The visual active memory perspective on integrated recognition systems

C. Bauckhage a,*, S. Wachsmuth a, M. Hanheide a, S. Wrede a, G. Sagerer a, G. Heidemann b, H. Ritter b

a Faculty of Technology, Applied Computer Science, Bielefeld University, P.O. Box 100131, 33501 Bielefeld, Germany
b Neuroinformatics Group, Faculty of Technology, Bielefeld University, P.O. Box 100131, 33501 Bielefeld, Germany

Received 2 August 2004; received in revised form 23 June 2005; accepted 18 August 2005

Abstract

Object recognition is the ability of a system to relate visual stimuli to its knowledge of the world. Although humans perform this task effortlessly and without thinking about it, a general algorithmic solution has not yet been found. Recently, a shift from devising isolated recognition techniques towards integrated systems could be observed [Y. Aloimonos, Active vision revisited, in: Y. Aloimonos (Ed.), Active Perception, Lawrence Elbaum, 1993, pp. 1–18; H. Christensen, Cognitive (vision) systems, ERCIM News (April, 2003). 17–18]. The visual active memory (VAM) perspective refines this system view towards an interactive computational framework for recognition systems in human everyday environments. VAM is in line with the recently emerged Cognitive Vision paradigm [H. Christensen, Cognitive (vision) systems, ERCIM News (April, 2003). 17–18] which is concerned with vision systems that evaluate, gather and integrate contextual knowledge for visual analysis. It consists of active processes that generate knowledge by means of a tight cooperation of perception, reasoning, learning and prior models. In addition, VAM emphasizes the dynamic representation of gathered knowledge. The memory is assumed to be structured in a hierarchy of successive memory systems that mediate the modularly defined processing components of the recognition system. Recognition and learning take place in the stress field of objects, actions, activities, scene context, and user interaction. In this paper, we exemplify the VAM perspective by means of existing demonstrator systems. Assuming three different perspectives (biological foundation, system engineering, and computer vision), we will show that the VAM concept is central to the cognitive capabilities of the system and that it leads to a more general object recognition framework.

© 2006 Elsevier B.V. All rights reserved.

Keywords: Cognitive vision; Contextual reasoning; Fusion; Architecture; System integration

1. Introduction

In traditional computer vision systems, the processes of model acquisition and recognition are decoupled in time and control. First, the system is provided with object models, which may be either handcrafted or learned from a sufficient amount of training data. Then, these models are used to accomplish a specific vision task. Although it works in constrained scenarios, there are several shortcomings, which prevent using this approach on a broader scale. If computer vision systems are to be applied in less restricted settings such as domestic or office environments, the drawbacks of the traditional approach are: (i) the impossibility of designing a complete model of possible objects, activities, and actions beforehand; (ii) the inability to deal with dynamic as opposed to static environments, i.e. the system continuously perceives new image data instead of processing a finite set of static images; (iii) the need to characterize previously unseen objects, so that pure identity recognition is insufficient; (iv) the system has to solve ad hoc tasks and not a mere pre-specified single task.

While there has been a lot of work on single aspects of more general computer vision systems (e.g. generic object recognition [3], contextual object recognition [4], perception action cycle approaches [1,5], or one-shot object learning [6]), less effort was spent on integrating different techniques in order to achieve robust performance needed in complex settings. Early work in this area was reported by Hanson and Riseman [7]. Similar approaches which, however, do not consider learning of new concepts include
Image retrieval systems constitute another application of computer vision techniques where ad hoc tasks have to be processed [10]. Here, relevance feedback and adaptation on the feature level of image processing have emerged as key concepts. However, it turned out to be very difficult, if not impossible, to close the semantic gap between user concepts and image features. A quite promising approach to learning grounded representations of the world was proposed by Roy [11]. In his DESCRIBER system, word semantics result from processing speech and image data in parallel. The system even realizes a first step towards learning more structured phrase descriptions. The recently proposed Cognitive Vision System paradigm [2] aims even higher. It claims that constructing integrated systems that are embedded in the world, interact with their environment to gather knowledge and articulate their knowledge by changing the state of the environment is essential for solving computer vision tasks in unconstrained real-world environments. In this context, cognition is defined as ‘generation of knowledge on the basis of perception, reasoning, learning and prior models’.

