Project report for: Attention guidance for multi-task displays using human-like cognitive assistants

Szonya Durant, PI, Department of Psychology, Royal Holloway University of London

Kostas Stathis, Co-I, Department of Computer Science, Royal Holloway University of London

Benedict Wilkins, Main post graduate researcher, Department of Computer Science, Royal Holloway University of London

Emanuele Uliana, Post graduate researcher, Department of Computer Science, Royal Holloway University of London

Callum Woods, Post graduate researcher, Department of Psychology, Royal Holloway University of London

Objectives

Our *research hypothesis* is that *human-like assistance in multiple display systems that model the user's joint activity with the display, take into account the user's attention in this context and establish a mutual understanding of the environment*. Our aim is to provide a proof-of-concept prototype that can facilitate future experimental evaluations of these concepts. To build the prototype we will focus on interface functionality, modelling a single user interacting with the MATB-II [1] computer-based task supplied by NASA and designed to evaluate operator performance and workload. The background knowledge is governed by explicit guidelines and constraints.

(O1) establish the informative eye movement patterns that a computer system will use as input for recognizing the cognitive state of the operator in a real-world situation, as well as identify the background joint activity knowledge for the system to use in guiding attention and information to be displayed;

(O2) develop a situation recognition model where cognitive assistants will collaborate to provide guidance locally for individual displays and globally over the context of multiple displays as well;

(O3) deliver a multi-agent system environment that integrates intention and activity recognition in human-like cognitive assistants, eye-tracking techniques and domain specific knowledge representation to provide human-like situation recognition and assistance for the user in guiding their eye movements optimally across the display.

Project Summary

Investigation into past literature on attention and multitasking

Literature on use of the MATB-II suggests that when errors occur in performance it is due to the user not switching in between tasks as required [2]. Typically, humans are only able to solve one task at once and a difficult task can lead to 'tunnel vision' where otherwise salient events are missed.

Past work on using eye tracking to help performance is often treated as a classification problem to identify the state of the user (e.g. low/high workload) often with the aim to decide on level of automation [3].

We ask how can we fully keep the human in the loop so that they are in control of the system but assist them in allocating their attention. Our aim is to monitor spatial attention as a proxy for estimating the content of the user attentional workspace [4, 5].

This information is used by agents responsible for the parts of the interface to deploy guides to attention in order to maximize the user attentional bandwidth. We equip the system with simple tools that guide attention in a bottom-up manner such as peripheral salient highlighting and gaze contingent arrows that point towards where attention is needed at the current location of focus.

Developing a system

We reproduced MATB-II functionality in Python and we refer to this new system as ICU. The reason for developing MATB-II from scratch with ICU was that existing versions were not suitable for our purposes due to not being easily configurable or overly reliant on various Python libraries.

ICU allowed us to do the following.

- *Easily add eye tracking* -- we made use of the PsychoPy [6] library which allows easy calibration and communication with a wide range of eye trackers. Our system was tested with a laptop screen based Tobii- x2-30 sampling at 30-40Hz.
- *Add extra flexibility in scheduling of events* -- the NASA issued MATB-II package requires and xml file as input with all events pre-defined, our system allows events to be configured based on probability distributions over time for easier experimental control.
- *Allow agents to have access to events* -- all events produced by the system can be accessed in real time, in our case for agents to use for decision making and provide guidance.
- *Allow additional overlays to be added to the display* -- our system allows for simple additional overlays in the form of highlighting of areas and arrows.
- *Allow events (in term of extra overlays) to occur in response or gaze contingent manner* -- with the help of agents we are able to deploy the highlights in response eye position in real time.

The combination of ICU with agents monitoring and controlling events form a more sophisticated system that we refer to as ICUa.

ICU system behavior

The ICU simulates MATB-II functionality, without the 'comms' task and scheduling section. It also has the option of collecting and streaming eye tracking data, it also streams all events online and calculates performance error over time. It can be configured more flexibly than MATBII in terms of size, running duration and the ability to allow events to occur probabilistically. When used as part of ICUa it also is able to display additional highlights and arrows. Thus it provides a further open source alternative to the recently developed OpenMATB2 [7].

