ARRANG(KOLAM APP USING AUGUMENTED REALITY)

A PROJECT REPORT

Submitted by

R.NIVETHIKA 2015115034

R.SOUNDARYA 2015115054

B.SURYA 2015115057

S.AISHWARYA 2015115122

submitted to the Faculty of

INFORMATION SCIENCE AND TECHNOLOGY

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

DEPARTMENT OF INFORMATION SCIENCE AND TECHNOLOGY

COLLEGE OF ENGINEERING, GUINDY

ANNA UNIVERSITY

CHENNAI 600 025

JANUARY 2018

TABLE OF CONTENTS

LIST OF FIGURES

ABSTRACT

Kolam is a good old tradition found in south and south east asian countries, particularly in India. Every morning, millions of women draw kolam on the ground. Kolams are thought to bring prosperity to homes. Kolam is a symbol of south Indian culture. Thats why rangoli competition plays an unique part in every cultural festivals.

The important difficulty while drawing Kolam is that the confusion whether it will fit a plane or not. Till now we dont have any digital assistance for this. We are now introducing our app which when user enters the number of dots to put a Kolam of their favour will detect the given plane(Using AR) and fits the kolam according to the plane and as it is zoomed in or zoomed out the correct projections of line that has to be drawn in that position of the plane so as to complete the kolam will be displayed.

CHAPTER 1

INTRODUCTION

A Kolam is a geometrical line drawing composed of curved loops, drawn around a grid pattern of dots. Kolams are a symbol of auspiciousness. It is Hindu belief that that the geometrical patterns and designs applied with rice flour at the entrance to a home, invites Goddess lakshmi into the household, and drives away the evil spirits. In South India, it is widely practised by female Hindu family members in front of their houses.Kolams are thought to bring prosperity to homes.Through the day, the drawings get walked on, washed out in the rain, or blown around in the wind; new ones are made the next day.

1.1 MOTIVATION

Nowadays people are busy with their day to day life that they forget the concept of kolam. It is now being an extra work to put kolam. We see that we have slowly lost practice as we shift towards easy alternatives like sticker kolams and templates. Many NRIs have lost touch. Some would have interest but would not have proper practice. Some would love to draw but run out of designs. They don't know where to start and would finally end up earning for an assistant. So we thought of making an app which would assist them in the best way possible to put kolam. [**?**]

1.2 MISSION

There we go ! We thought what would it like to be if there will be a digital assistance for them projecting the dots and curves and eventually the

entire pattern helping them out as they complete the kolam of desired size. We thought of developing an iOS app for those who wish to draw kolam but have faded experience in drawing strokes, curves and dots.They can zoom in and manage to archieve intricate stokes and details in the kolam of their choice. ”If you are an art-lover, ”ARRANG” application is for you!” [**?**]

CHAPTER 2

EXISTING WORKS AND RELATED SITES

There are enourmous applicatons exsisting for kolam designs or digital kolam images . They show many different types of kolam designs. But none would help you in real time to draw the kolam in its entirity. There is no proper guidance for the needy. ARRang fulfilles the purpose of guiding the people who is in need of proper guidance to put kolam.

RELATED SITES: LINK 1: This link shows an existing app for different types of kolam designs. This provides the kolam designs based on user preference.
https://play.google.com/store/apps/details?id=com.backdoor.kolamandrangolidesigns

LINK 2: This link provides the basics of augmented reality.
https://code.tutsplus.com/tutorials/beginners-guide-to-augmented-reality –active-4948

LINK 3: This link provides the complete knowledge about the swift language which is used for programming app in MAC based systems.

https://developer.apple.com/library/content/documentation/Swift/Conceptual/Swift$_{P}$*rogramming*$_{L}$*anguage*/*index.html*

CHAPTER 3

LITERATURE SURVEY

Augmented reality gives a view of the real world where elements are superimposed by computer generated files such as graphics, sounds, videos, or digital information. From the first see-through head-mounted AR display developed in the 1960s by Ivan Sutherland at Harvard (Sutherland, 1968), to the enhanced HD4AR and Mobile Augmented Reality System (MARS) developed by Golparvar et al. (Bae et al. 2012), augmented reality technologies have been used in various disciplines and arenas, e.g. engineering, entertainment, aerospace, medicine, military, and automotive industry, as a frontline technology to meet visualization difficulties in their specific domain. AR predates VR, because AR had a real-world use case. One of the key points of difference between AR and VR is that AR needs to be fed a continuously updated view of the world around. Without that, AR can't know what to augment, or where to augment it. First of all, ARKit is Apples own AR SDK, so its perfectly tailored to iOS 11. ARKit uses so-called Visual Inertial Odometry (VIO) for tracking the environment and placing virtual objects with great accuracy and without any calibration. VIO uses several sensors to track where the device is: the camera, accelerometer, and gyroscope. In fact, VIO can be considered a form of the SLAM systems used by some other AR SDKs. pples augmented reality SDK boasts advanced scene analysis capabilities. It can analyze scenes and detect horizontal planes to place virtual objects on and, moreover, ARKit can estimate the amount of light in each scene and adjust the lighting of virtual objects accordingly. The basic goal of an AR system is to enhance the users perception of and interaction with the real world through supplementing the real world with 3D virtual objects that appear to coexist in the same space as the real world. Many recent papers broaden the definition of AR beyond this

