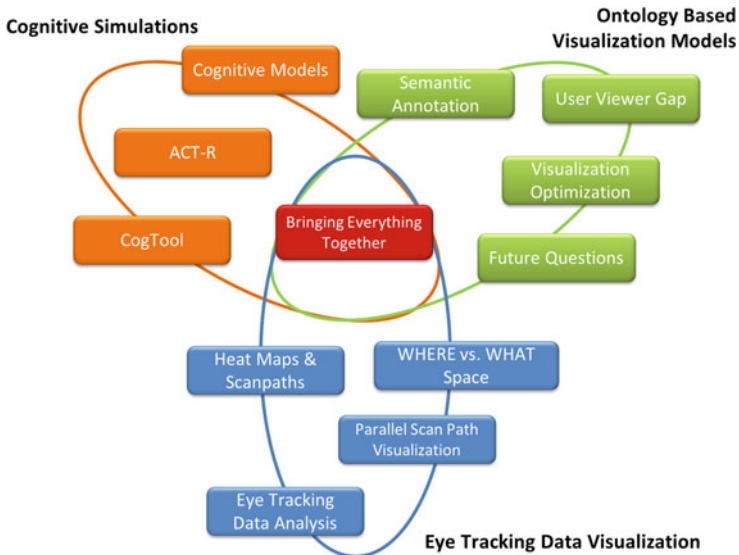


# Cognitive Ergonomics in Visualization

Michael Raschke<sup>(✉)</sup>, Tanja Blascheck, and Thomas Ertl

Institute for Visualization and Interactive Systems, University of Stuttgart,  
Universitätsstrasse 38, 70569 Stuttgart, Germany  
michael.raschke@vis.uni-stuttgart.de  
<http://www.vis.uni-stuttgart.de>

A key factor to assure success of a visualization technique is how efficiently users perceive information using this visualization technique. This efficiency is strongly correlated with visualization parameters which we will summarize in the following with the term *cognitive ergonomics*. In this chapter we will motivate the reader to study cognitive ergonomics in visualization using an interdisciplinary approach. This interdisciplinary approach is based on *eye tracking data visualization*, *ontology based visualization models*, and *cognitive simulations*. Figure 1 shows the chapter structure, and the interdisciplinary approach we suggest.



**Fig. 1.** This chapter presents an interdisciplinary approach to study cognitive ergonomics in visualization based on eye tracking data visualization, ontology-based visualization models, and cognitive simulations.

Many user experiments have been conducted to study perception of visualization techniques. Apart from measuring accuracy rates and completion times, eye

tracking experiments provide an additional technique to evaluate visualizations. Due to eye tracking devices becoming cheaper, eye tracking is a promising approach to study visualization parameters that are relevant for cognitive ergonomics. The results of eye tracking experiments allow researchers to investigate scan paths of eye fixations on the stimulus. Thereby, researchers can measure which areas on the stimulus have been focused on in which order, by which participants, and with how many fixations. Overall, eye tracking metrics allow evaluating cognitive stress by statistically analyzing fixation durations, distributions, and sequences, or cognitive workload by studying changes of pupil size of participants.

Besides using statistical algorithms to compare eye tracking metrics, visualization techniques allow to visually analyze fixation durations, distributions and sequences of several participants at one glance. However, only state of the art visualization techniques are usually used, such as scan path or heat map visualizations (cf. Fig. 2). For this reason, we are motivating readers to develop further visualization techniques for eye tracking data analysis in the first part of this chapter. We will analyze the structure of eye tracking data, and the visual analysis process of eye tracking results from an information visualization pipeline perspective. This systematic approach will help visualization developers to find new visualization techniques for graphical representations of eye tracking data. We will conclude this part with the presentation of the parallel scan path visualization technique which we have developed following this systematic approach.

Today, the development of visualizations is mostly driven by a technical perspective. The main goal of visualization research is to visualize as many data points in real time on high resolution screens as possible. However, there are now tendencies towards a user-centered design of visualizations. The user-centered design takes effects into account such as perception of graphical representations and cognitive workload. The second part of this chapter demonstrates how this user-centered design paradigm can be applied in a visualization scenario. In this scenario we annotate visualizations with semantic information to allow viewers to customize their visualizations.

The discussion of cognitive ergonomics is strongly related to the question if metrics such as fixation durations, distributions, and sequences can be modeled and then be simulated? If this question can be answered positively, the high effort which is required for the preparation, and conduction of eye tracking experiments could be reduced. In the future, interesting visual tasks and visualization parameters could be selected in advance by running a simulated experiment without conducting any experiments with real participants. This approach is mainly inspired by the successful application of cognitive simulations in human computer interaction (HCI) research. The third part of this chapter will give a brief motivation for using cognition simulations to test visualization designs. Therefore, we will discuss interesting aspects of CogTool from HCI, and the cognitive simulation framework ACT-R.

