An Augmented Reality Application for Home Styling using Unity and Vuforia

Shreeraksha A. 1, Tanya Navas2, Trinita Rebecca Lasrado3, Ujwal Pearl D'Souza4, Lavina D'Silva5

1,2,3,4,5 Computer Science and Engineering, St Joseph Engineering College, Mangaluru, 575028, India

Abstract—Augmented Reality (AR) enhances one’s current perception of reality by augmenting computer-generated or extracted real-world sensory inputs into a real-world environment. Burrow is an AR application that allows a user wishing to furnish their home, to have an interactive Augmented Reality experience to view how different furniture items will appear in their home environment.

Keywords—Augmented Reality, Unity, Vuforia, Home Decor Application, Modern Product Marketing.

I. INTRODUCTION

Interior designing or furnishing a space by a layman involves a lot of assumption and guess work. A piece of furniture that is on display in a showroom might not look the same and blend as desired with the user’s space and environment. This problem of uncertainty in design decision making can be addressed by the applications of Augmented Reality technology.

AR is a live direct or indirect view of a physical, real-world environment whose elements are "augmented" by computer-generated or extracted real-world sensory input such as sound, video, graphics or GPS data.

The Burrow application encapsulates AR technology to serve as an aid for planning and designing the interior for any space in a smart, efficient and aesthetic manner. This application will able to help a person wanting to furnish or decorate a particular area of their homes/offices by providing a clear idea about how an item would look in that spot and how it would integrate with the rest of the room.

AR technology will alter the perception of the world entirely. In terms of consumer adoption, AR is currently being regarded as an extension of existing technologies [1]. This technology can be both accessed and utilized by anyone with a device or smart-phone that has video capturing abilities. AR offers the biggest potential for the commercial market in leading key sectors. In the future, this application can revolutionize how marketing is done, for furniture/home decor retailers by making the customer’s shopping experience more hands-on and interactive.

II. DESIGN

A. Abstract Design

1) Architectural Design: The architecture defines the components, their interfaces and behavior. In Figure 1, Architectural Design for Burrow depicts the relationship between the User who takes advantage of the Augmented Reality furniture application on their Android phones, the Vuforia Database that facilitates Augmented Reality functionality by allowing the Admin to upload markers and objects on it.

Figure 1: Architectural Design

Use Case Diagram: The use case diagram gives the interaction between the actors and the system. Here, there are three actors—User, Admin and Camera. The user is the primary actor, who initiates the interaction by creating a project.
The admin is involved in updating the catalogue and the camera is used in the AR scene as well as, to capture the screenshot.

### Figure 2: Use Case Diagram

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Use Case</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Create Project</td>
<td>Actor: User</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The user can create an AR design project that augments a furniture object into their real world environment</td>
</tr>
<tr>
<td>2</td>
<td>Catalogue</td>
<td>Actor: User, Admin</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The catalogue is a collection of objects along with their description. The user selects an object from this collection to be added into their real world scene. The Admin updates and maintains the catalogue as required.</td>
</tr>
<tr>
<td>3</td>
<td>Position Camera</td>
<td>Actor: User, Camera</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The user has to position the camera in view of the image target so that the camera can detect it.</td>
</tr>
<tr>
<td>4</td>
<td>Build AR Scene</td>
<td>Actor: Camera</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The camera tracks the generated digital object.</td>
</tr>
</tbody>
</table>

### TABLE 1

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Use Case</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Edit Object</td>
<td>Actor: User</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The user can edit the appearance of the object such as colour, position, orientation using the AR scene's UI.</td>
</tr>
<tr>
<td>6</td>
<td>Capture Screenshot</td>
<td>Actor: User, Camera</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The user can capture screenshot of their design if they wish to save it to view later.</td>
</tr>
</tbody>
</table>

### B. Control Flow Design

**Complete System Flow Diagram:** The system flow diagram is a graphical representation of flow of data through an information system modelling its process aspects.

![Complete System Flow Diagram](image)

**III. IMPLEMENTATION**

Implementation is the step where all the planned activities are put into action. The implementation process involves designing, deployment, maintaining and use of final product of the project. The application is developed using Unity and Vuforia.