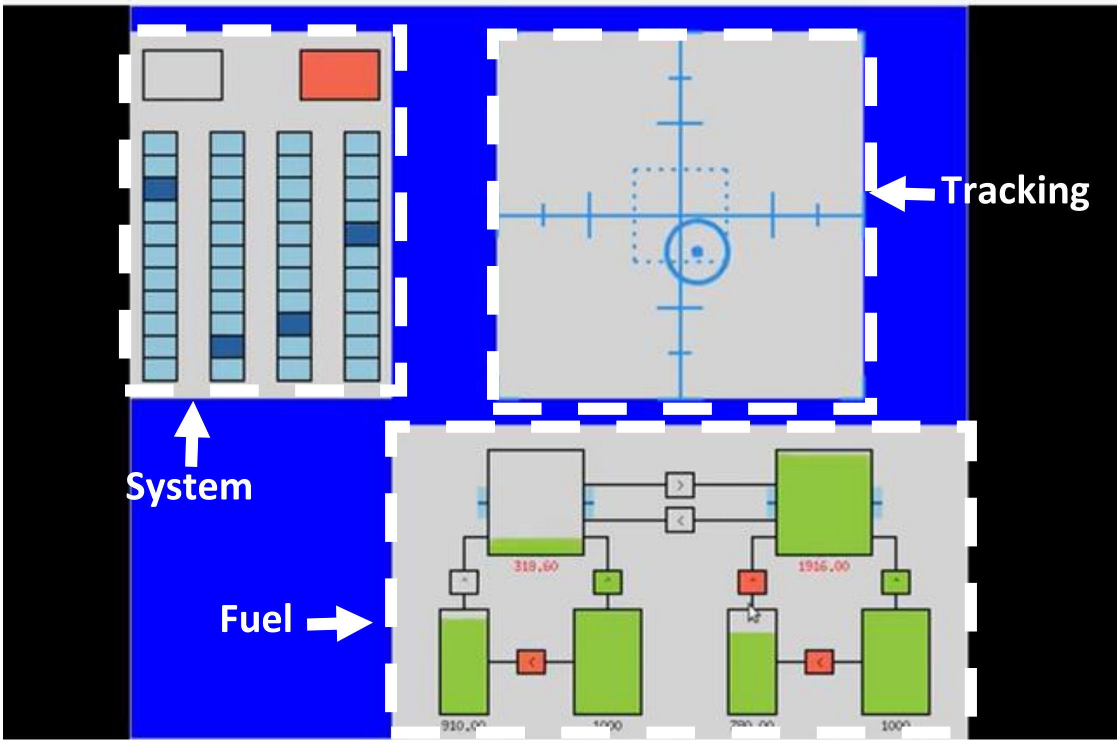

Figure 1. A snapshot of the ICU system with different task components indicated.

ICUa system behavior

In ICUa we Combined the ICU with agents monitoring and controlling events to form ICUa.made use of PyStarWorlds, a Python agent platform that supports the easy implementation of software agents based on the GOLEM framework [8], for controlling the ICU system. Agents perceive events with their sensors attached to component called the “body”, and make decisions with a separate component called the “mind” that communicates with the body to attempt actions with the body’s actuators.

Aims of agents:

- To provide additional alerts to guide attention.
- To not overload attention, by only allowing one additional alert at a time.

Each agent monitors and acts on one subtask in the ICU. They check for undetected errors (that have not been looked at or responded to) and display additional warnings in the form of highlights or gaze-contingent arrows, after a certain time, which are removed after they eyes look and the subtask is responded to. The agents will perform actions specific to their task based on a reasoning (decision) procedure that is specified in a teleo-reactive (TR) style [9].

It is straightforward to add new agents into the system: in addition to attention guidance agents, we have implemented simulated users – agents that play the role of a human and attempt to solve the tasks, and an evaluation agent which maintains a record of the user’s performance according to some predefined metrics.