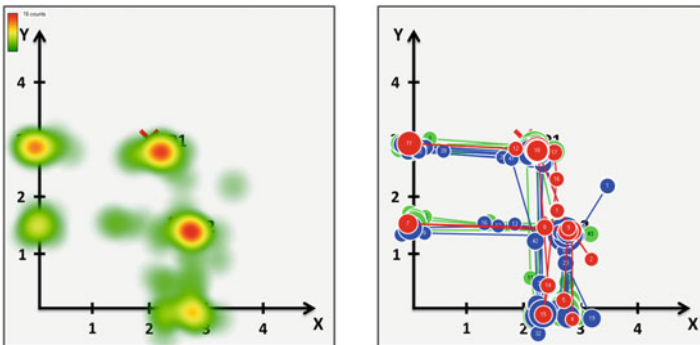
Finally, we will conclude the chapter by bringing together approaches, concepts, and techniques presented in this chapter to formulate a road map to study cognitive ergonomics of visualizations in future work.

# 1 Eye Tracking Data Visualization

To study the readability, efficiency, and cognitive workload of visualizations, controlled experiments, usability tests, longitudinal studies, heuristic evaluations, or cognitive walkthroughs can be performed [1]. Standard metrics to evaluate visualizations are accuracy rates and completion times. Since the recording of eye movements became easier during the last decade, many user study designers additionally use eye tracking techniques. Eye tracking data provides information about eye movements of a participant during a user experiment. In most cases the participants' fixation positions on the screen, the fixation durations, and the sequence of fixations on the stimulus (in the following scan path) is of interest.

Usually, the goal of eye tracking experiments is to find common eye movement patterns. One approach is to use visualizations which show the eye movements of several participants in an appropriate way to support finding common structures. Classic techniques to do this are heat maps and traditional scan path visualizations (cf. Fig. 2) [2]. Enhancements to these classic techniques have been presented by Aula et al. who developed a non-overlapping scan path visualization technique [3]. Another technique is used by eSeeTrack which combines a time line and a tree-structured visual representation to extract patterns of sequential gaze orderings. Displaying these patterns does not depend on the number of fixations on a scene [4]. If areas of interest are available, transition matrices [5,6] or string editing algorithms can be used [7–9]. A relatively new approach is presented by Andrienko et al. [10]. In their work, the authors discuss the application of visual analytics techniques for the analysis of recorded eye movement data. As a follow-up work, Burch et al. demonstrate how visual analytics techniques can be used to analyze an eye tracking experiment [11].

One example of an eye tracking study in visualization research is the comparison of different types of graph layouts such as radial, orthogonal, and traditional by Burch et al. [12]. Another example is the eye tracking experiment by Huang



**Fig. 2.** Most prominent visualization techniques for eye tracking data: Heat maps are time aggregated density based representations (left), scan paths are line based visualizations.

et al. Their results show that graphs are read following a geodesic-path tendency. As a result, links which go towards the target node are more likely to be searched first [13]. Another experiment by Kim et al. investigates the influence of peripheral vision during the perception of visualizations [14].

However, fundamental questions of using eye tracking to test the cognitive ergonomics of visualizations remain. The most important issue about using eye tracking techniques to measure cognitive ergonomics is the question, if the recorded eye movements reflect mental processes, what often is called the “Eye-Mind Hypothesis”? We think, that this question cannot be answered with a definite “yes” or “no”, and refer to the literature [15,16]. Our opinion is, that the answer depends on the complexity of the visualization, the visual task and the required mental processes. Another relevant point for visual analysis of eye tracking data is, that scan paths can have completely different shapes for different participants performing the same task. The question of how these different eye movement patterns could be compared with each other is still not sufficiently answered. However, we think that new visualization techniques for eye tracking data can bring benefit to scan path comparison.

In the following sub chapters, we will analyze the structure of eye tracking data from the visualization pipeline perspective to motivate the reader to develop further visualization techniques. In a second sub chapter we will present the parallel scan path visualization technique as a result of this analysis.

## 1.1 New Visualization Techniques for Eye Tracking Data

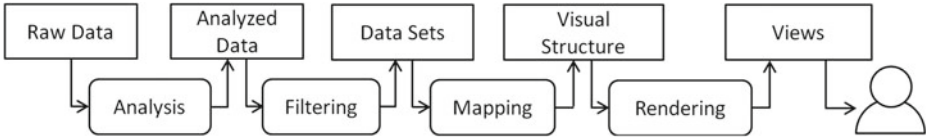
The visualization pipeline defines four steps for deriving a graphical representation from raw data. In the original work of Naber and McNabb these four steps are: data analysis, filtering, mapping, and rendering [17]. In the following, we will formulate a concept for developing new visualization techniques for visual eye tracking data analysis.

### Step 1: Analysis of Eye Tracking Data

Current eye tracking software systems generate large amounts of data representing the output of eye tracker sensors. The most important types of the data sets are: *various types of timestamps, fixation point information for left/right/both eye(s), pupil size for left and right eye, software meta data.*

### Step 2: Filtering

The raw data is filtered depending on research questions which will be answered using the new visualization technique. Usually, usability researchers are interested in: *timestamp of a fixation, fixation coordinates on the screen, and validity of this fixation.* Handling a large number of fixations can be impractical. To alleviate this effect, *areas of interest (AOI)* can be defined to group fixations (Fig. 3).



