# Developing Real World Applications using Augmented Reality

Pranav
*Information Technology Department*
*Maharaja Agrasen Institute of Technology*
*New Delhi, India*

***Abstract***

*As the hardware capabilities of mobile devices advance, Engineering is coming up with various innovations that would have not been possible to achieve a few years ago. Augmented reality is one of these innovations. Although a relatively newer concept, Augmented Reality is already being widely accepted in fields like medicine, architecture, the automobile industry and many more.*

*There is a plethora of applications available for download which are based on Augmented Reality. This paper discusses the different types of Augmented Reality technologies and various components essential to achieve an Augmented Reality environment. A few examples of the applications made using Augmented Reality are also discussed. While it cannot be implied that Augmented reality is something that has become a must in the industry, it is a growing technology which will have a broad spectrum of applications in the future.*

**Keywords** - *Augmented Reality(AR), Virtual Reality, Projectors, Unity, Vuforia*

## I. INTRODUCTION

Augmented Reality can be defined as a phenomenon in which the computer generated 3D imaging is brought into the real world through the camera of the device. The augmented view is made available on the device's display. New graphics [8], sounds and touch feedback is added to our world through Augmented Reality. Various programming languages and development engines can be used to implement augmented reality into the code and made available in the form of an application. Through this code, the positioning and functionality of the augmented object can be brought into the real world. The code requires specification of 3D coordinates in order to incorporate the object. Various coding languages also allow the modification of certain properties like the colour, shadows, texture and depth of the object.

The coordinates of the point where the user touched on the screen can be passed to a defined function in the code and it can provide the augmented object at that position. Multiple objects can be added to the real world environment by storing the coordinates of the user touched points into an array and can be passed to a function that takes in the array as an input and contains the necessary code to bring the object onto the provided points in the real world [1]. Although Augmented Reality seems like an interesting innovation, it requires specific hardware and old devices may not be able to utilize the absolute functionality of Augmented Reality. As it involves heavy processing, the Central Processing Unit as well as the Graphics Processing Unit are used close to their threshold. Running Augmented Reality applications might be an onerous task for some devices but if the hardware used is capable enough, one can enjoy the full capabilities Augmented Reality has to offer.

### *A. Augmented Reality versus Virtual Reality*

Virtual Reality [5] provides a complete virtual environment to the user. It is called "virtual" because it does not involve any interaction with the real world. The surroundings are virtual, and so are the objects in that environment. On the other hand, Augmented Reality brings the objects in the real world environment. The only thing "virtual" in Augmented Reality is the objects themselves i.e. the virtual objects are brought in to the real world environment whereas the former provides an environment that is virtual in its entirety. The user may require a head mounted display (like a Virtual Reality Box) to experience Virtual Reality. The user can experience Augmented Reality on their own physical devices which they use in their daily lives. Both of these technologies have come to light in the recent times and are broadening their usage in various industrial fields.

## II. TYPES OF AUGMENTED REALITY

There are several types of Augmented Reality technologies which have been put to use. Each of the technology [8] differs in its objective and the application for which they're used.

### *A. Projection Based Augmented Reality*

In this type of Augmented Reality, artificial light is emitted onto a surface. This artificial light may take form of objects which are recognizable by the user. The user may interact with the object on the surface by using their fingers and the device forming the object by emitting the light might be able to generate a response based on the user's action. This type of Augmented Reality is being used in the automobile industry to project the directions of the user's destination onto the windshield of the car while driving. It has also been used to make a keyboard that

is emitted in the form of light on a surface and is capable of reading the user's input by recognizing their touch onto the physical surface.

### B. Image Recognition Augmented Reality

This type of Augmented Reality [8] is also referred to as marker based Augmented Reality. In this, a marker (A QR Code, a simple 2D symbol) is placed in the real world which is identified by the camera of a device. This identification point (marker) becomes the point where the augmented object is to be placed. For example, a marker can be placed on a white wall. A set of different colors or different patterns may be used as different objects. Once the device identifies the marker, different objects can be placed on that position (on the position of the marker) in augmented reality. The markers are generally chosen as a simple 2D object so that they are easily detectable by the camera.

