

THE ARTIFICIAL RECOGNITION SYSTEM (ARS): NEW CONCEPTS FOR BUILDING AUTOMATION

Gerhard Pratl, Brigitte Lorenz, Dietmar Dietrich

*Institute of Computer Technology
Vienna University of Technology
{prat1, lorenz, dietrich}@ict.tuwien.ac.at*

Abstract: Building automation faces a development that leads to more and more sensory information available for processing. Existing approaches are challenged by this abundant amount of data, therefore the authors see a need to introduce new concepts for handling the challenges of the upcoming future. Looking at bionic approaches taken from the field of neurobiology, but also psychoanalysis, a system is created that applies neurological principles to get human-like perception, and psychoanalytical principles to evaluate the perceived scenarios. *Copyright © 2005 IFAC*

Keywords: Fieldbus, Sensors, Autonomous Control, Intelligent Control, Cognitive Systems, Bionics, Knowledge Representation, Image Recognition, Evaluation, Behaviour.

1. INTRODUCTION

By reading journals about Building Automation, like *LonMark* (April 2005: Building a Brighter Future), the problems in the future of fieldbus systems become visible. State of the art is to integrate many thousands of networked sensors, actuators and controllers in one system to control various processes in a building – temperature, light, motion, etc. – based on classic control loops with threshold values or similar principles. Higher functionalities, like energy optimizations or security aspects, are solved by similar control systems. It is hardly imaginable that it will be possible to handle – in particular to integrate and to maintain – a drastically increasing number of integrated, multi-functional components in an efficient and low-cost way. But we will be confronted with such high amounts of diverse sensor data in the not so distant future, especially if we consider sensor networks of thousands of intelligent nodes in products like carpets, walls or similar smart parts. We have to think about new possibilities. Of course the performance of embedded systems is increasing as well, but the question remains: Are the classic methods of automatically processing sensory input data enough? The authors are convinced that in order to answer this question we should analyze nature and try to follow its way by copying its solutions.

This paper reports about the bionic approach and the new results of the *Artificial Recognition System (ARS)* project, which was presented in Brainin, *et al.* (2004), or, more recently, in Pratl, *et al.* (2005). There, the authors argued that within the last decade the psychologists and psychoanalysts could verify that our brain does not compute current situations (images) and scenarios by analyzing *all* the incoming data from the various sensors of our body each time again and again. Instead, the human brain memorizes images and scenarios over a whole lifespan. However, this is not the only aspect we pointed out as decisive difference to the traditional thinking of engineers about human cognitive abilities and how to model them technically. The human brain is not able to fully

process all incoming data in real time. The brain is only able to perceive and recognize characteristic forms of the outside world. The rest is associated with the aid of the already memorized images (Dietrich, *et al.*, 2004). The association (comparison) of the new images – which creates a mix of computed data based on inputs from the environment and internal, previously memorized data – releases emotions, which, in turn, trigger and guide the actions and reactions of human beings.

The difficult question is how we can technically implement the observed principles. Classic methods, like neural networks, fuzzy logic, knowledge-based systems, etc. are placed on the market. However, although some solutions are quite brilliant, there is no striking application for the area of fieldbus systems. Therefore the authors follow the ideas and principles of the neurologists and psychoanalysts. Of course, it is impossible to artificially construct a system, which implements every functional aspect of the architecture of the human brain, especially if we want to take over principles of the theories of Sigmund Freud. We have to take the "usual" method of engineers and start with very simple parts. A very good example for this approach is presented in a dissertation dealing with an automatic component part sorter for an assembly line (Brenner, 2001), whose realization has been a great success on the market. The first step was to differentiate between only three different components; today the system is able to differentiate between more than 100 learned parts.

This recalls the story of Kaspar Hauser (Hauser, 1995) who was shielded from the outside world throughout his entire childhood and early youth and was kept alone in a dungeon. There, he could only see and, thus, get to know, a very limited set of objects. When he finally saw the outside world with all its complexity, he had the greatest difficulties to accept this "real" world because he did not know it – in our language: He did not have a memorized image of it, and it took him a lot of time and effort to learn how to interpret all the new impressions.

If we define constraints such that our system only needs to "understand" images similar to already memorized ones, and if we only accept a limited set of possible scenarios (applications), then it will be possible to come to a "simple" solution.