**Fig. 3.** The visualization pipeline defines four steps for deriving a graphical representation from raw data. We applied the model of the visualization pipeline on eye tracking data to systematically develop our parallel scan path visualization technique.

### Step 3: Mapping

The data selection from step two has to be mapped to geometrical shapes. At first, we have to choose a visualization concept. Even though it seems to be too simple, we can visualize eye tracking data with one dimension using a number line. In this visualization scenario one number line could represent the temporal characteristics of this data. If we want to visualize more information about participants' eye movements on a screen, two dimensional diagrams can be used. For example, heat map and scan path visualizations use Cartesian coordinate systems to display the positions of the eye movement elements. We can get inspirations from visualization collections like <http://www.visualcomplexity.com> or <http://infosthetics.com/> to find an adequate visualization technique.

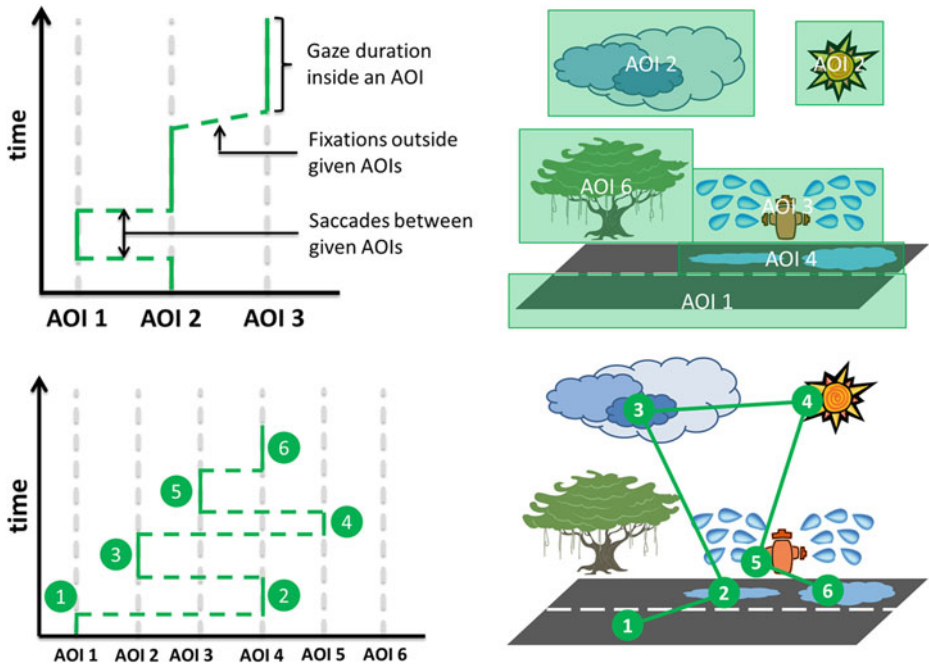
Besides geometrical dimensions of visualizations, colors can indicate additional characteristic of the eye tracking data. Scan paths from different participants can be distinguished using a color table, or can be colored differently when intersecting with areas of interest. Alternatively, interesting characteristics of the eye movements, like high eye movement frequencies, can be displayed using color gradients. Other data dimensions can be mapped to different types of symbols.

### Step 4: Rendering

Finally, the filtered and mapped data is rendered to the screen. Thereby, existing rendering libraries for information visualizations can be used.

## 1.2 Parallel Scan Path Visualization (PSP)

As a result of our analysis we have developed the parallel scan path visualization (PSP) technique [18]. This visualization uses areas of interest. It maps gaze durations and fixations to vertical axes. The top left picture of Fig. 4 shows a sketch of the PSP visualization, where three areas of interest are defined and are mapped to three vertical coordinate axes. The leftmost axis indicates time, starting from the bottom of the diagram with the start time of the eye tracking recording. The orientation of the parallel scan path visualization is arbitrary. In the following we use a vertical time axis from bottom (start of the eye tracking recording) to top (end of the eye tracking recording) as introduced in the original work. The horizontal axis displays all selected areas of interest as independent values. Saccades between areas of interest are indicated with dashed lines. Ascending lines indicate fixations outside given areas of interest.



**Fig. 4.** The PSP visualization maps fixation durations inside areas of interest and single fixations to vertical axes (top left). The leftmost vertical axis indicates time. Thereby, areas of interest (top right) are mapped to vertical coordinate axes in the diagram (bottom left). The corresponding traditional scan path visualization is shown at bottom right. Both the PSP visualization and the traditional scan path visualization show an exemplary scan path for the question “Why is the road wet?”.

Key feature of the **parallel scan path visualization (PSP)** is the visualization of eye movements of many participants in a single visualization with a parallel layout containing various levels of detail, such as fixations, gaze durations, eye shift frequencies, and time.

