COMPUTER SCIENCE DEPARTMENT

TECHNION Israel Institute of Technology

AR Shadowing

Almog Brand, **Dani Ginsberg** & **Lior Wandel**

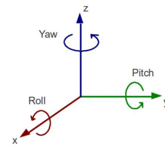
Supervised by: Yaron Honen , Boaz Sternfeld , Boris Van-Sosin

Motivation and Method

- The AR Shadowing project was driven by the need to create more realistic shadows of augmented objects in AR- for a better user experience.
- We created a device that samples the room's light using photoresistors, and then creates a light module of the room.
- The light module is then inserted into the AR world and thus making the whole AR experience more realistic.

HTC vive tracker

- In the AR shadowing prototype, we decided to use HTC Vive's tracker to get the coordinates and rotation of the device anytime we take samples.
- According to the way we designed the device, the Roll and Pitch rotations are fixed and therefore we were only interested in the Yaw rotation relative to Z axis.

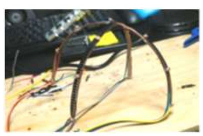


- The tracker position is given as a quaternion [x, y, z, w] - a complex number with 'w' as the real part and x, y, z as imaginary parts. The transition to degrees was made as part of the script running in the HTC Unity environment.

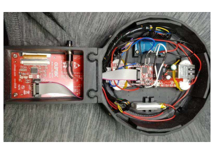
The Hardware

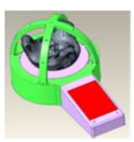
- The heart of the hardware is Olimex esp32-poe development card with a built-in WiFi, Bluetooth, ethernet and SD card slot. Faster and smaller than Arduino - better performances and data transmitting.
  The hardware also includes an LCD touch screen for better user interface.
- In order to decide if the light is coming from the ceiling or the walls, we planed the device with two verticals, half-cycled arcs. Each arc has seven photoresistors on it, positioned in 0°,30°,60°,90°,120°,150°,180° so that the photoresistor on 90° is shared for both arcs.

Hardware 1.0



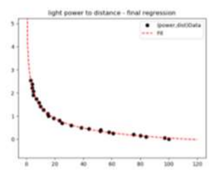
Hardware 2.0



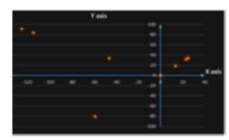


The Fusion Algorithm

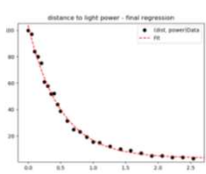
- The Algorithm gets the measurements from the 13 photoresistors and uses them to calculate the location of the light source.
- The measured light intensity is transformed into distance using a non-linear function that was built according to boot measurements we preformed.



- The algorithm takes into consideration the physical location of each photoresistor on the arch and combined with the light distance calculates one sample of the estimated light source position.



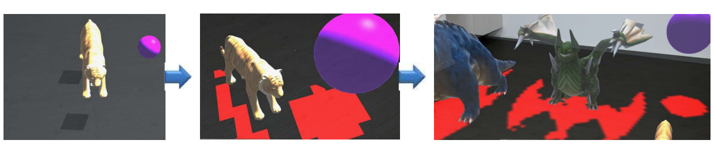
- By taking the average of those 13 samples, we ended up with single location in space represented by (x, y, z) coordinates.
- This location is transformed back to light intensity and the rotate and elevate angles are extracted from the (x, y, z) data.




```
result of quarter  0  is:
Position =  [15.19521975  8.20074894 18.94772743]
elevate angle =  47.6573
rotate angle =  208.3554
light_power =  82.84128186340446
```

Real Time Shadow and the Shadow appearance improvement

- The unity projects its AR scene onto reality by projecting the digital scene related objects onto the lenses of the glasses, that fact means that anything black does not show up on the Hololens, as there isn't a way to project black.
- The whole scene and scene calculations are done on the Hololens. This means that its calculations aren't as strong as ones run on a pc, this leads to poorer performance when calculating real time shadows.



- The poor performance on the real time shadow calculations lead to the shadows not looking as good on the Hololens as they do in the unity editor.
- A lot of tweaking and setting changes were needed in the custom shader in order to get the shadows looking better on the Hololens, closer to how they look in the unity editor.

The Unity Shaders

- We wanted to show the result of the light module creation in the augmented reality, part of it is showing the shadows on the real floor, rather than a digital floor.
- The built-in unity shaders don't show shadows on non-augmented objects.

- Therefore, we needed to create custom shaders to use in our projects.

Android Real Time Implementation

yaronh@cs.technion.ac.il
boazs@cs.technion.ac.il

Powered By