As already explained in Tamarit, *et al.* (2001), the base is a hierarchical system where the inputs are symbolized at different levels. The higher symbols are the input for another processing functional unit, where complex functions are defined and actions/reactions initiated. The information flow is evaluated by emotions, generated by comparison between inputs and memorized images. So, we have to separate the ARS project in two different parts: the hierarchical lower level functions, mainly from the point of view of neurologists, and the higher complex functions, where Freud's model is taken over (Brainin, *et al.*, 2004).

2. APPLICATIONS

We focus on the development of a system, termed the Artificial Recognition System (ARS), which makes use of diverse, redundant sensory inputs in a bionic way to create a perception and consistent representation of the surrounding world. This enables the system to perform the desired functions and to attend to the required applications. In order to demonstrate the approach of our system we have chosen four applications from different domains.

Application 1: Human Surveillance System

By making use of light barriers and detectors, tactile sensors in the floor, door contacts, and stereo cameras the system is enabled to know the position of persons in a building. People are considered to be anonymous, which means that the system has no additional knowledge about their identity. That is to say, unless a person is furthermore provided with an identification mechanism (e.g. due to authentication at a security door). The system is able to provide information about a person's current and past location, so that the path of a person through a building can be tracked and monitored.

Application 2: Child Safety System

The second application we consider is a *child safety* system. The system recognizes whether a particular person is actually a child, and can monitor and guard the actions of the child. When it appears that the safety of the child may be compromised due to a hazardous situation, the system alerts a (human) supervisor.

The decision that a particular person is actually a child is based on diverse sensor information, similar to other mechanisms in the system. This includes the use of camera images to derive height and shape of a person, as well as information from light barriers that are mounted at different heights, and weight information obtained from pressure sensors to support the decision. Example situations that are classified as hazardous are: an open fire, a hot stove plate, a cupboard with open doors, or a child climbing on

table. There are additional conditions and criteria that have to be taken into consideration. For example, a situation is only classified as hazardous if a child is alone and unattended (meaning that no adult is in the same vicinity). Other conditions include the facts that the fire has to be burning, or that the stove is indeed hot.

Application 3: Geriatric Care System

The third application is concerned with a *geriatric system* to care for elderly people. In this case the system is able to recognize when an elderly person collapses or faints. Furthermore, the system is able to identify and track the location of predefined objects, such as keys, glasses, and books.

Application 4: Theft Protection

The fourth application shall supervise the whereabouts of classified things. In a given room the system monitors a set of objects (e.g. books in a library) that are not allowed to be removed. In case a person takes away such an object, the system shall inform a (human) supervisor.

These applications share a common layout, in the sense that we define a number of rooms on a floor that has a layout, which is identical for all four applications. The sensors that are used are also identical, and mounted in the same position. In this way the symbolization mechanism shares a common set of symbols (although not all symbols have to be present in all applications).

Obviously the system has only a limited understanding of the world it perceives. The fact that cameras are installed does not automatically imply that the system is able to process all the information in a way that is similar to a human operator observing and evaluating a camera image. For example, suppose that a dog enters the room. This could possibly be perceived as a "person" (or, at best, as "child"), since the system has no initial concept of a dog. Hence, the system is bound to make incorrect decisions if it is confronted with facts or images that are outside the scope of its capabilities. This system attribute is intentional, since it does not form part of the task that needs to be fulfilled. If we introduce a new application, which makes it necessary to distinguish animals from persons, the knowledge of the system will have to be extended.

3. SYMBOLIZATION

We have to find a way to cope with the vast amount of diverse sensory input that the system has at its disposal. We want to extract relevant information, ignore everything that is not important and use the existing redundancy in sensory information to get a stable and robust perception of the environment. Therefore we break the available information into small chunks and label them *symbols*. A symbol in this context is a concept that the system has means to operate on. It contains information that can originate either from sensory input or from knowledge of different kinds. If a symbol is created only by sensory

input, we refer to it as a *microsymbol* (μ Symbol). μ Symbols represent small pieces of information that are the base for higher level symbols. Different symbols have different “weight” in the way they represent information. We assume a hierarchy in the symbols, meaning that there are μ symbols with only little weight (i.e. they represent a piece of perception, e.g. a footprint) and there are symbols of higher weight (e.g. there is a symbol that represents a chair leg and there is a symbol that represents a chair – which requires four chair legs and some other symbols to be present in a certain relation to each other). The first step in recognition is to create μ Symbols out of sensor input; from then on the system only operates on symbols and not on actual sensor data.