Figure 4 top right shows an example stimulus together with AOIs of an experiment where participants had to answer the question “Why is the road wet?”. Figure 4 bottom left shows one fixation sequence using the parallel scan-path visualization, Fig. 4 bottom right the traditional scan path visualization of the same fixation sequence. A fixation sequence could be to first focus on the road (1), then on the puddle (2), on the cloud (3), on the sun (4), and on the fire hydrant (5). Finally, the attention would move to the puddle again (6). Using the PSP visualization changes of participant’s attention can be studied by following the fixation sequence line in the visualization.

## 2 Ontology Based Visualization Models

Visualizations often don't have a unique meaning. They can be interpreted differently by several viewers due to different start conditions of their interpretation, such as cultural or intellectual differences. Also a different context of use can lead to different interpretations of the information which is represented graphically. To avoid these misunderstandings this chapter describes a method to annotate visualizations and their graphical elements with semantic information<sup>1</sup>. We annotate visualizations on two levels: the *visualization concept level* and the *graphical elements*. Every graphical element represents a piece of graphically encoded information. We propose to link every graphical element with a semantic web resource. This concept will allow viewers to customize their visualizations and thus, to close the user viewer gap discussed by Norman [20].

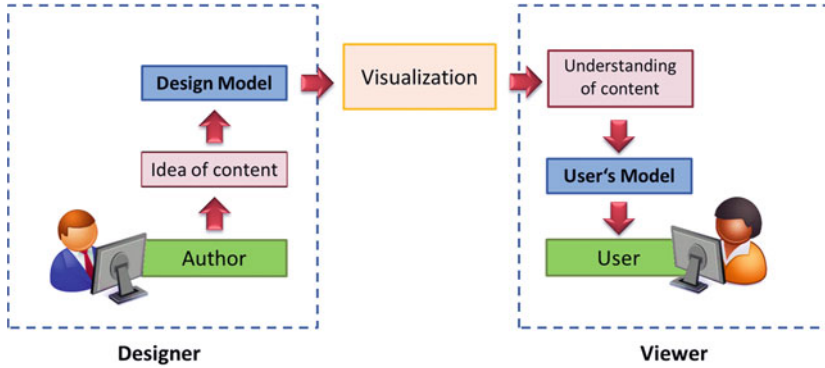
Graphical representations often don't have a unique meaning. This chapter describes a method to avoid these misunderstandings. Therefore, we propose to annotate visualizations on the *visualization concept level* and on the *graphical elements* level. Finally, we will show how this annotation will allow users to individually optimize their visualizations.

Our literature research has shown that one of the most applications of annotated visualizations is to better find graphical elements inside a visualization [21]. Only few approaches deal with the question how semantic annotations can improve the understanding of visualized information. For example, Janeck and Pu show how annotations can be used to find related information about the presented graphics [22]. Other approaches use annotations to allow a semantic filtering [23] or an intelligent zooming [24].

### 2.1 User Viewer Gap

Norman has described how the designer of a visualization transfers information into a graphical form for the viewer (cf. Fig. 5) [20]. The designer creates a *design model* of the visualization. The user conveys a *user model* from the visualization. This user model is based on the interpretation of the graphical elements, their shapes, colors, and spatial relations. In an ideal case the design model is equivalent to the user model. The designer can achieve this equivalence by paying attention to the task, requirements of the visualization, and by adapting the visualization to the user's skills. A user viewer gap emerges from a deviation from the two mental models of the designer and the viewer. This leads to a misunderstanding of the visualized information.

<sup>1</sup> We firstly presented this concept during the workshop "Interaktion und Visualisierung im Daten-Web (IVDW 2011)" (Interaction with and Visualization of Data in the Semantic Web) at the Informatik 2011 conference [19].



**Fig. 5.** Different mental models from the visualization designer, and the visualization viewer can lead to misunderstandings of the visualized information.

## 2.2 Interaction Model for a User Centered Visualization Optimization

We propose to use resources from domain ontologies, and resources from graphical ontologies of the semantic web for the semantic annotation of the visualization concept, and the graphical elements. The designer as well as the viewer both use semantic web resources for the user centered optimization. The annotation with domain ontological information describes the meaning of the graphical elements. References to graphical ontologies define the restrictions and dependencies of the properties of graphical elements, and of the visualization concept. Graphical ontologies allow the designer and the viewer to find alternative graphical elements with the same meaning which can replace an existing graphical element in a visualization. The interaction model is divided into two parts, one for the visualization designer and one for the visualization viewer.