### C. GPS Based or Position Based Augmented Reality

This type of Augmented Reality does not require any pre-knowledge of the user's environment. Instead, it uses the GPS (Global Positioning System) of the device, determines the location and implements the augmented reality in real time. For example, if a user walks on a street and points his/her camera towards the street, the augmented reality application will provide the names of the shops and present them on the user's device screen.

## III. COMPONENTS REQUIRED TO ACHIEVE AUGMENTED REALITY

Running Augmented Reality based applications requires extensive power and strong GPUs/CPUs. A lot of mobile companies have started making processors which are capable of running such applications. Some of the components required for an AR application to run properly are discussed below.

### A. Device Sensors and Cameras

Sensors of the device are responsible for performing tasks for a mobile just like the sensory organs of humans. The sensors can detect the rotation and movement of a device. The gyroscope and the accelerometer [9] of a device determine the three dimenisonal coordinates which are required by an AR application. The camera of the device provides the real world environment onto the screen of the phone. The data collected from the sensors is used by the AR application to present the objects in the real world.

### B. Central Processing Unit and the Graphics Processing Unit

The GPU and the CPU are the main processing units of a device. They are responsible for providing the necessary power required for an AR application to run. Better the capabilities of the CPU and the GPU are, more are the possibilities to utilize what the AR applications have to offer. The sensors discussed above, when combined with strong CPUs and GPUs can run applications with a good frame rate and make them appear as realistic as possible.

### C. Projectors

These devices [3] are only used for Projection Based Augmented Reality technique. The function of a projector, as the name suggests is to emit light on a surface which forms the user interactive object in the real world. For example, games like Ludo could be played on a table by projecting the Ludo board and the dice in an Augmented Reality environment.

## IV. AUGMENTED REALITY APPLICATION DEVELOPMENT ENGINES

### A. Unity and Vuforia for Android Devices

Platforms like Unity and Vuforia allow the development of Augmented Reality based applications for both Apple and Android phones. Vuforia is a powerful engine [11] that has been in use for the development of AR based applications. Vuforia allows the user to provide an image of a real world world object and converts it into a "target" file. This target file can be loaded in Unity and the Unity engine is able to build an application that will recognize the target image (processed by Vuforia) as an object and place an AR based object onto the target image in the real world. Both of these engines are most widely used for developing AR based games and other extensive applications.

### B. Apple's Augmented Reality Kit

Apple has recently introduced [10] "AR kit" that has brought in the functionality to write code for AR based applications. The AR kit can be implemented by writing code in Swift (a coding language by Apple) on their compiler/builder known as XCode. XCode and Swift enable the developer to add a lot of functionalities and depths to their AR based application. One limitation to Apple's AR kit is the limited number of devices that support running of AR applications built using the AR kit.

## V. REAL WORLD EXAMPLES OF AUGMENTED REALITY APPLICATIONS

The following applications are made on Swift (Coding Language for Apple Application Development) using XCode as the builder/compiler. Apple's new operating system, IOS 11 brings in support for a lot of Augmented Reality applications.

### A. 3D Dice Application

This application brings in dice into the real world. This application first detects the presence of a plain surface. Once detected, it creates a grid onto the user's screen which enables the user to place single or multiple dice onto the surface. On the user's screen, it appears as if the dice is actually kept on a flat surface.

The user can shake their device and the dice would give an animation of being rolled on the screen.

```
func renderer(_ renderer: SCNSceneRenderer, didAdd node: SCNNode, for anchor: ARAnchor) {
    if anchor is ARPlaneAnchor {

        let planeAnchor = anchor as! ARPlaneAnchor

        let plane = SCNPlane(width: CGFloat(planeAnchor.extent.x), height:
            CGFloat(planeAnchor.extent.z))

        let planeNode = SCNNode()

        planeNode.position = SCNVector3(x: planeAnchor.center.x, y: 0, z:planeAnchor.center.z)

        planeNode.transform = SCNMatrix4MakeRotation(-Float.pi/2, 1, 0, 0)

        let gridMaterial = SCNMaterial()
        gridMaterial.diffuse.contents = UIImage(named: "art.scnassets/grid.png")

        plane.materials = [gridMaterial]

        planeNode.geometry = plane

        node.addChildNode(planeNode)

    } else {
        return
    }
```