3.1 Microsymbols

For the scope of this paper we limit the variety of different μ Symbols and symbols and focus on a set of only three symbols: table, chair and person. The system has the goal to determine the position of persons, and to detect tables and chairs. The sensors we look at are shown in Figure 1.

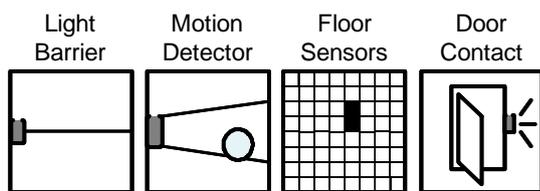


Figure 1. Sensors that provide data on which the perception of the system is based. Cameras are used for multiple purposes.

Cameras play a special role, since they are able to provide lots of information within one sensor (Figure 2). We do not aim at implementing a new set of image processing algorithms, since this is already state of the art. Instead we want to employ these techniques to process camera images and extract different features out of it. This processing happens on different levels of complexity: for once, we want to detect movements in the visible area of the camera: this is done by background subtraction (the cameras are mounted statically).

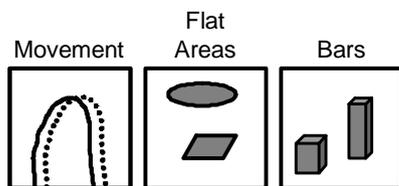


Figure 2. μ Symbols that are created by cameras.

Then we want to extract basic features like lines, edges, corners and areas. Using stereo cameras and feature matching we gain information about the spatial position of these features and can create flat surface areas. In a next step we will use sophisticated camera systems to extract persons that are divided into their body parts, namely head body, legs and

arms; this is however not covered in this paper. To match the requirement we introduced before (recognize persons, tables and chairs) we use cameras to detect motion, flat areas and bars (i.e. objects that are long and narrow).

The μ Symbols and the respective sensors that we need are listed below. The name of the μ Symbol is chosen to provide a human understandable meaning of the quality of this information (of course this principle is not extensible when dealing with larger amounts of diverse sensors; we will have to automate this process when it gets too time consuming). The names in brackets are properties of this μ Symbol; μ Symbol and properties are an entity, properties cannot exist without a μ Symbol.

Table 1 Microsymbols and their associated Sensors

Sensor	Microsymbol
light barrier	personPassesBy(sensorPos, direction)
motion detector	personMoves(sensorPos)
floor sensors	footprint(sensorPos)
door contact	doorOpens(sensorPos)
camera	camMovingPerson(sensorPos, size, direction); camFlatArea(sensorPos, size, position, orientation); camBar(sensorPos, size, position, orientation)

All μ Symbols contain information about the location of the sensor and its sensitive area, this is indicated by `sensorPos`. Like in embodied intelligence we need to have metainformation about the sensor to determine how to interpret the actual sensor value. This information needs to be configured at setup time and is then known to the system. Without this metainformation the actual sensory input is useless, therefore we indicate the knowledge of the whereabouts of a sensor by `sensorPos`. What is not shown here is for once a timestamp, indicating when the μ Symbol was created. μ Symbols have event character, meaning that they exist in only one point in time. Also left out is the sensor type. It is intrinsically contained in the name of the μ Symbol, but it is of course possible that more than one sensor can create the same type of μ Symbol.

`personPassesBy` indicates that a set of two light barriers has been triggered within a certain time period. `direction` is derived by looking at which light barrier was triggered first. `personMoves` is the information that a motion detector can contribute to the position of a person. The important metainformation is the trigger area of the sensor, i.e. which area the sensor "sees". `footprint` is created by floor sensor, which detects changes in their state. Depending on the size of the floor sensors a footprint μ Symbol can depend on only one sensor or on a couple of sensors. `doorOpens` requires a sensor to be installed in a door that is triggered when the door is opened.

camMovingPerson is the differential image mentioned earlier (using background subtraction). It indicates that a moving shape, which matches the size of a person, has been detected. It has two properties that describe the shape and the direction in which the shape has moved. camFlatArea identifies any kind of flat area; it requires the use of stereo cameras, because the real size (not the size on the camera image), position and orientation are required. camBar identifies any objects that have stick-shape, i.e. long and narrow shapes.

