ACT-CV: Bridging the Gap between Cognitive Models and the Outer World

Marc Halbrügge

Quality and Usability Lab, Deutsche Telekom Laboratories
Technische Universität Berlin, Germany
marc.halbruegge@telekom.de

Abstract

Cognitive modeling can provide important insights about the usability of software or hardware products at all stages of product design. It is not used very often, though, as the creation of cognitive user models is typically done by highly trained experts. These specialists need not only develop the models, but have to spend a lot of additional time on software interface problems: The cognitive models need to be connected to the interactive system under evaluation. This poster presents ACT-CV, a software library that aims at solving the interface problems of user modeling.

The Challenge: Modeling interactive Behavior

Cognitive modeling has proven more and more useful for the design and evaluation of human machine interfaces as it allows to predict usability metrics like the time to complete a task or the types of errors that may occur. Modeling relies heavily opon cognitive architectures like ACT-R (Anderson, 2007) which closely mimic the cognitive, perceptual, and motor capabilities of human users. Unfortunately, cognitive architectures tend to lock the modeler into highly specialized computer systems (Lisp in the case of ACT-R), creating significant hurdles for the application of a user model to the human machine interface in question (Byrne, 2013). Existing tools to overcome these hurdles either revert to mock-up prototypes of the interface in question (e.g. CogTool, John et al., 2004), require changes to the source code of the original software (e.g. AGIMap, Urbas et al., 2006), or search the computer screen for special pixel color combinations (Segman, Ritter et al., 2006).

The Solution: Computer Vision

The software library ACT-CV (Halbrügge et al., 2007) allows interaction between ACT-R and arbitrary software without the necessity of mock-ups or software changes. It uses computer vision for the identification of screen elements and adds support for time based visual parameters like optical flow. The graphic processing inside ACT-CV is handled by the open source framework OpenCV (http://opencv.org/) which provides state of the art machine vision on multiple platforms and includes more than 2500 individual algorithms.

Applications of ACT-CV

ACT-CV was successfully applied to research especially in the automotive domain, where it allowed to connect a cognitive model of car driving (Salvucci, 2006) to an already existing driving simulation (Halbrügge et al., 2008). This was the first time that this driving model was acting in the same environment as the human subjects that it was compared to.

62 Maisding
Wolfspring
Dornstadt
Ruckmars
500 m

Car driving experiment using ACT-CV.
Tasks: Maintain the lane position and react to periodically appearing road signs.

But the application of ACT-CV has also brought some weaknesses into light:

- High complexity

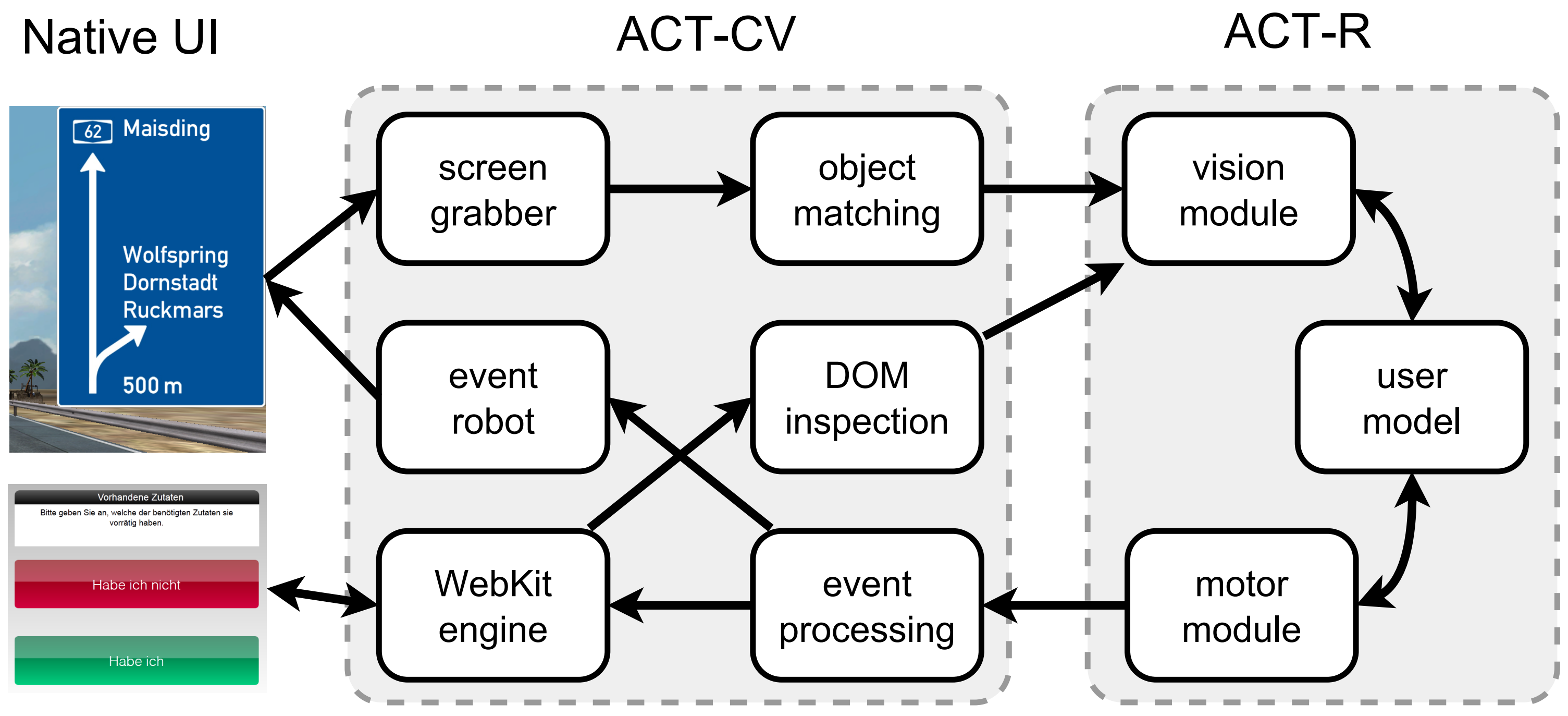
  The simple task of finding the position of a predefined visual element (e.g. button) on the computer screen can be accomplished by using ACT-CV out of the box and providing nothing but a picture file containing that visual element to it. But real world applications usually need more complex visual sensors, making the user of the library not only recompile it from source, but design his or her own machine vision code.

