

Extending JSegMan to Interact with a Biased Coin Task and a Spreadsheet Task

Farnaz Tehranchi (farnaz.tehranchi@psu.edu)

Department of Computer Science and Engineering

Frank E. Ritter (frank.ritter@psu.edu)

College of Information Sciences and Technology

Penn State, University Park, PA 16802 USA

Keywords: Cognitive model; Cognitive architecture; Human-computer interaction; Simulated eyes and hands.

Introduction

To act more like a human, current cognitive models require a way to see and operate in the world. This methodology and access to a world as a requirement were discussed before, and several ways have been created. For instance, cognitive model interface management systems (CMIMS) based on user interface management systems (UIMS) were introduced to provide cognitive models with eyes and hands to interact with the same interface that users see (Ritter, Baxter, Jones, & Young, 2001).

We report on developments extending JSegMan, which supports architectures to interact with uninstrumented environments (Tehranchi & Ritter, 2017). JSegMan was improved, and new features for the visual module were added. JSegMan creates a way to interact with all interfaces using an extended Java library (Robot package) to input motor commands (keystrokes, mouse moves, and mouse clicks), and uses an open source library to help with image processing (Sikuli) based on *OpenCV*. JSegMan introduced visual patterns that are small images that represent the visual objects in cognitive architectures—visual chunks in ACT-R. JSegMan parses the screen and uses the *Template Matching* method to find the target, the visual pattern, and area. Template Matching is a pattern-matching algorithm that compares a template (small image) against the overlapped image regions (the computer screen) pixel by pixel; the area that has the maximum matching score is the target area. JSegMan can identify pre-defined patterns. Visual patterns are defined for the cognitive model similar to memory chunks (Tehranchi & Ritter, 2018).

JSegMan has been used with previous models. In all applications, JSegMan provides ACT-R (Anderson, Bothell, Byrne, Douglass, Lebiere, & Qin, 2004) cognitive models with eyes and hands interaction. After describing it, we report here two new interfaces it uses.

JSegMan and the Dismal Task

JSegMan (Tehranchi & Ritter, 2018) has been used along with the Dismal spreadsheet task in Emacs using an existing large ACT-R model (Paik, Kim, Ritter, & Reitter, 2015). It has 29 rules and 1,152 declarative memory task elements. JSegMan illustrates the missing knowledge in the original model because with JSegMan the effect of actions on the interface are visible and are trackable. For instance, we

found one missing click in the original model. JSegMan allows the model not just to model the task performance but actually to perform the task. It conducts a large, 20 min., non-iterated task with 14 subtasks. Also, most of the key press requests to the motor module required a hand/finger adjustment. These differences between the task requirements and the model's performance were visible because the results in the interface did not match the expected output. Matching behavior in the target interface can be an essential way to validate models in the future. We adjusted 162 declarative chunks in the original Dismal ACT-R model by adding a new slot for visual objects. Additionally, to model eye movements, we added 52 new visual objects and visual locations. The use of JSegMan also provided a better fit to the human data. The model with JSegMan predicted the response times more accurately while, importantly, using the same, unmodified interface that the human subjects used. The correlation improvement is not a reliable increase, but the difference in MSE was reduced by 47% (Tehranchi & Ritter, 2018).

JSegMan and the Biased Coin

The biased coin model is based on learning in a probability choice experiment in the ACT-R tutorial (Bothell, 2017). Figure 1 shows the experiment window implemented in ASP.Net. After seeing "Ready," the model either clicks on the "Head" button for heads or the "Tail" button for tails. The feedback indicating the correct answer, either "Head" or "Tail" is displayed; "Match" is displayed in green, and for incorrect answers "Wrong" in red is displayed. Figure 2a. illustrates a summary of the model. In this experiment, heads are the correct choice on 70% of the trials without considering the previous user choices. The ACT-R model begins with a 50% chance of heads and tails. Figure 2b shows its first choice was a tail. The model adjusts the head and tail probability based on what the model, with JSegMan eyes, can see on screen through utility learning. Finally, after 100 trials the model average responding for heads approaches 70%.

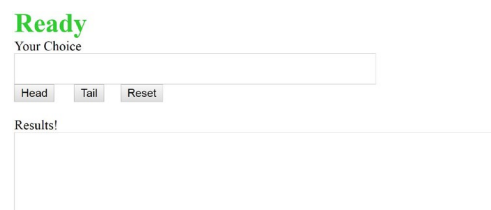


Figure 1. The biased coin interface used by JSegMan.