3.2 Symbols

The next step is now to take the preprocessed perception information (available as μ Symbols) and create symbols. Note that starting from this level there is no more sensor information to be processed, everything that the system knows about the outside world is merely available as μ Symbols. We have defined a fixed set of symbol templates, the system only knows symbols that are created from these templates and cannot learn new symbols. Just like with μ Symbols this approach will have to be extended in future, when the number of different symbols is vastly extended.

The scheme for symbols is similar to μ Symbols. We have a name for each symbol and we have properties. Additionally we define *associations* (see next section); these indicate which symbols can be associated with each other (referring to the type of symbol and μ Symbol, not the actual symbol). Symbols can be associated with an arbitrary number of μ Symbols, symbols or a mixture of both. And we use a hierarchy of symbols, meaning that symbols can inherit properties and associations of other symbols. The list of symbols we define is:

```
Symbol object
```

```
Symbol thing inherits from object
Properties: size, position, orientation,
direction
```

```
Symbol person inherits from object
```

The following symbols inherit from symbol thing and do not have any additional properties: table, tableLeg, tabletop, chair, chairLeg, chairBack, chairSeat.

Object is the root symbol in this hierarchy; it has no properties, but is the base for all other symbols, including thing and person (a person is an object in this context). This eliminates redundancies, because things in general have some common properties and allows to introduce more abstract information processing (knowing that a chair is a thing can be useful knowledge). Note that a lot of symbols do not have additional properties (except for the ones they inherit), still the definition per se is important, because the system has means to process symbols depending on what they represent (indicated by the symbol name). This approach raises a lot of questions, for example how to fill a formal symbol with semantic interpretation that is intrinsic to the

system. This *symbol grounding problem* is discussed in (Harnad, 1990).

If we take a look back to the system perception and look only at μ Symbols that are created by cameras, we see that there is still a semantic gap that needs to be filled: a camera can find flat areas and bars in the image. These are intended to be used as the building blocks for chairs and tables. However, not every bar makes a chair leg and not every flat area is a tabletop, chair back or chair seat. This gap is filled in the next section.

3.3 Associations

An association defines means to create symbols by combining other symbols or μ Symbols. For the goals defined in this paper we need the following associations:

```
person associates with: footprint, doorOpens
personMoves, personPassesBy, camMovingPerson,
```

```
table associates with: tableLeg, tabletop
```

```
chair associates with: chairleg, chairback,
chairseat
```

The system has to use the μ Symbols that have been perceived and combine them in a way that the symbols person, table or chair is recognized. The associations listed here contribute to this process by providing knowledge that is needed. In the case of tables and chairs this knowledge is of structural and spatial kind: a chair has certain geometry, i.e. geometric relations between its parts. All legs shall be parallel to each other and of the same size and the size needs to be in certain boundaries. The seat has to be parallel to the floor and on top of the legs, while the back has to connect orthogonally to the seat. Therefore the perception information that entered the system by means of μ Symbols is now analyzed towards matching pieces of information: if there is a flat area of adequate size and orientation, if there are bars that are as well of adequate size and orientation and additionally are connected to the flat area and if there is another flat area that suffices the requirements for a chair back, then the system creates a new symbol chair. The μ Symbols that are used for this chair (which are of type camBar or camFlatArea) are converted into the according parts of a chair, chairLeg, chairBack and chairSeat. Tables have similar associations, only that the dimensions are different. Symbols of type person are created whenever the associated μ Symbols are created. Note that all these μ Symbols contribute merely to the position of a person and do not allow any deduction of activities as needed for the applications described in section 2. These are left out in this paper. But they still provide a valuable contribution that is a key design issue of the system: the μ Symbols represent redundant information about the position and movement of a person: a footprint and a personPassesBy μ Symbol both contain this information. The system has to be able to use all the available information; therefore we need to classify

the contribution of each μ Symbol. This requires having a look at the origin where a μ Symbol came from. Light barriers (we assume the combination of two light barriers to gain the direction of a movement) that are installed horizontally in a hall have rather precise knowledge of one axis, but rather poor knowledge of the other axis: the position of the light barrier itself gives the first axis, but where on the hall it was interrupted is unknown. A tactile floor sensor exactly knows its position; however, it requires some computational effort to find footsteps that belong to the same person. The direction information of light barriers is useful for this matter.