Following these general ideas of cognitive systems, a couple of projects have been started during the last years. Some of these also focus on system integration issues. ACTIPRET [12] aims at establishing a quality of service principle by formalizing the visual capabilities of system components. In CAVIAR [13], an XML computer vision markup language is defined that provides global data type definitions for interacting components. Crowley et al. [14] propose a software architecture for observing and modeling human activity. The approach is based on an ontology for contexts and situations. The fundamental component for the software architecture are observational process that consist of a control and a transformation subcomponent. Most of these approaches purely focus on the process aspect neglecting data-centered aspects of model acquisition. Control is hierarchically organized potentially leading to long communication paths from one branch of the hierarchy to another one.

In the following, we present a cognitive vision framework that facilitates the construction of computer vision systems for complex everyday environments. The Visual Active Memory (VAM) approach couples model acquisition and recognition processes. It evaluates, gathers and integrates contextual knowledge for visual analysis. It interacts with its environment through annotation and retrieval processes that follow the human-in-the-loop principle. The VAM approach is motivated from different perspectives that are essential for the design of cognitive vision systems: (i) biological principles of cognition, (ii) computer vision and pattern recognition, and (iii) system engineering. We will show that constraints from all three perspectives mutually complement each other and lead to the proposed VAM framework.

In an ongoing research project, the VAM approach is applied in developing a memory prosthesis system which uses augmented reality techniques for tight human–computer interaction. Two demonstrator systems realized so far, work in an office environment (see Fig. 1). First results show that, by using the VAM framework, system components are easy to integrate and complex recognition tasks can be solved.

2. The visual active memory perspective

An understanding of cognitive systems is a shared and interdisciplinary goal linking the fields of biology, psychology, computer science, pattern recognition, and robotics. While there are many different definitions of cognition, even within these disciplines, they all agree that humans are prototypical cognitive systems with an amazing performance. The VAM perspective aims at constructing systems that are able to perform vision tasks in everyday environments of humans. Moreover, visual active memories should be able to communicate such tasks to human users while the tasks are specified, executed, and completed. Although it is strongly oriented towards cognitive systems, we will not try to prove if any definition of cognition is met by our framework.

2.1. Biological foundation

What are the key concepts of cognition and what enables an artificial system to reach a performance similar to biological cognitive systems? One interesting insight on this topic comes from theories of how cognitive abilities could have
evolved from early reactive systems. A second insight is motivated from neurophysiological studies of the human brain.

Cruse [15] sees an important incitement of the development of cognition in redundant tasks that involve a high number of degrees of freedom (DoF) which are only partially determined by sensory input. The ability of a system to cope with such redundant situations is often called autonomy and is frequently modeled by a separate deliberative or cognitive layer that steers a reactive layer which is strictly sensor driven [16]. However, Cruse argues that even the cognitive layer can be modeled in a sensor-driven way using the same representation as the reactive layer. The key concepts are: (i) a dynamic representation or manipulable world model that responds to a given sensory input on a longer time scale than lower level modules and can be used to play around with it; (ii) the principles of multiple computations and relaxation towards a consistent result; (iii) a common coding of perception and action leading to a representation of the mechanism rather than the explicit state; (iv) the possibility to decouple the representation from the external world for consolidation and planning reasons.