**Figure 1 : Code snippet of bringing the grid onto a flat suface**

```
override func touchesBegan(_ touches: Set<UITouch>, with event: UIEvent?) {
    if let touch = touches.first {

        let touchLocation = touch.location(in: sceneView)

        let results = sceneView.hitTest(touchLocation, types: .existingPlaneUsingExtent)

        if let hitResult = results.first {

            let diceScene = SCNScene(named: "art.scnassets/diceCollada.scn")!

            if let diceNode = diceScene.rootNode.childNode(withName: "Dice", recursively: true) {

            diceNode.position = SCNVector3(x: hitResult.worldTransform.columns.3.x,
                y: hitResult.worldTransform.columns.3.y + diceNode.boundingSphere.radius,
                z: hitResult.worldTransform.columns.3.z)

                diceArray.append(diceNode)

            sceneView.scene.rootNode.addChildNode(diceNode)

                roll(dice: diceNode)
```

**Figure 2 : Code Snippet of recognizing user touch and placing the dice**

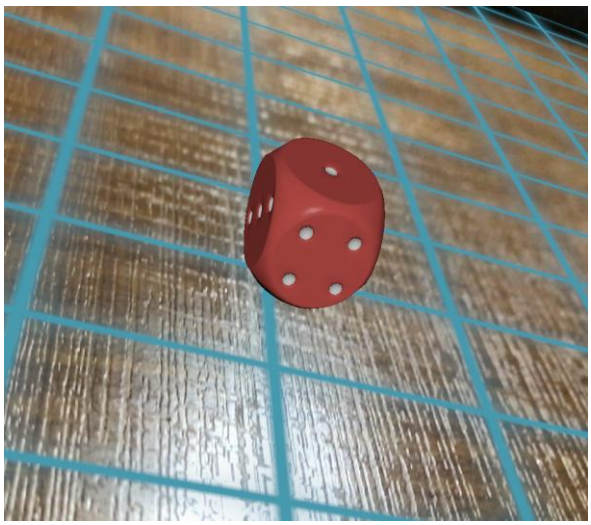

**Figure 3 : Screenshot of the application with dice placed in the real world**

### *B. Performing real world measurements using Augmented Reality Application*

This application enables the user to measure the lengths of objects (like flat surfaces) in his/her environment using Augmented Reality. On the mobile screen, the user can tap on two points in the environment. The application returns the actual distance between those two points on the screen in the form of a floating font. This application does so by taking the x, y and the z coordinates on the plane (that is the environment) and the functions defined in the application contain the necessary code to calculate the distance between them.

```
func addDot(at hitResult: ARHitTestResult) {

    let dotGeometry = SCNSphere(radius: 0.005)
    let material = SCNMaterial()

    material.diffuse.contents = UIColor.red

    dotGeometry.materials = [material]

    let dotNode = SCNNode(geometry: dotGeometry)

    dotNode.position = SCNVector3(hitResult.worldTransform.columns.3.x,
        hitResult.worldTransform.columns.3.y, hitResult.worldTransform.columns.3.z)

    sceneView.scene.rootNode.addChildNode(dotNode)


    dotNodes.append(dotNode)

    if dotNodes.count >= 2 {

        calculate()
    }
```

**Figure 4 : Code snippet to enable the user to add the two points between which the distance is to be measured**

```
func calculate (){
    let start = dotNodes[0]
    let end = dotNodes[1]

    print(start.position)
    print(end.position)

    let a = end.position.x - start.position.x
    let b = end.position.y - start.position.y
    let c = end.position.z - start.position.z

    let distance = sqrt(pow(a, 2) + pow(b, 2) + pow(c, 3))

    updateText(text: "\(distance)", atPosition: end.position)
```