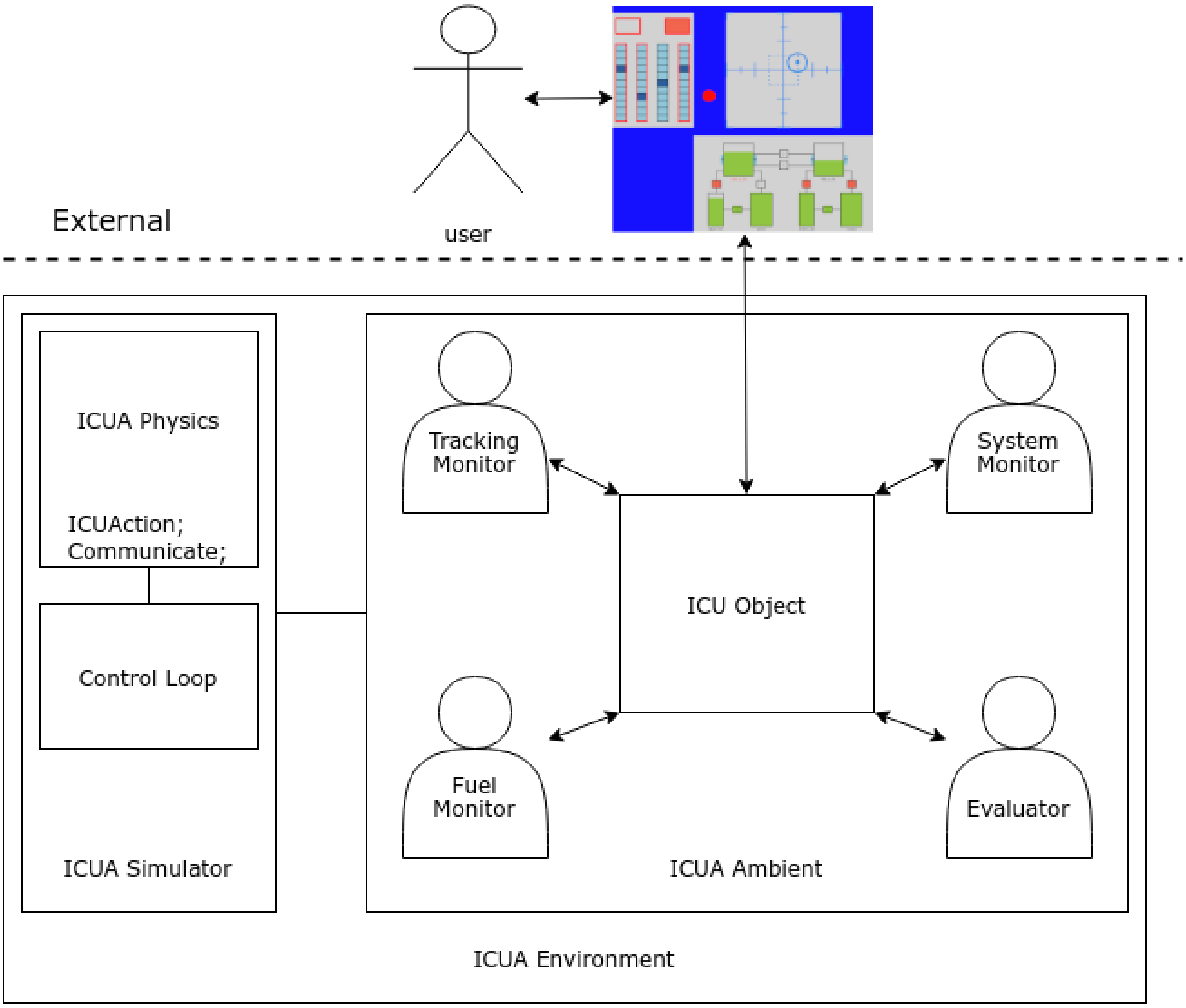

Figure 2. ICUa system architecture overview. The environment is implemented in on top of the PyStarWorlds library. The diagram shows three attention guidance (monitor) agents and an evaluator agent. Each agent interacts with the ICU object via its sensors/actuators. The ICUObject is a bi-directional channel that sends/receives events to/from the ICU system and exposes some additional information about the system (configuration etc).

Testing the system

We tested the capacity of the system and found that the ICUa was able to deal with up to a million events per second, for reference the event load under normal operation doesn't exceed more than one thousand per second with a high-throughput eye tracking device

We created an 'simulated user' agent for reproducing observer reactions and eye movements for comparison with real human observer use of the system. We created toy models of potential user behavior to run experiments on how performance would be affected by the agents.