Each contribution is therefore processed and contributes to the goals of the system. Embodied intelligence teaches us that a lot of "intelligence" is grounded in the way sensors are installed and in the capabilities of the sensors. Therefore the concept and layout of the system and the knowledge of capabilities of each sensor are important when classifying different incoming information.

4. WORLD REPRESENTATION

In the previous sections the flow of information, which originated in sensory input had been processed in a way that allowed the system to catch a "snapshot" of the environment it supervises (we will call this environment the *world*, meaning merely the environment that the system can perceive with its sensors; this "world" will usually be limited to a building). These snapshots exist in one point in time; as soon as one snapshot has been created, the system starts to look at new incoming sensory input and creates the next snapshot.

The sequence of snapshots, however, does not make a continuous representation of the world. If we look at the world representation of a human being, we see that there is a lot of knowledge about the surrounding world, which is not strictly bound to the current sensory input. The world representation consists mainly of memories of older perceptions (which are, of course, regularly updated by more current perceptions). And even when perception does not provide us with additional input, we do not immediately forget what we knew: assume that Alice and Bob are in the same room and Bob leaves the room by walking through a door. Although Alice does not know the exact position of Bob any more (maybe in the next room, maybe somewhere else in the building), she still knows the door he has left through. The same applies, if Alice sees Bob walking behind an obstacle: the gaps in perception while Bob is not visible are filled by knowledge about the continuity of events.

Therefore we introduce the *world representation*. It is based on the "Inner Representation of the Outer World" as presented in (Solms, 2002). It is fed by the information that enters through perception, but applies additional rules to get a continuous representation of the world as the system sees it. Looking at the person and the determination of a

persons position we have so far seen a set of snapshots in each of which there might have been a person with a slightly different position. The task of the world representation is now to concatenate all these "person-perceptions" and make a single person out of it – a person that moves. This implies a change in the timing information we have to keep now. In perception everything was bound to a single point in time – reflected by the timestamp. Now we have to extend time information to a time period, because symbols are created and updated during their lifetime.

To ensure the continuity of the world representation, the system has to use the context of the previous world representations. This context helps to create predictions about the future state, which can be verified (or falsified, respectively) later. In the case of persons that are hidden by obstacles, the system knows both position and direction of the person before it loses track. So it can assume that the person is still somewhere around the last position. If briefly afterwards a person is perceived in this vicinity and the assumed direction matches, the gap in the flow of perception can be filled by sheer knowledge (namely that persons can only move continuously) and the path of a person can be reconstructed.

This process is, of course, improvable by increasing the amount of available information. The cameras can be used to not only detect movement, but to also create a color histogram of a person (this histogram would be another μ Symbol). When processing disappearing and reappearing person, the histogram is then used to ensure that the system did not lock onto a different person who happened to pass by as well. Again this is based on knowledge, not perception, namely the knowledge that persons cannot change their color distribution (at least not in a short amount of time).

At the end of this process we get a current representation of the world as the system sees it together with a history of this representation. Therefore we now are able to take *time* into consideration – not just the time between adjacent perceptions, but time in the order of several minutes or hours. The system can now look for *scenarios*, that is, sequences of events that are distributed over a certain time period. If it detects a known scenario it can either trigger the creation of a new symbol or indicate this to a human supervisor. Looking at the applications introduced in section 2 we can identify scenarios in geriatric care defining that the system shall detect, if a person lies on the floor and has not moved for some time. Or, in case of the theft protection application, we can trace objects of interest and identify the person that poses such an object and attempts to leave a predefined area (e.g. a library).

5. EVALUATION OF SCENARIOS

The system has to evaluate the perceived situations and scenarios. It has to "judge" whether there is an out-of-normal situation which requires action or not. Examples for actions the system could have to

perform are: to save a person from harm in case of a dangerous situation, to assist a person when he or she needs help, to protect objects from being stolen or to keep the room temperature adjusted within a comfortable range. Some actions are very time critical – meaning that they need to be performed immediately – others not. This is the first thing the evaluation mechanism has to detect: whether there is an emergency situation or not.

To identify hazardous and urgent situations a kind of alarm mechanism will be implemented. An alarm is a rather “naked” signal with no or only very little internal structure (Johnson-Laird, 1988). It triggers an immediate reaction in a mostly hard-wired way – a reflex circuit. This purely reactive control loop is the simplest one in a hierarchy of control loops. To use such a hierarchy is biologically motivated.

