Comparing Visualizations for Tracking Off-Screen Moving Targets

Sean Gustafson and Pourang P. Irani
Department of Computer Science, University of Manitoba
Winnipeg, Manitoba, Canada

Motivation

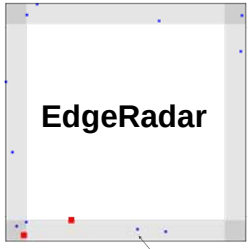
- In games, aircraft navigation systems and in control systems, users track moving targets around workspaces that may extend beyond the users' viewport.
- We compared two different off-screen visualization techniques for accurately tracking off-screen moving targets: Halo [1], the most common off-screen representation and EdgeRadar, a new fisheye-based visualization technique.

- Tracking objects with Halo is difficult because the arcs overlap and become cluttered.
- Halos can bunch up on the edge of the screen and it can be very hard to distinguish between different targets.
- The goal was to develop a new technique to avoid problems with clutter and experiment with new ways of representing off-screen objects.

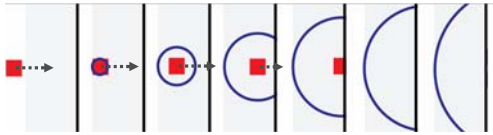
EdgeRadar

New visualization technique for representing off-screen objects, an alternative to Halo that alleviates some of Halo's drawbacks, specifically clutter.

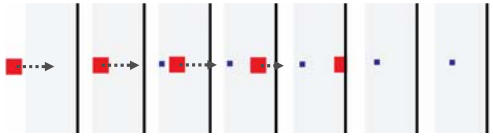
- EdgeRadar is based on the idea of splitting up the overview window (i.e. radar view) found in overview+detail displays.
- The radar view is split up into elongated sections that are shown at the edge of the screen.
- EdgeRadar blips are positioned in the band along the edge such that those closer to the edge of the display are farthest away. This creates a simple fisheye view that provides context in a minimally intrusive way.
- Specifically designed for a moving object tracking task and it will not perform well in tasks that require explicit positioning information.



An object and its EdgeRadar *blip*.



The Halo grows as the object move away from the screen.



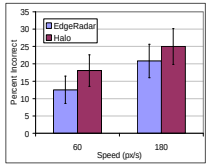
As the object leaves the screen an EdgeRadar *blip* is left behind to show the object's relative position off-screen.

Experiment and Results

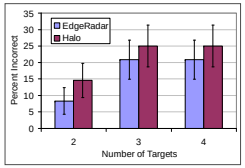
We compared the performance of EdgeRadar to Halo by replicating Pylyshyn's [2] Multiple Object Tracking task but allowing targets to leave the on-screen area. The task was performed like this:

- User is shown a set of target and distractor objects.
- Target objects are highlighted.
- All objects start moving and the user tracks the target objects as they leave the screen.
- After 10 seconds the objects stop moving and one object is highlighted.
- The user is asked if the highlighted object was one of the targets.

EdgeRadar consistently outperformed Halo in this task. However, the results were not statistically significant. A more extensive study is required.



Error rate by target speed.



Error rate by number of targets.

Future Work

This work presents a preliminary but promising step toward the design of visualization techniques for tracking off-screen moving targets. Future work will involve:

- Designing a new technique to address problems with EdgeRadar and Halo. We are in the process of an extensive iterative design process towards this goal.
- Evaluation with a more realistic platform, such as a PDA.
- Comparison with overview+detail and fisheye views.
- Evaluation with tasks other than target tracking. Attempting to mimic real PDA map usage.

Acknowledgments

We gratefully acknowledge the valuable input received from Patrick Baudisch, Microsoft Research and Carl Gutwin, University of Saskatchewan.

The authors can be contacted at sean@gustaf.ca and irani@cs.umanitoba.ca

References

[1] Baudisch, P. and Rosenholtz, R. 2003. Halo: a technique for visualizing off-screen objects. *CHI '03*, 481-488.

[2] Pylyshyn, Z.W. and Storm, R.W. 1988. Tracking multiple independent targets: Evidence for a parallel tracking mechanism. *Spatial Vision*, v3(3), 179-197.