Although the functionality of the human brain is not understood in detail, neurophysiological studies support several general assumptions. Memory is a time-dependent process that can be divided into a short-term and long-term memory. Based on content, the long-term memory is structured into different memory systems that are assumed to form a successive hierarchy where the content of the higher system is at least partially grounded in lower systems (SPI-model) [17]. Storage is performed through this hierarchy. However, accessing memory content is independent from the storage mechanism. Retrieving information from a memory does not depend on the state of the memory at storage time. The stored items of the memory systems are dynamic. They change through the principle of re-encoding if they are frequently accessed. The episodic memory is assumed to be the most complex system. It stores autobiographic events that are embedded in the dimensions of time and space. Stored events are typically bound to emotional contents that provide a selective access to the episodic memory.

2.2. System engineering

Developing complex vision systems is not only a matter of conceptual design but also a software engineering task. Within the VAM framework, this challenge is even more complicated because the VAM approach has to serve different application as well as dynamically specified ad hoc tasks. Learning processes are performed over a longer period of time so that the persistent storage of processing results is becoming much more important. Therefore, a data-centered approach as opposed to a task-centered approach must be chosen. Furthermore, rapid-prototyping must be possible to meet typical demands in research projects [18]. Interfaces must be easily extensible and independently developed vision modules must be easy to integrate. A communication framework must be provided for distributed and asynchronous processing in order to guarantee real time performance. The visual memory is an active component that allows a dynamic representation that is consolidated over time by internal processes. Therefore, appropriate trigger mechanisms must be provided. As a consequence, database technology plays a much more important role in system design of cognitive vision systems and has a significant impact on the design of a VAM.

2.3. Computer vision

Generic object recognition has always been a long-term research goal and it is still far from being achieved. The aim of the VAM approach is to provide a computational framework that addresses important facets of this general goal. The coupling of model acquisition and recognition processes has important consequences on visual processing. Object recognition and scene reconstruction have to be performed on different levels of abstraction and object models have to be acquired on different time scales. An object that has never been seen before has to be tracked based on a one-shot model acquisition procedure. Based on this correlated image sequence more invariant models can be learned so that the individual object is recognized independent of temporal or local constraints. If more image data is collected for this object, further improvements can take place, so that object recognition can become increasingly independent of background clutter or occlusions.

If the robustness of recognition results is to be increased by using processes on higher levels of abstraction, the dynamic and static context of an object play a key role in categorization. For example, a before unseen cup which is placed in an environment where cups are likely to be found and, moreover, is used in a way cups are being used, can be categorized as a cup. In fact, object recognition, activity or action recognition, and classification of the scene context form a triple where the components influence each other.

Another key concept for selective processing and model acquisition is attention modeling. This aspect is twofold. On the one hand, attention can be extracted bottom-up by looking at interesting image regions; on the other hand, it can be determined by the user through a human–machine interaction, e.g. by means of pointing for view selection.

2.4. Cognitive vision as an interactive memory process

Pattern recognition in complex and in most instances unknown environments is a highly redundant task [19]. In accordance with the cognitive vision paradigm, the VAM approach proposes a system-oriented perspective that utilizes biological and psychological principles in order to solve such tasks.

The coupling of model acquisition and recognition processes leads to a hybrid approach integrating data-centered (persistent memory) and process-centered (consolidation and re-encoding) principles that realizes a dynamic representation of the world. The system is able to play around with
memory content by triggering different processing modules that act on the memory. Each vision algorithm is loosely coupled to the database through a flexible communication framework. This makes it easy to integrate different recognition modules and to realize the biological principle of multiple computations. Each module has its own view onto the database and accesses it in an asynchronous way.

The database is structured into different memory systems (pictorial, feature-based, episodic, categorial) that are organized in a successive hierarchy. The recognition of previously seen objects, activities, and scene contexts leads to a storage operation in all memory systems and, on a longer time scale, a re-encoding of learned recognition models and concepts.