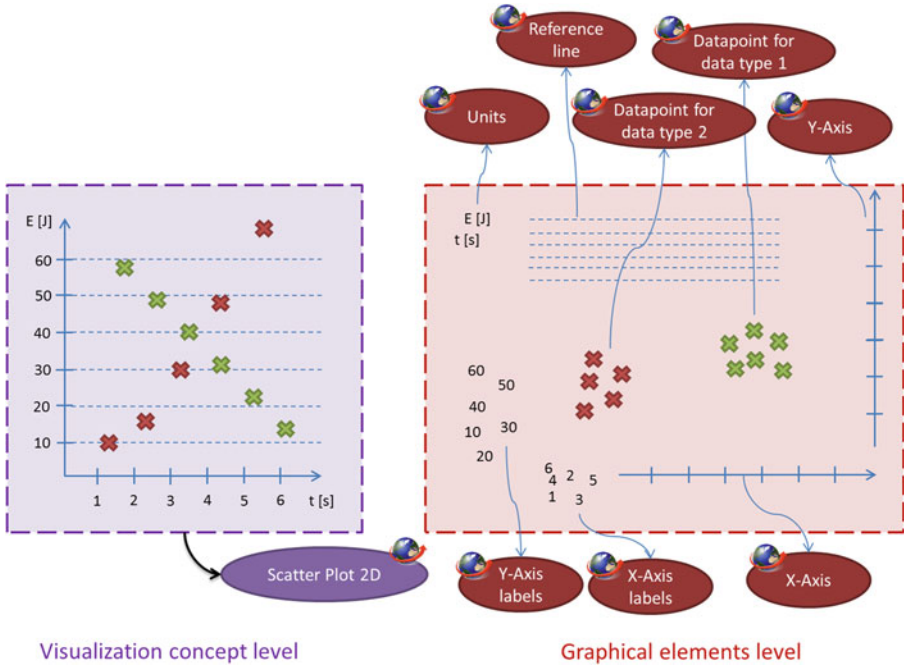
### Visualization Designer

The visualization designer annotates both graphical elements and the visualization concept with resources from domain ontologies and resources from graphic ontologies (cf. Fig. 6 left side). Every graphical element is assigned one or more URIs (Unified Resource Identifier).

### Visualization Viewer

The visualization viewer can explore the visualization using the assigned annotations (cf. Fig. 6 right side). The viewer can replace graphical elements or change the visualization concept. Dependencies and restrictions of the ontologies guarantee a simultaneous persistence of the meaning of the visualization and its graphical elements.





**Fig. 6.** We propose to annotated visualizations on two levels: the *visualization concept level* (left) and the *graphical elements level* (right).

### 2.3 Future Questions

The annotation concept seems to be very simple and useful. However, we identified the following questions for future work during the implementation of our prototype that are crucial for a successful implementation of the concept:

- As described in Sect. 1.1 the visualization process can systematically be divided up into several steps. The last step of every visualization presentation is the rendering step. Thereby, the rendering is always based on parameters such as the geometrical layout, shapes, and colors from the steps before. These parameters are defined by the visualization designer. During the implementation we asked ourself how a renderer which is based on semantic information could look like? What are important input parameters from the semantic models to the renderer algorithms? How can a visualization layout be described in a semantic model?
- We developed a WIMP (Windows, Icons, Menus and Pointer) prototype, where different visualization concepts and graphical elements could be chosen via pull down menus. Due to that, the prototype provided a very simple interaction concept, the question remains how more powerful HCI concepts can be used to improve the user center optimization of visualizations?

- One important drawback of our ontology based concept is, that there does not exist a well defined comprehensive ontology for visualizations and their graphical elements. How could an ontology look like? What are important semantic elements of a visualization? What are their relations? One starting point could be the VISO ontology [25].
- And finally, what are other possible applications of semantic annotated visualizations?

### 3 Cognitive Simulations

Cognitive simulation frameworks provide a promising simulation technique to study formalized cognitive processes during the perception of visualizations. In general, cognitive scientists who are using cognitive simulations aim at using results of psychological experiments to develop models for mental processes which are processes by these simulation frameworks. Cognitive simulations are used to model a wide field of human behavior from problem solving, planning, learning, knowledge representation over natural language processing, perception, expert systems, psychological modeling, to robotics, and human computer interaction. This section motivates using the simulation framework ACT-R in visualization research to model visual search, and the perception of graphics. This section concludes with a brief presentation of results from the successful application of ACT-R in the HCI simulation tool CogTool.

This section presents the basic concept of the cognition simulation framework ACT-R and motivates for using this framework to study aspects of cognitive ergonomics in visualization.

#### 3.1 Brief Introduction to the Adaptive Control of Thought-Rational Simulation Framework (ACT-R)

ACT-R is a modular cognitive architecture, using a production system to operate on symbolic representations of declarative memory<sup>2</sup> [26]. In its core the ACT-R system comes with a visual module for the identification of objects in the visual field, a manual module for hand control, a declarative module for retrieving information from the memory, and an intentional module for the current action goals and intentions. All modules are coordinated through a central production system, which can respond to a limited amount of information in the buffers of the visual, manual, declarative, or goal module. This central production system can recognize patterns in these buffers, and make changes to these buffers. The buffers form one of the fundamental parts of the ACT-R framework, and are noted to cortical regions.

<sup>2</sup> ACT-R is available at <http://act-r.psy.cmu.edu/> together with a large number of articles, conference papers, examples, and the programming documentation.