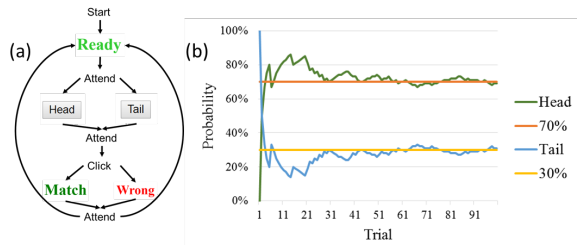


Figure 2. (a) The simplified flowchart used by the ACT-R model and patterns used by JSegMan. (b) The probability of choosing the head, green, and the tail, blue, over 100 trials.

We learned that due to the bidirectional communication what the model can see can affect the learning procedure.

JSegMan and the Excel Spreadsheet Task

We have started to extend JSegMan to interact with an Excel spreadsheet, shown in Figure 3, to perform the Dismal task, shown in Table 1 (similar to Paik et al., 2015). When compared to the existing dismal model of the task in an Emacs spreadsheet, this model will help explore the effect of the interface on task and requires JSegMan to be extended. We have (a) added a marker showing the area of attention's location (shown in Figure 4), (b) provided bidirectional communication between JSegMan and ACT-R model, and (c) reduced further limitations in cognitive modeling (e.g., models can directly interact with the same environment as a user and made it more realistic).

It remains to gather data on this version, collect more eye movement data to simulate attention shifts more accurately, model more of the task, and thus better predict human performance. Also, with JSegMan we can show whether the eyes follow the hands successfully.

	A	B	C	D	E
1	Command Name	Frequency	Normalization	Length	Typed Characters
2	log	20			
3	learn	6			
4	excise-chunks	12			
5	excise-task	5			
6	go	23			
7	help		13.7		
8	excise-all		5		
9	load		6.5		
10	excise		10.1		
11	time		17.3		
12					
13	Total	139	100		
14	Your Total				

Figure 3. The dismal task starts, on the left side, and on the right side, the final task state is in Excel.

Table 1. The dismal task subtasks.

Tasks
(1) Open File
(2) Save As
(3) Calculate Frequency (B7 To B11)
(4) Calculate Total Frequency (B14)
(5) Calculate Normalization (C2 To C6)
(6) Calculate Total Normalization (C14)
(7) Calculate Length (D2 To D11)
(8) Calculate Total Length (D14)
(9) Calculate Typed Characters (E2 To E11)
(10) Calculate Total Typed Char. (E14)
(11) Insert Two Rows
(12) Type In Name (A1)
(13) Insert Current Date (A2)
(14) Save As ...

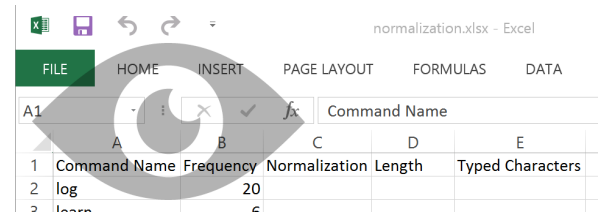


Figure 4. JSegMan's area of attention (the pupil) on an Excel spreadsheet.

Conclusion and Further Research

These interactive models and agents are essential for cognitive science and also important for agent-based modeling and computational organizational theories because they provide social agents that can interact with the world.

JSegMan is about adding the capabilities to model the details of visual, and motor modules for cognitive architectures, and what they can teach us about how the human mind works, how an analysis of vision and motor as they support and implement cognition might further our understanding of the human mind.

Acknowledgments

This work was partially funded by ONR (N00014-15-1-2275). We thank Dan Bothell for his assistance with the ACT-R tutorial.

References

- Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review*, 111(4), 1036-1060.
- Bothell, D. (2017). ACT-R 7 Reference Manual [Electronic Version] from act-r.psy.cmu.edu/wordpress/wp-content/themes/ACT-R/actr7/reference-manual.pdf.
- Paik, J., Kim, J. W., Ritter, F. E., & Reitter, D. (2015). Predicting user performance and learning in human-computer interaction with the Herbal compiler. *ACM Transactions on Computer-Human Interaction*, 22(5), 25.
- Ritter, F. E., Baxter, G. D., Jones, G., & Young, R. M. (2001). User interface evaluation: How cognitive models can help. In J. Carroll (Ed.), *Human-Computer Interaction in the New Millenium* (pp. 125-147). Reading, MA: Addison-Wesley.
- Tehranchi, F., & Ritter, F. E. (2017). An eyes and hands model for cognitive architectures to interact with user interfaces. In *MAICS, The 28th Modern Artificial Intelligence and Cognitive Science Conference*, 15-20. Fort Wayne, IN: Purdue University.
- Tehranchi, F., & Ritter, F. E. (2018). Modeling visual search in interactive graphic interfaces: Adding visual pattern matching algorithms to ACT-R. In *Proceedings of ICCM - 2018-16th International Conference on Cognitive Modeling*, 162-167. University of Wisconsin, Madison, WI.