Multimodal Ouija Board

Kojo Acquah (kacquah@csail.mit.edu)
Randall Davis (davis@csail.mit.edu)

Our system illustrates first steps toward a digital Ouija Board and, in doing so, suggests additional functionality available beyond traditional Ouija Boards. The system is designed to provide a familiar interface, using a table-sized digital rendering of an aircraft carrier deck. The system has a basic understanding of deck operations and can aid Deck Handlers by carrying out common tasks. It understands the location of various regions on deck (such as the "the street" or "fantail") and can distinguish different types of aircraft. It has a basic understanding of space and can determine whether a location has room for the aircraft it has been told to move. It can ensure that there is a clear path for aircraft movement, generating a plausible path to the desired destination, then clearing away other aircraft if need be.

The system supports multimodal interaction, i.e., it understands a range of speech and hand gestures from a Deck Handler and responds with its own speech, gestures and graphics. This style of interaction – which we call symmetric multimodal interaction – creates a conversation of sorts between a Deck Handler and the Ouija Board.

Interaction

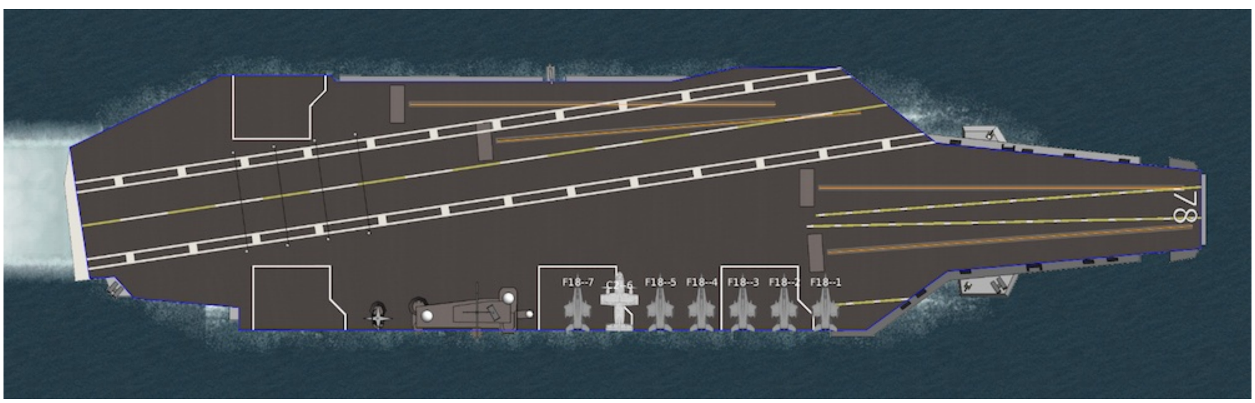
A user stands in front of the Ouija Board display and issues commands with a combination of voice and gesture. Some example commands include:

- "*Move this aircraft to the fantail.*" Spoken while pointing to the selected aircraft.
- "*Move this aircraft over here.*" Spoken while first pointing to the selected aircraft followed by pointing to aircraft destination.
- "*Move aircraft #18 to the fantail.*" Spoken without additional gestures. Aircraft selection is based on tail number (visible on the display).
- "*Move this F-18 to launch on catapult 1*." Spoken while pointing to a specific F-18.

To aid aircraft selection, the system understands the type of aircraft being referred to, and can select a single aircraft from a group or multiple aircraft based on selection criteria and neighboring aircraft.

In situations where the desired actions can be carried out directly, the system does so, providing audio confirmation of its actions. If additional steps are required the system will first indicate them. For instance, if an aircraft is commanded to launch on a catapult but there is no clear path to the catapult or room for launch, the system will detect this and suggest additional actions such as moving other aircraft to accommodate the launch. These additional actions are described aloud and illustrated graphically by the system. The user can accepted these additional actions or issue alternative commands to achieve the desired results.