5.1 Bionic Evaluation Principles

In nature, living beings have evolved a cascade of more and more complex control levels for behavior selection, all of them aimed at ensuring survival in highly demanding and constantly changing environments. Each newly emerged control level is built up of parts of the lower levels (Damasio, 2003). However, the different levels are not clearly separated, but rather very much intertwined, with a lot of back-and-forth communication going on over the linkages connecting the various modules. This makes it so hard to analyze their functionality. Moreover, as each level is crucially based on a *feedback loop*, it cannot be decomposed in its parts without losing some of its emergent features. This is because feedback leads to non-linearity and thus the whole process cannot be described as a simple linear superposition of its parts.

Feedback is essential for evaluation. In biology, it can always be described as a loop between experience and expectation whereby both terms are used in a very general way. Experience refers to the input the system receives from the outside world, its environment, and expectation refers to its inner representations and constraints. Outer world and inner representation have to be related to each other. The better this is achieved, the more intelligent the autonomous system will be able to behave. Ideally, the process of relating outer and inner world will be done in a dynamic, adaptive way. This is especially important for constantly changing environments. However, the technical design and implementation of such a flexible association mechanism is a quite difficult task.

Nature starts with simple mechanisms like bodily reflexes, unconditional reactions to appetizing stimuli, initiatives for active behavior like drives and motivations, mechanisms for the intensification or inhibition of specific behaviors mediated via lust or pain, etc. Conditional associations represent an important step forward towards a more adaptive and context-sensitive control of behavior. Conscious reasoning and decision making, guided by ideas, theories, and even a general world view, stand on top

of the behavioral control pyramid (Riedl, 1984). This bottom-up, evolutionary derived view of our cognitive abilities is compatible with the picture Sigmund Freud has worked out about the functioning of our brain.

5.2 Technical analogy

We focus at investigated in which way the above mentioned bionic principles can inspire the technical design of the evaluation and behavior selection unit of intelligent autonomous systems. The field of building automation is insofar special, as in this field the autonomous system cannot “move around” and actively explore its environment like for instance a mobile robot or a software agent can do. A building automation system is comprised of sensors with a fixed position, together with a communication structure and one or more reasoning units. However, there are quite a lot of sensors and they are very much distributed. Thus, the system will not only have access to central knowledge. Instead, it will rather rely on decentralized sources of information. Moreover, the fact that the spread out sensors are getting more and more intelligent raises the amount of locally available knowledge. Another particular aspect of autonomous systems in the field of building automation is that they should be “invisible” most of the time. This means, that, ideally, they will be exerting a more passive than active behavior. Actions will only be taken in case something goes wrong.

What we will take up from biology is a hierarchical, modular structure of control loops. Although, at the beginning, we will try to keep things as simple as possible. Only some parts of the whole complex biological control architecture will be worked out in more detail, and for this, only a selected set of design principles inspired by nature will be picked up.

Ortony, *et al.*, (2002) distinguish three levels of information processing for behavior selection for cognitive architectures:

- *Reactive processing*: This is the simplest form of producing behavior. Especially in emergency situations, hard-wired action patterns shall be released with as little delay as possible. Those actions can be for example alarm signals.
- *Routine processing*: On this level, primitive, automated behaviors are executed. They are initialized by stimuli which trigger them. In the simplest form, this happens in an unconditional way.
- *Reflective processing*: Here, explicit hypothetical representations of possible alternative behaviors are compared and the most suitable one for fulfilling the current goal is selected.

At all these levels, the concept of *emotion* – in all its variants – can be of great use for the evaluation and action selection process. According to Arbib and Fellous (2004) a key function of emotion is to communicate simplified but high impact information.

Simple signals with no or only very little semantic content, like the *alarm signals* of the reactive layer, can be viewed as the starting point for emotions. They are the most rudimentary form of an affect, communicating among the parts of a system the occurrence of a special situation and preparing them for the release of an immediate, coordinated action. For an “alarm system” to work in a reliable way, there must be some characteristic sensor inputs for each alarm condition. Otherwise, those conditions cannot be detected and set off correctly in time.