The content of the memory is consolidated by fusion of different recognition results and contextual analysis. In the VAM framework, we even add a fourth component to the triple of objects, activities/actions, and scene contexts that is based on the human-in-the-loop principle. The user interaction is twofold. First, a user starts retrieval tasks on the memory, gets results back, and reacts on those results by formulating a new retrieval task. Thereby, labels can be introduced in a top-down manner that help the system during categorization. Secondly, the user interacts with the world by changing view points, pointing, or moving objects. Using augmented reality equipment including mounted head cameras, inertial sensors, and an active display, these actions are directly fed back to the system (see Figs. 1(b) and 10).

3. System architecture and components

Fig. 2 shows the conceptual architecture of our system. In the center, we recognize the memory component. On its lowest level, image data (i.e. patches cropped from images) is stored. The higher levels contain feature based object descriptions, abstract models of observed events, and categorial representations. Several computational modules are grouped around the memory. Note that some of the building blocks represent several algorithms running in parallel.

3.1. VAM infrastructure and communication framework

Since it is very flexible and suited for abstract concept descriptions, XML was chosen to describe content stored in the memory. Correspondingly, a native XML database [20] provides the basic infrastructure for the VAM. On the top of this, a software layer was implemented that allows for interconnected information fragments and enables powerful insert, query and update functionalities. This layer is used by a runtime environment where memory processes like forgetting or information fusion are realized. Furthermore, a subscription model for distributed event listeners was implemented, so that memory events can trigger registered processes thereby the memory allowing to become active (see Fig. 3). The VAM realizes a repository-style architecture. The trigger and event mechanisms can be used to declaratively define control mechanisms that are bound to memory elements. Therefore, the system is not hierarchically controlled in a top-down fashion, but the components are coordinated through the active memory. This leads to a flexible framework supporting asynchronous processing without communication overhead.

In order to guarantee real-time performance, we decided to realize the VAM as a distributed system. A thorough comparative study of existing framework technologies for distributed system integration revealed that by now there is no solution tailored to the needs of cognitive vision [21]. As most vision researchers are not trained in using middleware, the use of CORBA, for example, was ruled out because of its complexity and bloated standardization. Rather, owing to the academic background of this work, an integration framework for an agile software process (cf. [18]) is needed.

This led to the development of an XML enabled communication framework (XCF) [22]. It is based on the internet communication engine (ICE) [23] and provides an easy use middleware for building distributed object oriented systems. XCF features a pattern based design and offers communication semantics like streams, remote procedure calls and event channels. Similar to the data storage in the VAM component, data exchange between different mod-
ules is based on XML. Since interfaces are specified using XML schemata, runtime type safety is ensured and interface programming is intuitive even for middleware novices. In combination, the XML based memory infrastructure and the XCF framework enable us to realize architectures of very loosely coupled components. This decoupling and the capability of the memory to asynchronously gather and provide information yields a high robustness against component failure.

3.2. Object recognition, information fusion and online learning

Currently, our system employs two different appearance based methods for object recognition. On the one hand, a VPL classifier as introduced by Heidemann et al. [24,25] is applied. This approach was motivated by biological information processing principles which are believed to underlie early visual processing in the brain. It combines a receptive field layer, saliency processing, PCA-dimension reduction and local linear map neural networks for object classification. The LLM classifier is trained using labeled samples. On the other hand, our system makes use of the well-known cascaded weak classifier approach [26,27]. In order to provide a basis for higher level processes, such as action recognition, several cascaded classifiers were trained beforehand, so that objects typically found in everyday environments would be recognizable upon starting the system.

