Intro:
During workshop, everyone already downloaded the Vuforia SDK and target database then dragged ARCamera and Image Target to Hierarchy. In the hierarchy, click ARCamerca, go to its inspector and change world view button to “Device Tracking” option in the drop down menu. These instructions will assume everyone did that. At the end, 4 cubes, 1 sphere, a script, spawn point, and plane were made. Here’s steps on

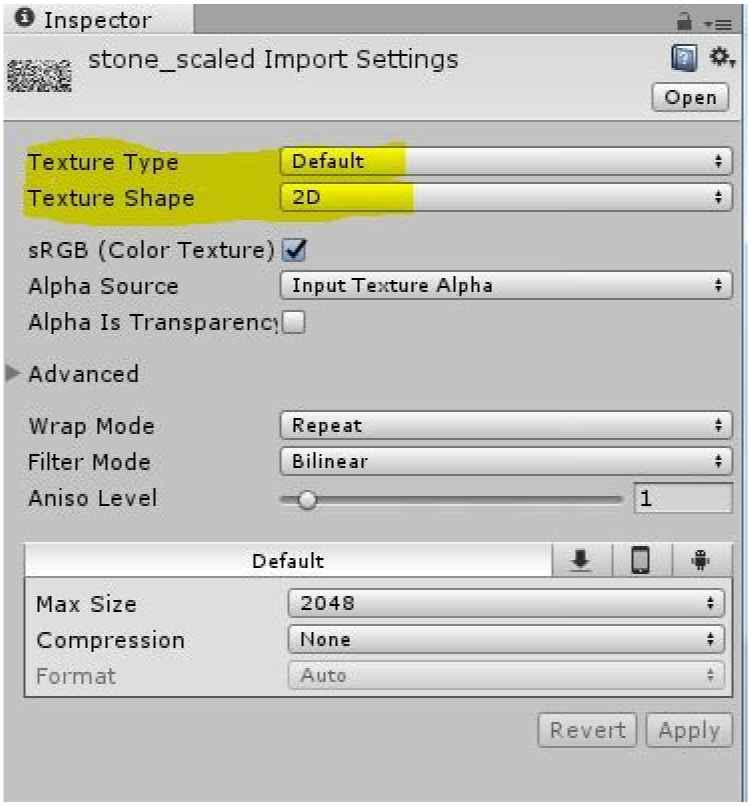
This project can be found on Youtube where Mr. Hallberg explains it step-by-step: https://www.youtube.com/watch?v=uXNjNcqW4kY.

Possible Errors:

White image on Unity

Solution 1: Bottom of Unity in the Project window, Go to folder “Editor” > “QCAR” > “Image Target Texture” > Click (Your Database) Then in Unity inspector on top right above the open button will be a settings button. Click on that and then click “reset”.

Solution 2 (shared during workshop): Go to folder “Editor” > “QCAR” > “Image Target Texture” > click your maze jpg. Go to its Inspector and change texture to 2D.



stone_scaled Import Settings

Texture Type: Default
Texture Shape: 2D
sRGB (Color Texture) ☑
Alpha Source: Input Texture Alpha
Alpha Is Transparency ☐

Advanced

Wrap Mode: Repeat
Filter Mode: Bilinear
Aniso Level: 1

Default

Max Size: 2048
Compression: None
Format: Auto

(image from https://developer.vuforia.com/forum/creating-ar-trackables/target-image-not-displaying-unity)

Ball doesn’t move:

Solution 1: Click ARCamera on Hiearchy > Inspector Tab then change world view to Device tracking

Instructions:

1. Drag ARCamera and Image Target to Hiearchy
2. Create the maze with Game objects cube and repeat until the maze is created. CTRL + C and CTRL + V is the quickest way to create it
3. Create Sphere object
4. Create Spawn point with a cube object and delete everything in inspector except transform
5. Create Plane game object this will act as your floor for the maze
6. Assets (top tab) > import packages > Characters Uncheck everything except for the Physics. Import
7. Search for Ice physics and drag it to all game objects created except for the spawn point
8. Go to sphere and create script by add component (Paste #9 and save) :
9.
```
using UnityEngine;
using System.Collections;

public class ballScript : MonoBehaviour {

	public GameObject plane;

	public GameObject spawnPoint;

	// Use this for initialization
	void Start () {

	}

	// Update is called once per frame
	void Update () {

		if (transform.position.y < plane.transform.position.y - 10) {
```

```
            transform.position = spawnPoint.transform.position;

        }
    }
}
```

10. Drag and drop plane and spawn point object to the scripts empty “Point” and “spawnpoint” in sphere

In theory, this will work. The video has a good visual example on how to create this.

For loading the app on your Android phones, you’ll need to turn on developer mode in the settings of your Android phone and use a cord to tether it, so it will upload when you build the settings on Unity.

For iPhones, make sure to have xcode downloaded to transfer the app to your phone.