Unity is a multipurpose game engine that supports 2D and 3D graphics, drag-and-drop functionality and...
scripting using C#\(^2\). As of version 2017.2, Unity integrates the Vuforia Engine, making it even easier to create cutting edge Augmented Reality experiences for both handheld and head worn devices\(^3\).

Vuforia is a cross-platform AR and Mixed Reality (MR) application development platform, with robust tracking and performance on a variety of leading phones, tablets and eyewear. The Vuforia Engine is the client side library that is statically linked to the app\(^4\).

A. Pseudo code

This section includes the pseudocode for all the major modules of the application. They represent C# scripts that are attached to game objects in Unity – as is the design of the game building engine.

1) Pseudo code for Rotate, Translate: This module is responsible for performing rotation transformations of the augmented object. Finger refers to the touch instance created when user touches the component. This class stores touch information. Screen center refers to the position on the screen.

\[\text{Begin}\]
If Component not selected then
Cancel rotation
Return
End if

If rotateComponent = true then
Get Fingers
Get Screen Center of Fingers
Get Twist Degree
Transform Component around Rotate Axis
by Twist Degree
End if

If translateComponent = true then
Get Fingers
Calculate Screen Delta //how far finger has moved since last frame
Set Screen Position to current position of Component
Increment Screen position by Screen Delta
Convert back to world space // update position in real world space

End if

End

2) Pseudo Code for Color Picker: This module is responsible for changing material (color) of the augmented object. The BodyColor array holds all the different materials for the object. The Renderer is the rendering component attached to the object.

\[\text{Begin}\]
Set BodyColor Array
Set index = 0
Get Renderer
If index < BodyColor.Length then
Increment index
else
index = 0
End if
Set Renderer.material = BodyColor[index]
End

3) Pseudo code for Screenshot: This module is responsible for taking screenshot of the Augmented Reality scene and it saves the resulting screenshot in the in-application gallery as well as the gallery of the device on which the application is running.
Begin
Create a render texture.
Apply it on the camera.
Reference the camera in the script
Set the render texture as active
Create new Texture2D = current RenderTexture
RenderTexture.active
Put back the original RenderTexture.active back.
Save the Texture2D to static gallery object
Convert Texture2D to image
Save it in device memory

End

4) Pseudo Code for Gallery: This module will maintain a collection of images that the user will take while designing their AR project. The user may browse through this collection using the UI, view and/or delete the image as they wish.

Saving images to local device memory:

Begin
Deserialize binary file containing images
Convert png to Texture2D format
Assign images to Texture2D gallery collection
End

Retrieving images from local device memory:

Begin
Convert Texture2D collection to png format
Assign png collection to serializable class
Serialize class into binary file on disk
End

Browsing:

Begin
if next button is selected then
increment galleryIndex
else if previous button is selected then
decrement galleryIndex
end if

set UI display image to Gallery[galleryIndex]
End

Sharing:

Begin
Deactivate the UI buttons
Capture screenshot
Place the screenshot in a temporary file
Call AndroidJavaClass API’s- Android Share function
Wait until image is shared
Reactivate the UI buttons
End

IV. CONCLUSION AND FUTURE SCOPE

The Burrow application for home styling using AR will enhance the customer’s home shopping experience. It will give the customer a certainty of how an item will look in their homes before they make the decision to purchase it. This will give them satisfaction of making the right purchase. The interactive application will be easy for the layman customer to use and will give them an intriguing AR experience.
Developing and using AR application has its limitations. First, data cannot be rendered only in the partial scope of vision, it must be rendered fully. The technology should be intuitive with lightweight displays. Connectivity to the internet must also be made common everywhere.

Another significant constraint is the battery life of a device, which must be improved so that the applications can be an easy extension of existing technology. Common mobile devices are still not capable of processing large amount of data and the ones that can, are expensive [5].

In the future, with the normalization of AR support hardware and software in handheld devices, the Burrow application can be made to accommodate the AR experience without the need to possess a physical copy of the image target, to add multiple objects in the same scene etc. This could be achieved through development of ground plane detection technology, advancements in the smartphone’s kinetic sensors and computational power. AR can not only add graphics but also sounds and haptic feedback to the real world [6].

References