The ACT-R architecture divides knowledge up into two categories: *declarative knowledge* and *procedural knowledge*. The declarative knowledge represents factual knowledge. For example, declarative knowledge describes what the parts of a bicycle are. Declarative knowledge is processed with so called *chunks*. The procedural knowledge describes actions, for example how parts of a bicycle have to be used in order to drive it. Procedural knowledge is describe by *production rules*. Pattern matching algorithms allow the production system to find appropriate production rules for declarative knowledge chunks in the retrieval buffer considering a given *goal* in the goal buffer.

The framework is mainly written in a LISP dialect, and uses several modules which represent different brain areas. The modules are connected via buffers. Information between the modules is exchanged trough these buffers. ACT-R uses several metrics to measure cognitive activities. These metrics allow the comparison of the simulation results with results from psychological experiments or fMRI images. A comfortable graphical user interface allows to set up all simulation parameters, and to view the simulation results.

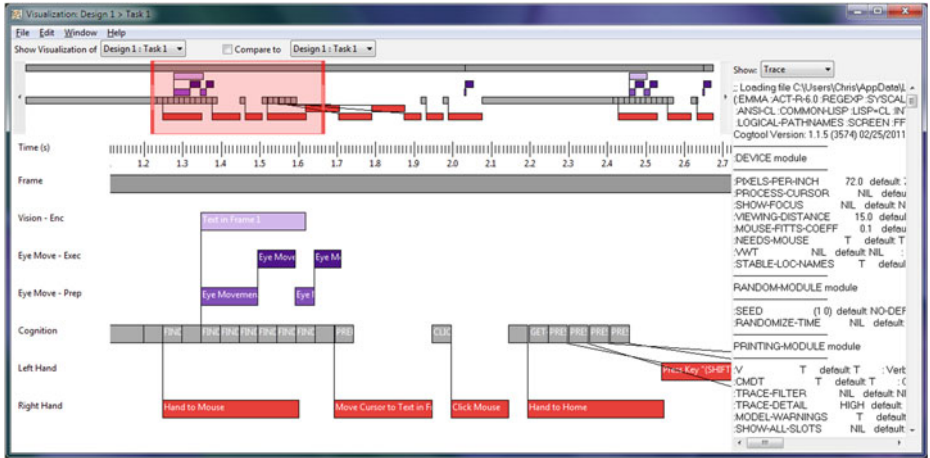
Additional to the built-in visual module of ACT-R Salvucci et al. have developed the “Eye Movements and Movement of Attention” (EMMA) module [27]. This ACT-R module is used to calculate fixation positions during processing a visual search task. EMMA extends the built-in visual module by taking into account effects of fixation frequencies and foveal eccentricity when encoding visual objects. EMMA can predict timings and positions of when and where eyes move, and hence serves to relate high-level cognitive processes with low-level eye movement behavior.

### 3.2 CogTool - Simulation of Human Computer Interaction

Besides the application to model basic intelligent capabilities, cognitive simulation frameworks are used as a basis to model human computer interaction. One example of a tool which models human computer interaction processes is *CogTool*<sup>3</sup> which is based on ACT-R [28]. CogTool provides a framework to design user interface prototypes, and to test their usability. Thereby, CogTool models the execution of prescribed human computer interaction steps, and presents simulation detail results such as timings for vision processes, eye movements, cognition, and manual actions such as hand movements (cf. Fig. 7).

CogTool describes the graphical user interface of applications by *frames* which are views of an application (this could be a dialogue or a complete window). Changes between different frames are called *transitions* which describe interactions leading to a transition. Standard transitions are keystrokes or mouse actions. Therefore, CogTool uses an enhanced keystroke level model (KLM) [29]. Perception and visual search is modeled via EMMA. CogTool provides operators for eye movement preparation, eye movement execution, vision encoding, system

<sup>3</sup> CogTool is an open source project released under LGPL. Executables, documentary, examples, and source code is available at: <http://cogtool.hcii.cs.cmu.edu/>.



**Fig. 7.** CogTool visualizes interaction tasks in a time line diagram. Each row of the time line diagram represents a category of perceptual, cognitive, or motor activity such as vision, eye movement preparation, eye movement execution, cognition, and motor activities of the hands.

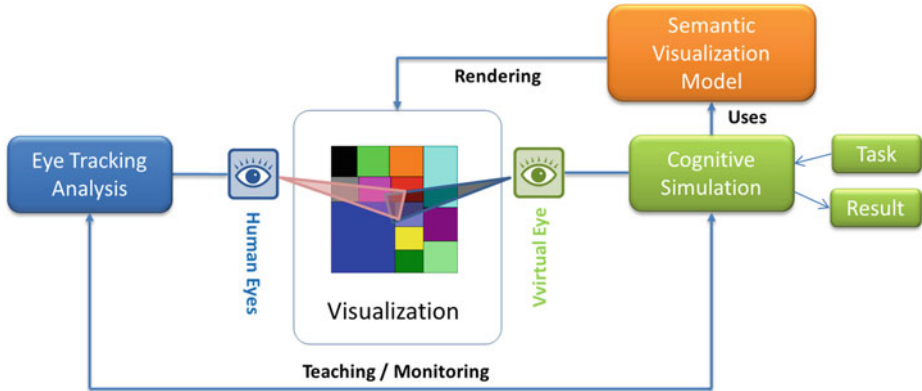
wait, cognition, key presses, cursor moves, mouse clicks, and simple hand movements. These operators are lined up in parallel. For example, the user can move the mouse and can think in parallel. The computation of the duration times of the single operators is done via ACT-R. Thus, duration times are not fix like in the KLM model. They can be different depending on their point in time during the interaction process.