In biology, creatures can reduce the probability of being surprised by an unpleasant situation by drawing their attention to a possible source of danger as soon as the first hints of its occurrence show up. Whether in nature or in engineering, resources are always limited and not just processing time. Therefore, a good mechanism for *attention allocation* can be very useful, if not essential. To focus the system's attention on the currently most important aspect of the environment helps to reduce the large amounts of sensory data continuously streaming into the system. In this case, “emotions” can be functionally seen as state-evaluative processes that adjust the relative weighting of different behavioral modes, raising the urgency level for one mode while lowering it for others, depending on the context. Again, this shift of mode is triggered by some stimuli, but here things are more complex than in the purely reactive case. First, a complete routine sequence of actions can be activated with one single stimulus. Second, emotions on this level are no longer simple signals any more. Instead, there is a more elaborated control loop, based on various *drives* or *motivations*, each of them possessing a range of possible values, indicating the respective current activation level. Depending on this, some modes of behavior are favored compared to others. Successful behavioral commitments are rewarded by an intensifying mechanism – referred to as “lust” – inappropriate ones are punished with “pain”, which leads to their avoidance. This mechanism introduces internal “sensations” into the system. They can be compared to sensory inputs stemming from the outside, only that they are providing the system with information about internal constraints, not external.

At the level of reflective processing, the behavior selection control loop is even more sophisticated. External and internal states are monitored, evaluated and associated to each other. Some aspects of these processes have already been addressed in section 4. This is because the stages of perception, representation and evaluation cannot always be clearly separated and run through in a subsequent way. They are continuously updated in a cyclic way and influence each other. So, the symbolic representation of the currently perceived scenario is compared with former scenarios stored in an *episodic memory*. Additionally, knowledge from a *semantic memory* is taken into account. Thereby, not only information which is currently active influences the scenario evaluation process, but also general knowledge about the relations of the world. The aim of the comparison process is to find similar previous

situations. Again, there has to be a mechanism narrowing the focus of the ongoing processes. Important features have to be highlighted, unimportant ones ignored. Not every detail of the potentially available data is considered. The mechanism looks for characteristic forms, and after having detected some of them, the rest is associated by using previously memorized scenarios.

Moreover, memories are not stored neutrally. Instead, every memory is afflicted with an “emotional” value. These ratings help to guide the evaluation process. They enable a simple, short-cut form of cognitive reasoning. To equip the system with emotional ratings allows it to distinguish between “desirable” and “harmful” situations and behaviors without the necessity of having all of them explicitly experienced or carried out before. It is a mechanism for a rough, predefined rating and selection of necessary actions, which works with some probability within a variety of complex and unforeseen situations.

When talking about “desirable” and “harmful”, these terms do not necessarily have to refer to the system *itself*. They can, and will in most cases, refer to the tasks the system has to fulfill. For example, if it has to take care of a person, as in some of our applications, every situation putting the protected person into danger will be classified as “not so good” and get a negative emotional rating.

6. IMPLEMENTATION

The system, as we have pictured it in the previous sections, contains several different modules, which have to cooperate. The central component is a database, which acts as an information sink for all kinds of information. For once there is a database that collects all data that is generated by the sensors. This is necessary because we assume that we are not able to process all data in real-time, at least not in the prototype implementation. To free development of the requirement of timeliness, all sensor data is stored in one database scheme. There is a software module that takes this sensor data and creates μ Symbols and symbols. These μ Symbols and symbols are also stored in a database scheme (logically this is another database that has no knowledge of the sensor database; just for reasons of administration, both are running on the same physical machine). Because permanent polling of the database for new sensor data is impractical, we have introduced a module that receives updates, writes it to the database and at the same time informs all registered modules about the new data. This way we avoid polling, reduce data traffic, but still maintain consistency, since the database is still the central data storage. This module is transparent to applications, since they only receive data updates, no matter if they were polled or pushed.

Aside of the update pushing the modules can obtain the current and historical state of the world representation by using database queries. This way we get a defined and decoupled interface between the different modules. Perception is done separately for

each of the four applications mentioned in section 2. This will have to be changed and the perception module will generate a common set of symbols, which can be used by all applications.

Since the system shall be able to deal with a big amount of sensor information, we have to provide the system with this amount of information. Unfortunately a real world installation with such a great number of sensors is currently not feasible. Therefore, we have decided to introduce a *simulator* (Rösener, *et al.*, 2004). This module serves two purposes: For once, it is able to generate a virtual environment, where sensors can be installed in big numbers. This environment is then filled with objects and persons that interact with the environment. The simulator calculates these interactions and creates virtual sensor data out of it. This data is stored in the database, so that other modules notice no difference between simulated and real world data. The second purpose of the simulator is visualization: It shall not only display the virtual environment with all its ongoing actions, but also visualize the system's world representation of this virtual environment. This way we can judge the quality of the world representation and detect gaps between what the system "perceives" and how the world "really" is.