As both perceptive modules typically do not yield perfect recognition rates, the VAM stores incoming recognition results not as irrevocable facts but as hypotheses. Fig. 4 shows an example of a hypothesis in XML representation. Hypotheses are labeled with a reliability value. Memory processes that adopt reliabilities according to the scene context will be discussed in Section 3.3. Dealing with object labels resulting from different classifiers, a memory process for information fusion permanently compares the hypotheses produced by the two recognition algorithms. 3D coordinates of object hypotheses are estimated relative to an artificial target that is placed somewhere in the room. When it is in the view of the user, the artificial target is used to compute the pose of the camera which is continuously tracked [28]. Currently, we assume that the user deals with objects that are lying on a table. This restriction is used in order to compute the 3D coordinates of an object. A more general approach using stereo cameras will be integrated in the near future [29]. Based on these 3D object coordinated spatio-temporal similarities of incoming hypotheses can be computed. Assume a hypothesis to be a triple $h = (x, t, o)$ where $x$ denotes the spatial coordinates, $t$ the current time and $o$ the type of an object. Then, the probability that two hypotheses $o_a$ and $o_b$ resulting from classifiers $a$ and $b$, respectively, refer to the same object is modeled by the distance

$$P_{h_a, h_b} = e^{-d^2((x, t, o_a), (x, t, o_b))} \delta(o_a, o_b).$$

here, $\delta$ is the Kronecker function and $d$ is the variance normalized Mahalanobis distance. Corresponding to the biological principle of multiple computations, hypotheses with small spatio-temporal distances will be fused so that a robust interpretation of the current scene content is obtained. Though rather pragmatic, this approach is appealing for it is efficient but rests upon a broad sensorial and empirical basis which automatically takes into account the performances and peculiarities of the underlying recognition processes.

Using a VPL classifier provides an avenue to a closer coupling of learning and recognition. VPL classification consists of three stages. First, vector quantization is applied (V-step) to obtain a raw partitioning of the input data. After assigning an incoming image patch to one of the clusters, PCA is locally applied to extract suitable features for classification (P-step). This simple realization of local PCA enables fast training and avoids manual tuning of training parameters. Finally, several local linear maps

![Fig. 3. Example of a coupling between memory processes in the visual active memory.](image)

![Fig. 4. XML memory hypothesis example.](image)
(L-step) project the extracted features onto a probability vector. Each component of this vector corresponds to an object known to the system. The final classification result is the object with maximum probability. As it can be trained in two different ways, this classification scheme is well suited for the task of online learning. Applying the human-in-the-loop principle, the VAM can ask for labels of highly salient image regions which do not depict a known object. Whenever novel views of a (new) object were acquired and should be quickly incorporated, the first two stages of the VPL classifier are left unchanged. This assumes that the local feature spaces available so far cover the appearance of the new object. Only the neural classifier is retrained and—in case of a novel object—its output dimensionality will be increased. A full retraining of all three processing stages (which is much slower) can be performed later on. An example of this online training mechanism will be presented in Section 4.

3.3. Action recognition and contextual reasoning

A comprehensive interpretation of events in human inhabited environments requires components for human activity recognition. Since action recognition deals with dynamic processes, different issues of contextual coherence have to be considered. While the situational context of an action describes its preconditions and its effects on the scene, the spatial context relates trajectories of hands or people to objects. Also, different temporal characteristics must be brought into a consistent overall picture. Environmental changes or discrepancies between object and action recognition must not lead to hypotheses contradicting the current memory content.

If human activities are considered, skin color detection is a major building block for many operations. In our current system, gesture and action recognition rely on adaptive skin colored region detection based on Gaussian mixtures models [30]. Skin colored image patches are forwarded to a corresponding VPL classifier which decides whether they depict a hand or even a pointing gesture. Actions involving objects are recognized using an extension of a particle filter approach to hand trajectory classification introduced by Black and Jepson [31].

