. OPASCA (OPto-Acoustic SCene Analysis). `http://www.opasca.org`.

[9] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn. Speaker verification using adapted Gaussian mixture models. In *Digital Signal Processing*, page 2000, 2000.

[10] J. Sander and J. Beyerer. Fusion agents - realizing Bayesian fusion via a local approach. In *Proceedings of the 2006 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI06)*. 249-254, September 2006.

[11] A. Swerdlow, T. Machmer, B. Kühn, and K. Kroschel. Sound source classification under mismatched conditions for a humanoid robot. In *Proceedings of the IEEE Region 8 Eurocon 2009 Conference*, May 2009.

[12] The Collaborative Research Center 588. Humanoid Robots - Learning and Cooperating Multimodal Robots. `http://www.sfb588.uni-karlsruhe.de/`.

[13] P. Viola and M. Jones. Robust real-time object detection. In *International Journal of Computer Vision*, 2001.