7. CONCLUSION

We have presented a system that is envisioned to bring automation to the next level by employing bionic models for the control of a technical automation system. The concepts for this system have been defined, and a first implementation is currently in prototype phase. Using existing algorithms for classical image processing and other basic functionalities of the system, we are able to rely on state of the art technology to build a system that implements perception, scenario recognition and evaluation by the means of "emotions" to judge and qualify scenarios.

The steps for the near future include an increase in predefined symbols to extend the perception of the system. The current set is very limited and will be enhanced. Moreover, the existing symbols need to be extended by adding more properties and more associations. Following this, we need to find means to automate symbol generation and association. Currently this knowledge is "hard coded" into the system, but another group of the ARS project is currently doing research on this topic.

Additionally, it will be necessary to introduce context-dependent perception. So far, there is no influence between perception and higher-level mechanisms of the system – the system always perceives the same, no matter what its current goals are. We know however that perception in humans is strongly influenced by the current goals and we will have to model this.

What will remain is that the system will make errors when confronted with unforeseen events. As long as we do not introduce, for example, dogs into the

system, it will always mistake a dog for something else (a person) or simply ignore it. The same applies, if a person quickly changes its outer appearance. The system will mistake this person for a new person, assuming that it strongly relies on color histograms. This behavior is intended and can also be found in humans – only on a much more sophisticated level.

8. REFERENCES

- Arbib, A. and J.-M. Fellous (2004). Emotions: from brain to robot. *TRENDS in Cognitive Sciences*, **8**, pp. 554-561.
- Brainin, E.; D. Dietrich, W. Kastner, P. Palensky, C. Roesener (2004). Neuro-bionic Architecture of Automation – Obstacles and Challenges. In: *IEEE Africon 2004, Gaborone, Botswana*. pp. 1219-1222.
- Brenner, A. (2001). *Performanceverbesserung bei der automatisierten Einzelsortierung mit Zeilenkameras in Echtzeit*. Dissertation at the Vienna University of Technology, Institute for Automatisierungstechnik.
- Dietrich, D., W. Kastner, H. Schweinzer (2004). *Wahrnehmungsbewusstsein in der Automation - ein neuer bionischer Dekansatz*. In: *Automatisierungstechnik*, **52**, 3, pp. 107-116.
- Damasio, A. (2003). *Looking for Spinoza: Joy, Sorrow, and the Feeling Brain*. Harvest Books.
- Hauser, K., A. Bittleston, A.D. Gibson, W.B. Forward (1995). *Kaspar Hauser Speaks for Himself: Kaspar's Own Writings*. Camp Hill Press.
- Johnson-Laird, P.N. (1988). *The Computer and the Mind: An Introduction to Cognitive Science*. Harvard University Press.
- LonMark (2005). *LonMark Magazine, European Edition*. TEMA Technologie Marketing AG, Germany, **2**, 4.
- Pratl, G., W. T. Penzhorn, D. Dietrich, W. Burgstaller (2005). Perceptive Awareness in Building Automation. In: *Proceedings of the 2005 IEEE International Conference on Computational Cybernetics (ICCC 2005), Mauritius*.
- Tamarit, C., D. Dietrich, K. Dimond, G. Russ (2001). A Definition and a Model of a Perceptive Awareness System (PAS). In: *Proceedings of the IFAC FeT 2001*. pp. 177-182.
- Harnad, S.: The Symbol Grounding Problem. *Physica D* **42**. pp. 335 – 346, 1990.
- Ortony, A., D.A. Norman and W. Reville (2005). Affect and proto-affect in effective functioning. In: *Who Needs Emotions? The Brain Meets the Robot* (J.M. Fellous and M.A. Arbib (Ed)). New York, Oxford University Press.
- Riedl, R. (1984). *Biology of Knowledge: The Evolutionary Basis of Reason*. John Wiley & Sons.
- Solms, M. and O. Turnbull (2002). *The Brain and the Inner World. An introduction to the neuroscience of subjective experience*. Karnac Press.