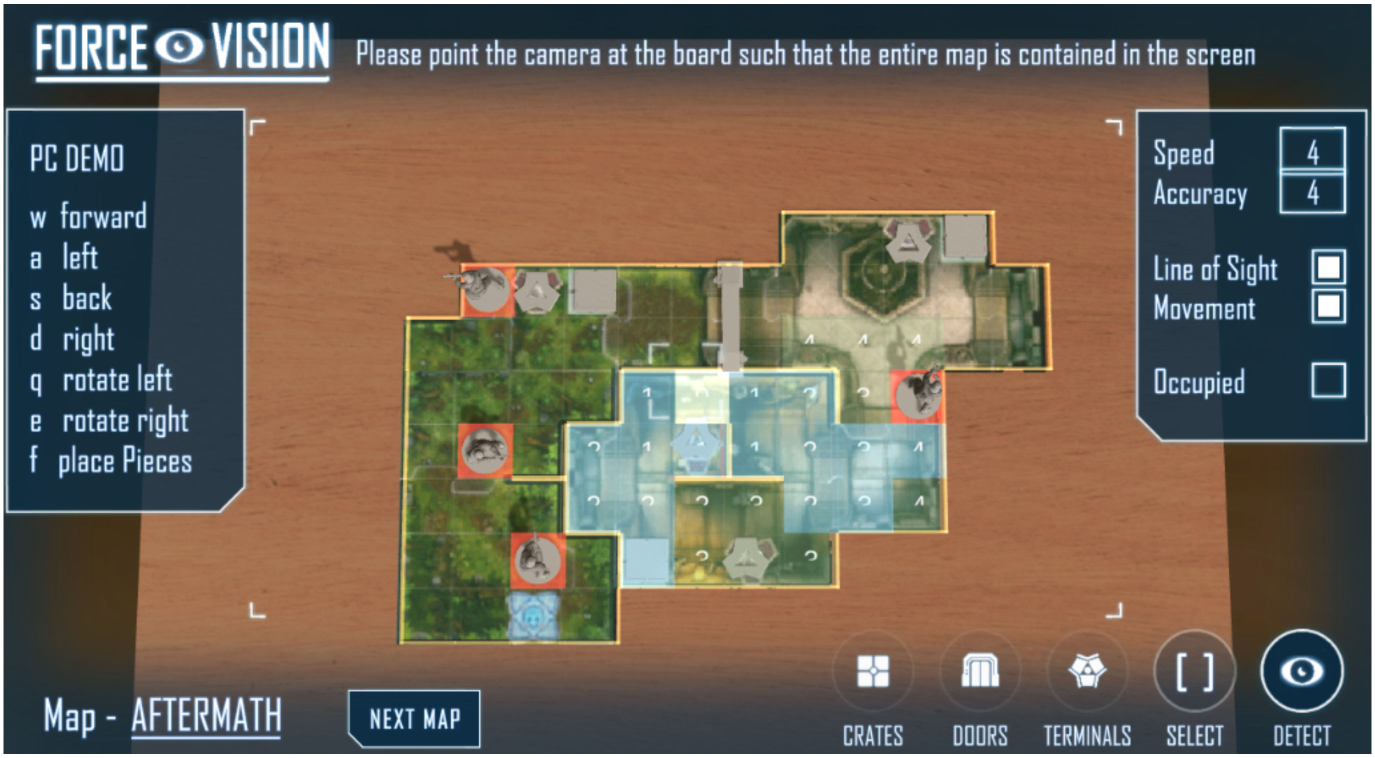
MVP Report

Enhancing Tabletop Miniature Strategy Game Experience with AR and Object Recognition (Computer Vision)

(Force Vision: Imperial Assault AR Companion App)

Manny Kwong 2022



Please point the camera at the board such that the entire map is contained in the screen

PC DEMO

w	forward
a	left
s	back
d	right
q	rotate left
e	rotate right
f	place Pieces

Speed	4
Accuracy	4
Line of Sight	□
Movement	□
Occupied	□

Map - AFTERMATH

1 Features Overview

1.1 Map Features Overlay

The app detects the game board and recognises the map type when the device camera is pointed at the game board. Map features such as boundaries, grid spaces, blocking objects, doors, computer terminals and crates are emphasized by an AR overlay. The visibility of the AR doors, terminals and crates can be toggled on and off. Doors can be opened or closed by tapping on the door, which affects the line of sight and movement calculations. Similarly, terminals can be activated or deactivated and crates can be marked as having been picked up or not. I removed the hints and story text mentioned in the proposal as I decided it was unnecessary for an MVP, which should be more concerned with keeping track of game state and calculating possible actions.

1.2 Select Space Menu

Aiming the pointer (situated in the middle of the screen) to a space on the map will highlight the space. Tapping the select button will select the space and a menu will appear that allows the user to toggle the line of sight and movement displays or mark the selected space as occupied.

This differs from tapping on the space in the proposal. The decision for this change was because the spaces on the mobile phone were too small to accurately tap on when the board is in full view.

1.3 Display the LOS for the Selected Space

When the display LOS toggle is on, the app will calculate and highlight all the spaces on the game board where a hypothetical unit on that selected space has a line of sight for an attack. A line will be drawn from the selected space to any highlighted spaces occupied by a detected piece and the distance (measured in board spaces) will be displayed above the piece. With the space highlighted, drawing a line to an attackable unit as described in the proposal was unnecessary.

1.4 Display the Movement for the Selected Space

When the display movement toggle is on, the app will calculate and highlight all the spaces on the game board where a hypothetical unit on that selected space can move in a single turn. Above these highlighted spaces will show the movement points required to move to the space.