In [30,32], Fritsch demonstrated how to incorporate contextual knowledge into particle filter-based action recognition. The situational context is applied in selection step of the filter where it initializes and selects those samples whose preconditions match the current situation. In order to cope with spatial context, context areas are defined which indicate image areas where objects potentially relevant for a specific action can be expected. This realizes an attention mechanism which not only indicates where symbolic context is expected but also allows to specify what kind of context is required. The spatial context \( \Theta_t \) guides the update step and modifies the weights \( \pi_s^{(t)} \) of samples \( s^{(t)} \) that match the observations \( z_t \). The calculation of sample weights is thus extended by a context factor \( p_{\text{symb}} \) that indicates how well the observed scene fits the expected symbolic context:

\[
\pi_s^{(t)} \propto p(z_t|s^{(t)})p_{\text{symb}}(\Theta|s^{(t)})
\]  

(2)

The value of \( p_{\text{symb}} \) depends on whether the expected context object is present or not. To train the action recognizer, hand trajectories from different videos were averaged and information on object context was annotated manually.

The risk of conflicting object and action recognition results is minimized by means of memory processes which rate stored and incoming hypotheses according to contextual and functional relations. As they easily integrate different types of information, we apply Bayesian networks to model expectations for the relations between different types of hypotheses. This approach is appealing because it is applicable to any functional context. It allows for top-down as well as for bottom-up control and, as described in [33], a Bayesian network based representation of contextual knowledge can guide object recognition and scene understanding. Given the evidences \( e = (e_1, \ldots, e_m) \) of variable assignments in a Bayesian network, conflicting memory content is detected using a conflict value \( \text{con} f \) as defined in [34]:

\[
\text{con} f(e) = \log \frac{\prod_{i=1}^m P(e_i)}{P(e)}
\]  

(3)

The conflict measure is positive if the expected relation between evidences does not fit the data. In this case, the joint probability of the contextual model is lower that the apriori probability of the independent recognition results.

Uncertainties, which are inherent to the involved perception processes, are incorporated using \( s \) for the observable nodes. They are set according to the reliability of the corresponding hypothesis. The more reliable the hypothesis is, the harder is its evidence. If a conflict is detected, the memory process lowers the reliability of the involved hypotheses which, in turn, decreases the conflict value.

Several contextual and functional models are stored in the VAM and are independently applied in the contextual reasoning process. As an example, Fig. 5(a) shows a Bayesian network our system applies to represent contextual pre-requisites for a ‘typing’ action. Dealing with input images as shown in Fig. 5(b) and (c), ‘typing’ hypotheses must be doubted if they do not comply with the current scene context. Inferring a computer, for instance, requires to observe at least some of the objects keyboard, mouse and monitor. The probabilities for the conditional dependencies shown in the tables in the figure were estimated from correctly preprocessed video data. If all nodes of a network were observable, parameter estimation simply resulted from counting the different configurations. Otherwise, if some variables cannot be observed, an EM-algorithm was applied as proposed in [35].

A further example of a model distinguishes two different field of view contexts (Fig. 6). Such contexts are inferred from typical object constellations. Here, a computer desk and a desk for doing other paper work are distinguished.
4. Demonstration of system results

We have tested the applicability and reliability of our VAM concept in two complementary experimental settings. In the static setting shown in Fig. 1(a), a human acting in an office environment is monitored from different camera views. In the mobile setting depicted in Fig. 1(b), the user is self-monitored through a head-mounted augmented reality (AR) gear consisting of two cameras and a display. Next, we present exemplar results and discuss experiences with these demonstrators.

As shown in Fig. 1(a), the static VAM demonstrator analyzes video signals from two calibrated static cameras which monitor an unconstrained office environment; one of the cameras provides a side-view of the scene the other monitors it from above. Fig. 7 exemplifies the process of action recognition from side view images. At each time step, the sample set of the particle filter contains several hundred instances of ten different action models. The curves at the bottom of the figure visualize the summed probabilities of each model. At the beginning of the sequence shown here, the particles representing the action ‘grasp book’ had the highest overall weight; shortly afterwards the weights for the instances of ‘book taken’ exceeded a certain threshold and the action was believed to be observed. Currently, complex actions like ‘drinking from a cup’, ‘browsing a book’, ‘phoning’ or ‘typing on the keyboard’ can reliably be recognized. A test with 420 sequences yielded an accuracy of 93% [30,32].