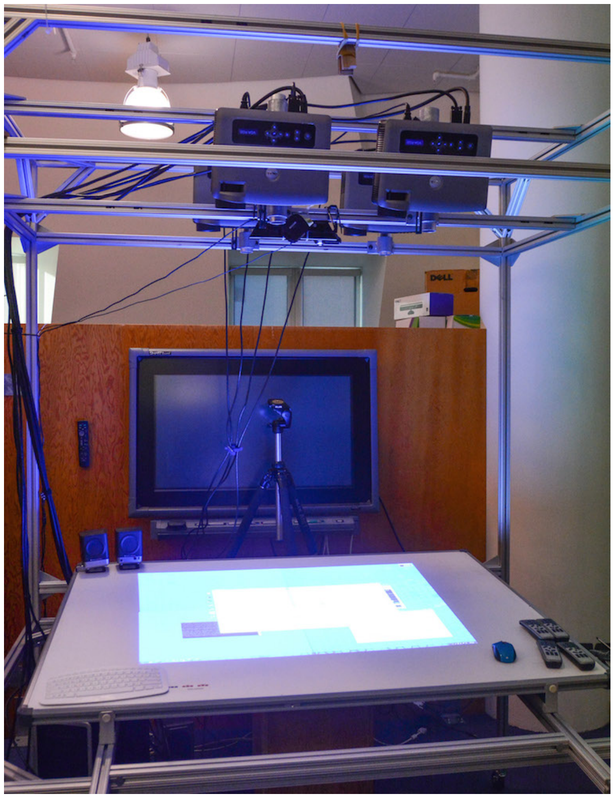
User interacting with table.



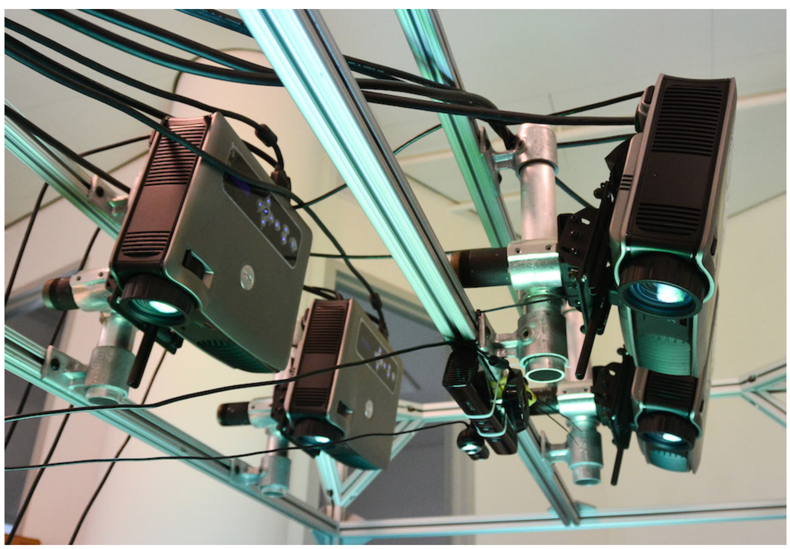
Screenshot at start of deck simulation.

Implementation

The software runs on a Windows desktop computer powering multiple projectors that are coordinated to create a 42 by 32 inch seamless display with a resolution of 2800 x 2100 pixels. A depth sensing camera is mounted among the projectors, looking down on the Ouija Board in order to track the position and orientation of hands. The software is written entirely in Java, which handles all functionality: graphics, speech recognition, speech synthesis, and hand tracking. All processing is done internally without the need for internet communication.



Overview of table setup.



Projectors and depth sensing camera (Kinect V1) mounted overhead.

The system is designed to serve as a platform to be extended for further applications in aiding Deck Handlers through the digital Ouija Board. Much of the underlying functionality has been embodied in modular, self-contained components that interact through well-defined APIs. Concepts such as different aircraft, path routing, deck regions, and movement all exist as objects in the system that can interact with each other. Gesture recognition, speech recognition and synthesis, and the interactive conversation model can be modified or leveraged to build more functionality.

As one example, aircraft movement paths are currently modeled as straight lines from start to end location, but a more intelligent path planner could easily be substituted, e.g., one that examined deck obstacles and built paths around them, resulting in more efficient aircraft operations. This could be seamlessly added to the current system.

Demo

Video can be viewed here.