1.5 Set the Selected Space as Occupied

When the occupied toggle is on, the app will highlight the space in red and recalculate the game state as if the space is occupied by a blocking game piece.

1.6 Game Piece (Figurine) Detection

The app will advise the user to point the camera at the board such that its image fits inside the frame to increase the likelihood of correct piece detection. Upon tapping the detect toggle, the app will attempt to detect game pieces and highlight the spaces occupied by the pieces. This will consider the game pieces when calculating movement and line of sight. Pressing the detect button again will reset the highlighted spaces.

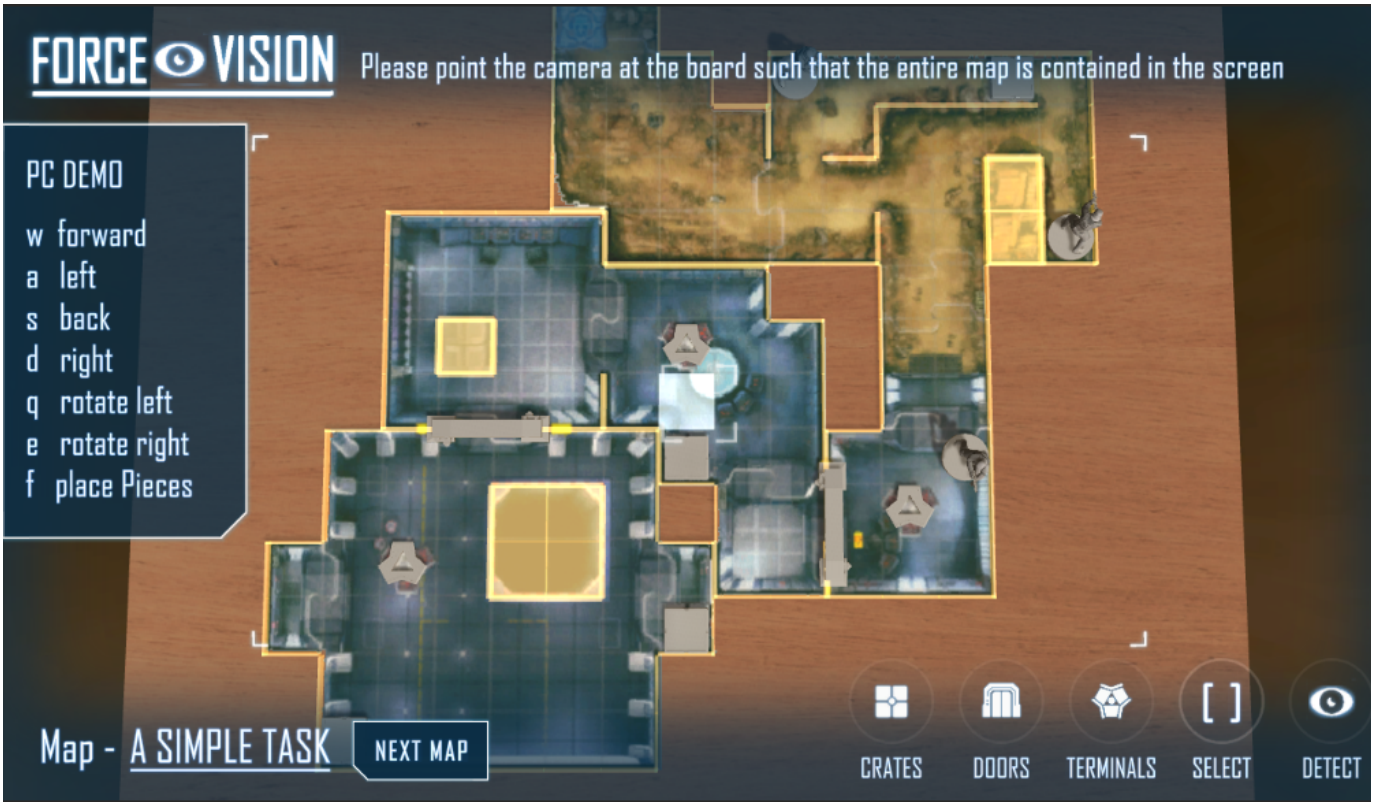
The decision for this change from the proposal was due to the poor performance in detecting every frame on the phone and poor accuracy when the board is not fully visible within the camera view.

1.7 Manual Map Selection

Pivoting to the Vuforia AR Engine for map detection meant that this was unnecessary.

1.8 About Menu

Tapping on the logo will show the about menu. I removed the about button from the proposal to simplify the user interface.



FORCE VISION Please point the camera at the board such that the entire map is contained in the screen

PC DEMO
w forward
a left
s back
d right
q rotate left
e rotate right
f place Pieces

1.9 Test Demo

In the Unity project, there is a Demo scene that lets the user test the functionality on a PC without the physical board game. Replaces the camera feed with a virtual scene that can be translated with WASD keys and rotated with QE keys. The virtual scene consists of the game board on top of a table populated by randomly placed game pieces (figurines). Pressing the F key will randomly reposition the game pieces.

2 Evaluation

The accuracy of the game piece detection is poor. False positives are quite frequent due to the detailed illustrations of the game map. Cascade classifiers were used, which are less accurate than a convolutional neural network (CNN) proposed. I decided to use a cascade classifier instead of CNN due to the difficulty, performance constraints on the mobile phone, and time constraints.

5 Link to Project

Please open the unity project files and play the Demo scene if possible.
mansinh/ForceVision (github.com)