**Prototyping**

The functional prototyping for InteriAR consisted of three main technical questions:

* Is virtual wall colouring feasible to implement and what is the best method of doing so?
* Is “marked tracking” a viable method of us displaying and moving 3D objects in our augmented reality space?
* Will MongoDB be able to handle the volume of users and transactions the app may need in future? ???????

Wall colouring

This prototype was created on Processing 3 using the Ketai for Android library [REF] to access the camera on a mobile device. The software allows a user to tap a pixel on the live video, grabbing the RGB values from it. It then analyses every pixel on the camera feed and calculates whether they are similar enough to the grabbed colour. If they are, the pixel is repainted in red.



InteriAR wall colouring functional prototype v1

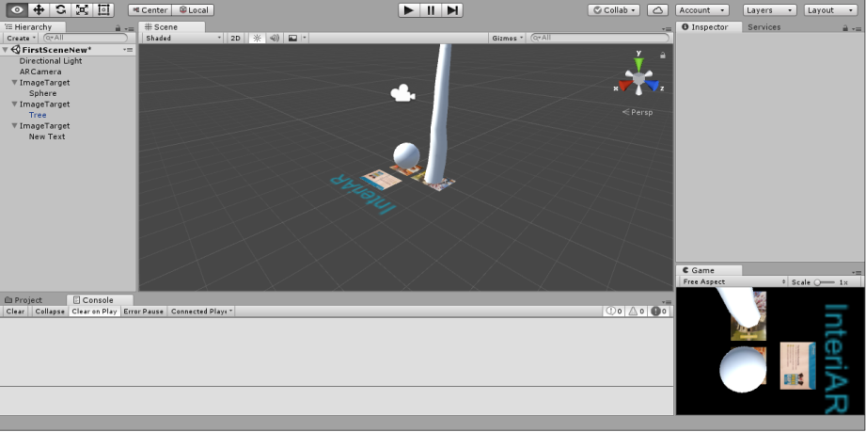
The prototype manages to successfully detect part of the surrounding wall, and avoid the more obviously differently coloured obstacles; however it also misses out large portions and does pick up some unwanted additions. Another factor is performance; when calculating colour distance on each pixel in the feed, especially on a mobile-phone processor, the program starts to stutter.

What we have learned from this prototype:

* It is feasible to implement. Even on a very basic level this functioned in some capacity.
* Further research must be done into colour matching for improved accuracy.
* We need to look into improving the performance drastically, whether via grouping pixels together or relying on another method of detection such as image segmentation.

Augmented Reality Objects

This prototype was created on Unity [REF?] using the Vuforia AR library [REF]. The software utilises a database of markers which are images of real objects or surfaces with enough unique features to be distinguishable from the surrounding area. Computer generated 3D objects are then assigned a marker so that when the camera detects it, the object will superimposed upon it wherever it moves.





InteriAR objects prototype v1 in Unity Marker with “features” highlighted

The close up functionality is impressive, even when dealing with inconsistent lighting. However as soon as the range increases past 2-3 meters it quickly becomes incapable of consistently tracking the markers.

****

InteriAR objects prototype v1 deployed to android

What we have learned from this prototype:

* Marked tracking is only viable at close range.
* While very basic, this software ran seamlessly on mobile.
* It may still be useful for some elements of our project, but we need to research further into marker-less tracking.

Database

To assess the feasibility of using MongoDB [REF] for our project we implemented a cloud-based database using mLab services [REF]. We wanted to test if the cloud-based service was capable of handling high quantities of data being thrown at it and updated via Pymongo [REF] scripts.

To do this, I used python to insert 1,000,000 user documents into a collection and ran find commands to grab out users based on field properties. Both the insertion and any interactive find/update script I ran functioned smoothly and in a timely manner.

What we have learned from this prototype:

* We can efficiently add users to our database using python scripts.
* We can update, remove and pull information from the documents on the database in real-time.