To evaluate our consistency validation approach, Bayesian network models of functional dependency concepts for different constellations of typical office objects and actions were defined. Fig. 8 shows how the content of the memory evolves over a video sequence. Note that at the end of the sequence the field of view classification gets into an inconsistent state because only a ‘keyboard’ is detected in the visible scene. Such kind of conflict states trigger the dynamics of the consistency validation process. Fig. 9 depicts an performance example for these kind of dynamics using the ‘typing’ functional dependencies known from Fig. 5. Initially, hypotheses for a monitor and a typing action were generated but no keyboard was detected so that a contextual requirement was violated. Weakening hypotheses by lowering their reliability measure can solve such conflicts. In our example, both hypotheses had an initial reliability of 1. Due to the conflict, both reliabilities were scaled by a factor of 0.9 which in turn leads to a decrease of the conflict value (Eq. (3)). Since, for several time steps, no other
evidence was produced by the perceptive modules, this process continued until the conflict value became negative thus indicating that there was no conflict anymore. In our example, this also meant that the reliabilities of the involved hypotheses dropped below 0.5. Consequently, they were discarded from the memory.

Fig. 10 exemplifies online learning of new objects in the mobile selling. Working in an office environment, the user wears a head-mounted device which is equipped with two cameras and a display (see Fig. 1(b)). The scene recorded by the cameras is displayed to the user; on the right hand side of the field of view, a control menu for interaction with the system is cast into the display. If switched to ‘record’ mode (Fig. 10(a)), the system records views of the salient region in the center of the field of view. Here, the human-in-the-loop paradigm displays its full potential. By focusing on the object, the user guides the system’s attention. By manipulating the object, the user closes the perception action cycle and the system is able to acquire a wide range of views. As a result, even articulated objects like the keys in this example can be learned in an ad hoc manner. After the user provides an object label, the VPL classifier is retrained (see Fig. 10(g)).

1 For details on user–system interaction, please refer to [36].
and (h)). An evaluation with up to 10 objects learned interactively yielded an average recognition accuracy of 95% [37].

5. Conclusion

The visual active memory perspective aims at the construction of vision systems for unconstrained everyday environments. It strives to adhere to biological foundations of cognition as well as to comprise principles from the perspectives of system engineering and computer vision. We presented an XML-based infrastructure and communication framework that provides the functionality for coupling model acquisition and recognition processes. Contextual reasoning is realized as an active memory process that is triggered by memory events and consolidates memory contents. The VAM approach matches biological principles in that it is structured in a successive hierarchy of memory systems, provides a dynamic representation through triggered memory processes that play around with memory content, and supports distributed processing, as well as an easy integration of new modules which are prerequisites of multiple computations.

We presented components that have been integrated in two demonstrator systems. The systems are capable of robust data-driven object tracking, integrating different approaches to appearance-based object recognition, fusing of object recognition results, action recognition, and con-
textual reasoning. Attention mechanisms have been realized by spatial contexts in the action recognition module, salience values for object detection, and interpretation of pointing gestures performed by a user. Visual object knowledge trained in an offline manner can be extended by using pictorial and episodic memory content, that is labeled by human–computer interaction using the augmented reality gear, data-driven object tracking, and recognition results of already learned detectors.

Given the proposed infrastructure and the capabilities of our first VAM demonstrators, interesting experiments will be possible soon. First of all, it will be interesting to let the system play around with processing components. Automatic selection of configurations with increased interpretation robustness may yield a first computational model to support Cruse’s evolution hypothesis [15] for higher cognitive processes. Also, it will be interesting to explore different fade-out dynamics of forgetting processes for this could lead to a computational model that simulates human-like visual learning. Both kinds of experiments, however, must be carried out with respect to the human in the loop and the usability of the system will have to be considered. Here, experience from earlier work on the evaluation of cognitive interfaces for advanced human–machine cooperation [38,39] provides promising guidelines.

References