- Lack of text processing

  Most interfaces rely at least partly on written language. ACT-CV in its first incarnation comes with many sensors for colors and shapes, but it cannot decipher text. This is hard to work around, e.g. by feeding individual picture files of every single letter in the correct text size and color to the library.

ACT-CV 2 adds HTML5 Support

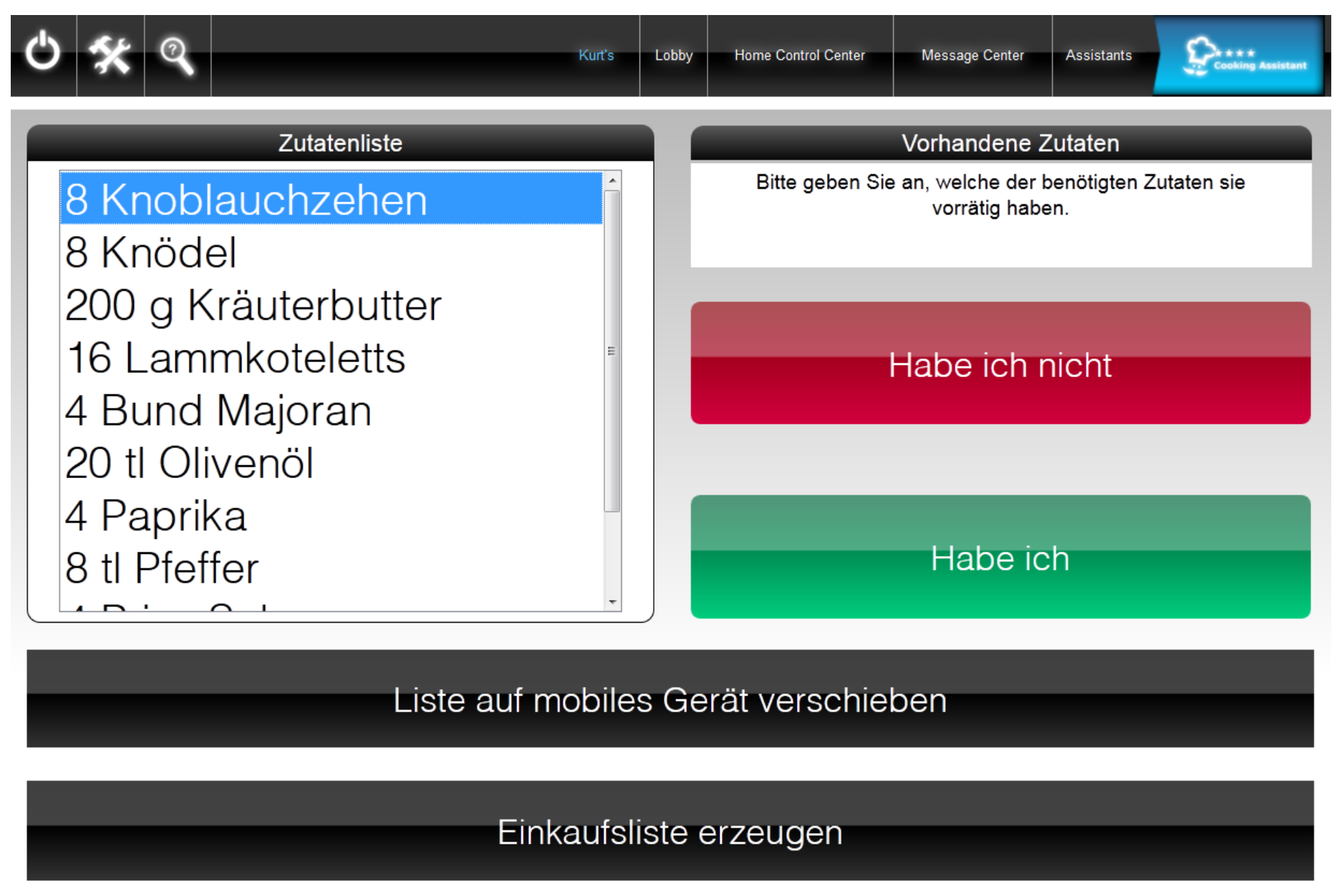
ACT-CV 2 is a major overhaul of the system. It simplifies the use of the library and at the same time addresses the lack of text processing in the previous revision of ACT-CV by adding HTML5 support. HTML5 marks the progression of HTML from a markup language for static text documents to a programming language for interactive user interfaces. It powers not only the web, but many native mobile applications, and is used as platform for rapid prototyping as well. ACT-CV 2 features a state of the art web browser that allows both human subjects and cognitive models to interact with HTML5 applications.