**Figure 5 : Code snippet to calculate the distance between the points specified**

```
func updateText(text: String, atPosition position: SCNVector3) {

    textNode.removeFromParentNode()

    let textGeometry = SCNText(string: text, extrusionDepth: 1.0)


    textGeometry.firstMaterial?.diffuse.contents = UIColor.blue

    textNode = SCNNode(geometry: textGeometry)

    textNode.position = SCNVector3(position.x, position.y + 0.01, position.z)

    textNode.scale = SCNVector3(0.01, 0.01, 0.01)


    sceneView.scene.rootNode.addChildNode(textNode)

}
}
```

**Figure 6 : Code snippet to display the distance in a floating text form**

**Figure 7 : The two points (in red) placed in the real world**


**Figure 8 : The distance between the two points**

## VI. CONCLUSION

Every year, all the leading smartphone manufacturers are coming up with smartphones with advanced technologies and better hardware capabilities. With these capabilities, new innovations in the software industry are coming in for play and one of them that has picked up a spark is Augmented Reality. The developers see a strong future in developing augmented reality applications in terms of market sales as well as satisfactory consumption of these applications by the masses. In 2017, a game called "Pokemon Go", one of the first games based on augmented reality took the world by storm. The use of augmented reality is not restricted to the gaming industry. Archetecture, medicine and many other fields are realizing the potential of augmented reality and how it's affecting the users in a positive way. In years to come, augmented reality may have applications among all the mobile platforms and may become accesible to everyone. Its immersive experience could soon be experienced on all the devices and would not be limited only to the expensive devices.

## ACKNOWLEDGEMENT

I would like to thank Ms. Angela (London App Brewery) whose tutorials tought me the implementation of Augmented Reality in application development (IOS). I would also like to thank my family for motivating me to write this paper.

## REFERENCES

[1] Mekni, Mehdi, and Andre Lemieux. "Augmented reality: Applications, challenges and future trends." Applied Computational Science—Proceedings of the 13th International Conference on Applied Computer and Applied Computational Science (ACACOS '14) Kuala Lumpur, Malaysia. 2014.

[2] Agrawal, Mayur, Adwait Kulkarni, Sneha Joshi, and Nishi Tiku. "Augmented Reality." International Journal of Advance Research in Computer Science and Management Studies Volume 3, Issue 2, February 2015

[3] Grier, R. A., Thiruvengada, H., Ellis, S. R., Havig, P., Hale, K. S., & Hollands, J. G. (2012, September). "Augmented Reality–Implications toward Virtual Reality, Human Perception and Performance". In Proceedings of the Human Factors and Ergonomics Society Annual Meeting (Vol. 56, No. 1, pp. 1351-1355). Sage CA: Los Angeles, CA: SAGE Publications.

[4] Antonioli, M., Blake, C. and Sparks, K., 2014. "Augmented Reality Applications In Education". The Journal of Technology Studies, pp.96-107.

[5] Mehroosh Sidiq, Taha Lanker, and Khalid Makhdoomi. "Augmented Reality Vs Virtual Reality" International Journal of Computer Science and Mobile Computing, Vol.6 Issue.6, June- 2017, pg. 324-327

[6] Joan, D. R. "Enhancing Education Through Mobile Augmented Reality." Journal of Educational Technology 11.4 (2015): 8-14.

[7] Daiwat Amit Vyas, Dvijesh N Bhatt. "Augmented Reality Applications: A Survey on Current Trends, Challenges and Future Scope". International Journal of Advanced Research in Computer Science, Volume 8, No.5 2017

[8] http://www.realitytechnologies.com/augmented-reality

[9] https://www.gsmarena.com/glossary.php3?term=sensors

[10] https://developer.apple.com/arkit/

[11] V.G. Karthiga, Beniel.D, Aravind Kumar.M, Siva Shankar.S. "Augmented Reality Game Development Using Unity and Vuforia". International Journal of Advance Engineering and Research Development, Volume 5, Issue 3 March 2018