## 4 Bringing Everything Together - Roadmap to Study Cognitive Ergonomics of Visualization

In contrast to HCI, modeling of visual search strategies is not yet a widespread tool for evaluating visualization with respect to their cognitive ergonomics. Analogue to arguments in HCI research by John et al. [28], we believe that the cost of constructing models of visual tasks, even simple ones, is perceived to be too high to justify the advantages of modeling visualization tasks.

We think that by combining results from eye tracking data analysis, semantic models, and cognitive simulation frameworks the cost of constructing cognitive models and running simulations of visualization perception can be reduced.

Once valid models are available the overall effort for conducting user experiments can be reduced by running simulated pre-studies. To reach this goal, we propose to develop a simulation tool similar to CogTool in visualization research. Figure 8 shows a sketch of how the three presented topics can be combined. Eye tracking is used to analyze scan paths from user studies. This analysis is done both with respect to the fixation distribution on the screen (*WHERE space*) and



**Fig. 8.** We propose to use results of eye tracking experiments in the *WHERE* and *WHAT* space to formulate cognitive models for simulating visual search strategies.

with respect to the semantic structure of the scan paths (*WHAT space*). The analysis in both spaces leads to results in two different directions: first, time durations of different visual tasks, and common visual search strategy patterns of participants; second, a model for knowledge processing of visual elements by the temporal order of focused semantic entities, their relations, and meanings. Based on these two results, ACT-R or other cognition simulation frameworks can be used to simulate cognitive and perceptual processes that lead to visual search strategies.

We propose to use results of eye tracking experiments in the *WHERE* and *WHAT* space to formulate cognitive models for simulating visual search strategies. These simulations will lead to a better understanding of visualization parameters leading to optimize cognitive ergonomics of visualizations.

We conclude this chapter with the following remarks and questions formulating a road map to study parameters of cognitive ergonomics in visualization:

1. Most visualization techniques as well as the parallel scan path visualization technique allow to analyze eye movements in the *WHERE* space; they graphically represent participants' scan paths on the screen. We proposed to annotate graphical elements of a visualization with semantic information. Using this annotation the "WHAT" space of scan paths can be studied.
2. To study the *WHAT* space, a sufficient visualization technique of semantic attention, and declarative knowledge processing has to be developed.
3. We proposed to use results from eye tracking experiments in the *WHAT* and *WHERE* space to formulate both a cognitive model of visualization perception, and to simulate visual search strategies. How could such a model look like? What are the development strategies for such as model? Would it be

based on a KLM approach with operators [29] or on a declarative knowledge processing simulation? Would it be possible to combine both approaches?

**Acknowledgements.** We would like to thank our students for their contribution: Tanja Agapkin, Christoph Bergmann, Xuemei Chen, Alex Duscheck, Christian Dittrich, Tobias Eikmeier, Stephan Engelhardt, Dominik Herr, Paul Hummel, Michael Kircher, Andreas Neupert, Hannes Pfannkuch, Sven Plohmer, Nico Ploner, Edwin Püttman, Oliver Schmidtmer, and Stefan Strohmaier (in alphabetical order).