Signal flow of ACT-CV 2

Planned Applications of ACT-CV 2

The first application of ACT-CV 2 will be the usability evaluation of a cooking assistant that has been developed as part of the ambient assisted living project MASP (Blumendorf et al., 2008). ACT-CV will be used in conjunction and comparison with the MeMo workbench for automated usability evaluation (Quade et al., 2013).



8 Knoblauchzehen
8 Knödel
200 g Kräuterbutter
16 Lammkoteletts
4 Bund Majoran
20 tl Olivenöl
4 Paprika
8 tl Pfeffer

Bitte geben Sie an, welche der benötigten Zutaten sie vorrätig haben.

HTML interface of the MASP based cooking aid

Future Extensions: Saliency Maps

Future extensions of ACT-CV will include automatically generated saliency maps based on the graphical input to the system. This will form the basis for more accurate models of visual search, as needed for example to predict the probability to overlook a button in a given UI. The hybrid approach of ACT-CV 2 is well suited for this extension.

Conclusions

In order to use cognitive modeling during the usability engineering process, one has to combine the model on the one hand with the interactive environment of the software in question on the other hand – a technically challenging task. ACT-CV solves this by computationally generating symbolic representations of graphical user interfaces. This approach renders laborious tasks like manually annotating screenshots of the UI unnecessary and allows the modeler to fully concentrate on the cognitive user model itself.

Download ACT-CV 2 from http://act-cv.sourceforge.net

References

Anderson, J. R. (2007). *How can the human mind occur in the physical universe?* Oxford University Press, New York.

Blumendorf, M., Feuerstack, S., and Albayrak, S. (2008). Multimodal user interfaces for smart environments: the multi-access service platform. In *Proceedings of the working conference on Advanced visual interfaces*, pages 478–479. ACM.

Byrne, M. D. (2013). Computational cognitive modeling of interactive performance. In Lee, J. D. and Kirlik, A., editors, *The Oxford Handbook of Cognitive Engineering*, chapter 27. Oxford University Press, New York.

Halbrügge, M., Deml, B., and Neumann, H. (2008). A cognitive model of visual search on direction signs alongside the autobahn. *International Journal of Psychology*, 43(3/4):759.

Quade, M., Lehmann, G., Engelbrecht, K.-P., Roscher, D., and Albayrak, S. (2013). Automated usability evaluation of model-based adaptive user interfaces for users with special and specific needs by simulating user interaction. In Martin, E., Haya, P. A., and Carro, R. M., editors, *User Modeling and Adaptation for Daily Routines: Providing Assistance to People with Special Needs*, pages 219–247. Springer, Berlin / Heidelberg.

Salvucci, D. D. (2006). Modeling driver behavior in a cognitive architecture. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 48(2):362–380.