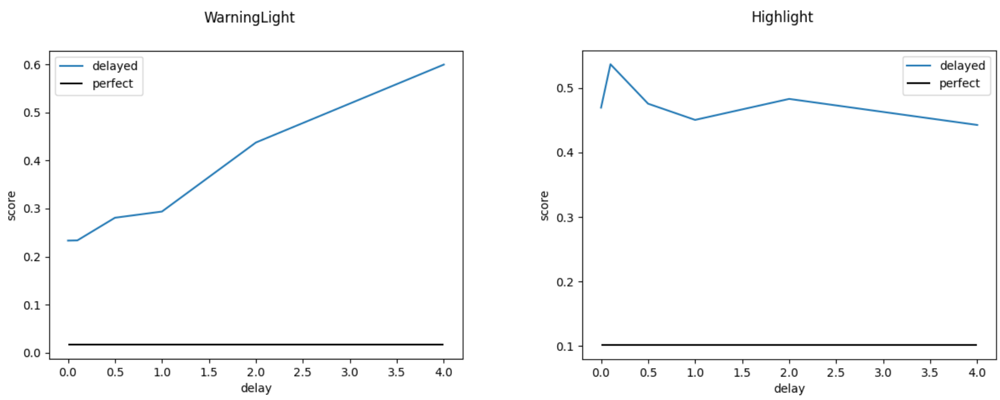

Figure 3. An example of simulated human behaviour. The simulated (delayed) user moves their eyes and responds with some delay in this case to the warning light that requires a mouse click. A delay in clicking causes more error (a), but does not necessarily mean the agents display more warnings. This is because the (delayed) user immediately switches its attention when a highlight is displayed, abandoning its current task. (b) We compare this with a baseline where an all-seeing user responds to all required actions immediately (some error is still present in the perfect user agent due to the simulation speed – 100ms per agent cycle).

Results

The ICU has been realised and distributed via pip, allowing researchers to alter MATB2 to their own needs, more flexibly, incorporating eye tracking and add overlays

ICUa is the extended version of the ICU with agents available for download from GitHub.

Results from the simulated user have confirmed that given some simple assumptions on eye movement behavior and related actions the system functions in terms of guiding attention, i.e. performance improves for simple simulated observers when agents deploy attentional guidance.

We have set up a webpage containing information about the project and links to following downloads.

ICU is available for download from the webpage https://dicelab-rhul.github.io/ICU/

ICUa is available for download from the webpage via GitHub.

In preparation: paper for ETRA ACM conference, deadline 25 Nov.

Future work

The system will be tested on humans to measure the effects on human performance as soon as COVID measures allow.

This will form the basis of an article to be submitted to IEEE Transactions on Human-Machine Systems.

The system can now be extended in several ways:

- More complex agent behaviour, both in terms of aims and guidance, but potentially implementing learning.
- Additional display and attention guidance flexibility (including auditory components for instance).
- Additional human attention tracking measures including for example galvanic skin response of EEG.

Thus these further aspects require a larger team and we are currently discussing with potential collaborators at Royal Holloway and will be reaching out the HLC network.

Funding will be sought potentially in follow-up funding from the HLC network and building towards a large RCUK bid.

References

[1] Santiago-Espada, Y., Myer, R. R., Latorella, K. A., & Comstock Jr, J. R. (2011). The multi-attribute task battery ii (matb-ii) software for human performance and workload research: A user's guide. *NASA TM-2011-217164*

[2] Gutzwiller, R. S., Wickens, C. D., & Clegg, B. A. (2014, September). Workload overload modeling: An experiment with MATB II to inform a computational model of task management. In Proceedings of the Human Factors and Ergonomics Society Annual Meeting (Vol. 58, No. 1, pp. 849-853). Sage CA: Los Angeles, CA: SAGE Publications.

[3] Debie, E., Rojas, R. F., Fidock, J., Barlow, M., Kasmarik, K., Anavatti, S., ... & Abbass, H. A. (2019). Multimodal fusion for objective assessment of cognitive workload: a review. IEEE Transactions on Cybernetics.

[4] Durant, S. (2016). A window to the soul and psyche?. *PSYCHOLOGIST*, *29*(11), 842-845.

[5] Findlay, J. M. *The Mind's Eye* 143–155 (2003).

[6] Peirce, J. W. (2007). PsychoPy—psychophysics software in Python. Journal of neuroscience methods, 162(1-2), 8-13.

[7] Cegarra, J., Valéry, B., Avril, E., Calmettes, C., & Navarro, J. (2020). OpenMATB: A Multi-Attribute Task Battery promoting task customization, software extensibility and experiment replicability. Behavior Research Methods, 1-11.

[8] Bromuri, S., & Stathis, K. (2007, October). Situating cognitive agents in GOLEM. In International Workshop on Engineering Environment-Mediated Multi-Agent Systems (pp. 115-134). Springer, Berlin, Heidelberg.

[9] Nilsson, N., 1993. Teleo-reactive programs for agent control. Journal of artificial intelligence research, 1, pp.139-158