## References

1. Tory, M., Staub-French, S.: Qualitative analysis of visualization: a building design field study. In: Proceedings of the 2008 Workshop on BEyond Time and Errors: Novel evaLUation Methods for Information Visualization, pp. 1–8 (2008)
2. Spakov, O., Miniotas, D.: Visualization of eye gaze data using heat maps. *Electron. Electr. Eng./Elektronika ir Elektrotechnika* **2**(74), 55–58 (2007)
3. Aula, A., Majaranta, P., Räihä, K.-J.: Eye-tracking reveals the personal styles for search result evaluation. In: Costabile, M.F., Paternó, F. (eds.) INTERACT 2005. LNCS, vol. 3585, pp. 1058–61. Springer, Heidelberg (2005)
4. Tsang, H.Y., Tory, M., Swindells, C.: Esetrack-visualizing sequential fixation patterns. *IEEE Trans. Visual Comput. Graph.* **16**(6), 953–62 (2010)
5. Ponsoda, V., Scott, D., Findlay, J.M.: A probability vector and transition matrix analysis of eye movements during visual search. *Acta Psychol.* **88**(2), 167–85 (1995)
6. Goldberg, J.H., Kotval, X.P.: Computer interface evaluation using eye movements: methods and constructs. *Int. J. Ind. Ergon.* **24**(6), 631–45 (1999)
7. Privitera, C.M., Stark, L.W.: Algorithms for defining visual regions-of-interest: comparison with eye fixations. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(9), 970–82 (2000)
8. Duchowski, A.T., Driver, J., Jolaoso, S., Tan, W., Ramey, B.N., Robbins, A.: Scanpath comparison revisited. In: Proceedings of the 2010 Symposium on Eye-Tracking Research and Applications (ETRA), pp. 219–226 (2010)
9. Heminghous, J., Duchowski, A.T.: iComp: a tool for scanpath visualization and comparison. In: Proceedings of the 3rd Symposium on Applied Perception in Graphics and Visualization, pp. 152–152 (2006)
10. Andrienko, G.L., Andrienko, N.V., Burch, M., Weiskopf, D.: Visual analytics methodology for eye movement studies. *IEEE Trans. Vis. Comput. Graph.* **18**(12), 2889–98 (2012)
11. Burch, M., Andrienko, G.L., Andrienko, N.V., Höferlin, M., Raschke, M., Weiskopf, D.: Visual task solution strategies in tree diagrams. In: Proceedings of IEEE PacificVIS 2013, pp. 169–176 (2013)
12. Burch, M., Konevtsova, N., Heinrich, J., Höferlin, M., Weiskopf, D.: Evaluation of traditional, orthogonal, and radial tree diagrams by an eye tracking study. *IEEE Trans. Vis. Comput. Graph.* **17**(12), 2440–8 (2011)
13. Huang, W., Eades, P., Seok-Hee, H.: A graph reading behavior: geodesic-path tendency. In: Proceedings IEEE PacificVIS 2009, pp. 137–144 (2009)
14. Sung-Hee, K., Zhihua, D., Hanjun, X., Upatising, B., Yi, J.S.: Does an eye tracker tell the truth about visualizations?: findings while investigating visualizations for decision making. *IEEE Trans. Vis. Comput. Graph.* **18**(12), 2421–30 (2012)

15. Just, M.A., Carpenter, P.A.: A theory of reading: from eye fixations to comprehension. *Psychol. Rev.* **4**(87), 329 (1980)
16. Anderson, J.R., Bothell, D., Douglass, S.: Eye movements do not reflect retrieval processes. *Psychol. Sci.* **15**(4), 225–31 (2004)
17. Haber, R., McNabb, D.A.: Visualization idioms: a conceptual model for scientific visualization systems. In: Nielson, G.M., Shriver, B., Rosenblum, L. (eds.) *Visualization in Scientific Computing*, pp. 74–93. IEEE Computer Society Press, Los Alamitos (1990)
18. Raschke, M., Xuemei, C., Ertl, T.: Parallel scan-path visualization. In: *Proceedings of the 2012 Symposium on Eye-Tracking Research and Applications (ETRA)*, pp. 165–168 (2012)
19. Raschke, M., Heim, P., Ertl, T.: Interaktive verständnisorientierte Optimierung von semantisch-annotierten Visualisierungen. *INFORMATIK 2011: Informatik schafft Communities*; 41. Jahrestagung der Gesellschaft für Informatik e.V. (GI) (2011)
20. Norman, D.: Some observations on mental models. In: Gentner, D., Stevens, A.L. (eds.) *Mental Models*, pp. 7–14. Lawrence Erlbaum Associates, Hillsdale (1983)
21. Ferecatu, M., Boujemaa, N., Crucianu, M.: Semantic interactive image retrieval combining visual and conceptual content description. *Multimedia Syst.* **13**(5–6), 309–22 (2008)
22. Janeczek, P., Pu, P.: Searching with semantics: an interactive visualization technique for exploring an annotated image collection. In: Meersman, R. (ed.) *OTM-WS 2003. LNCS*, vol. 2889, pp. 185–96. Springer, Heidelberg (2003)
23. Kalogerakis, E., Moutoutzis, N., Christodoulakis, S.: Coupling ontologies with graphics content for knowledge driven visualization. In: *Proceedings of the IEEE Virtual Reality Conference (IEEE VR'06)*, pp. 43–50 (2006)
24. Patel, D., Sture, O., Hauser, H., Giertsen, C., Groller, E.: Knowledge-assisted visualization of seismic data. *Comput. Graph.* **33**(5), 585–96 (2009)
25. <http://www-st.inf.tu-dresden.de/smtvis/blog/>. last Accessed 20 Aug 2013
26. Anderson, J.R., Lebiere, C.: *The Atomic Components Thought*. Lawrence Erlbaum Associates, Hillsdale (1998)
27. Salvucci, D.D.: A model of eye movements and visual attention. In: *Proceedings of the Third International Conference on Cognitive Modeling*, pp. 252–259 (2000)
28. John, B.E., Prevas, K., Salvucci, D.D., Koedinger, K.: Predictive human performance modeling made easy. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 455–462 (2004)
29. Card, S.K., Moran, T.P., Newell, A.: The keystroke-level model for user performance time with interactive systems. *Commun. ACM* **23**, 396–410 (1983)