vision, but in the spirit of the original survey we define AR systems to share the following properties: 1) Blends real and virtual, in a real environment 2) Real-time interactive 3) Registered in 3D Registration refers to the accurate alignment of real and virtual objects. Without accurate registration, the illusion that the virtual objects exist in the real environment is severely compromised. Registration is a difficult problem and a topic of continuing research.

CHAPTER 4

DESIGN

FLOW DIAGRAM

SELECTION :

The type of the kolam is first selected.

PLANE DETECTION:

We safely unwrap the anchor argument as an ARPlaneAnchor to make sure that we have information about a detected real world flat surface at hand. Here, we create an SCNPlane to visualize the ARPlaneAnchor. A SCNPlane is a rectangular one-sided plane geometry. We take the unwrapped ARPlaneAnchor extents x and z properties and use them to create an SCN Plane. An ARPlaneAnchor extent is the estimated size of the detected plane in the world. We extract the extents x and z for the height and width of our SCNPlane. Then we give the plane a transparent light blue color to simulate a body of water. We initialize a SCNNode with the SCNPlane geometry we

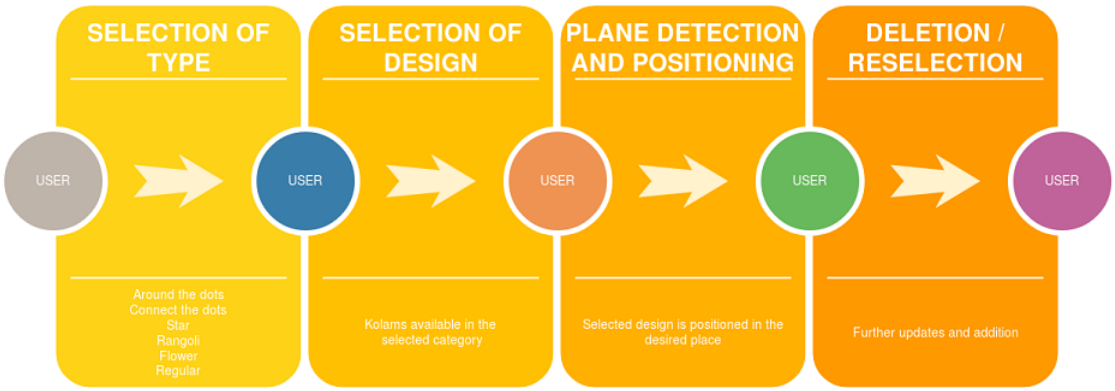

Figure 4.1: FLOW CHART

just created. We initialize x, y, and z constants to represent the planeAnchors center x, y, and z position. This is for our planeNodes position. We rotate the planeNodes x euler angle by 90 degrees in the counter-clockerwise direction, else the planeNode will sit up perpendicular to the table. And if you rotate it clockwise, David Blaine will perform a magic illusion because SceneKit renders the SCNPlane surface using the material from one side by default. Finally, we add the planeNode as the child node onto the newly added SceneKit node.

POSITIONING:

the nodes first child node. Lastly, we safely unwrap the planeNodes geometry as SCNPlane. We are simply extracting the previously implemented ARPlaneAnchor, SCNNode, and SCN plane and updating its properties with the corresponding arguments. Here we update the planes width and height using the planeAnchor extents x and z properties. At last, we update the planeNodes position to the planeAnchors center x, y, and z coordinates. After you get the first object ,its worldTransform columns will provide the real world coordinates of the touch location. Just create a SCNNode of your geometry and set its position to those columns values. Add that node to scene view.

ARCHITECTURE DIAGRAM

Design in blender:

Blender is the free and open source 3D creation suite. It supports the entirety of the 3D pipelinemodeling, rigging, animation, simulation, rendering, compositing and motion tracking, even video editing and game creation.

.blend files:

BLEND files contain all objects, textures, sounds, images, effects,

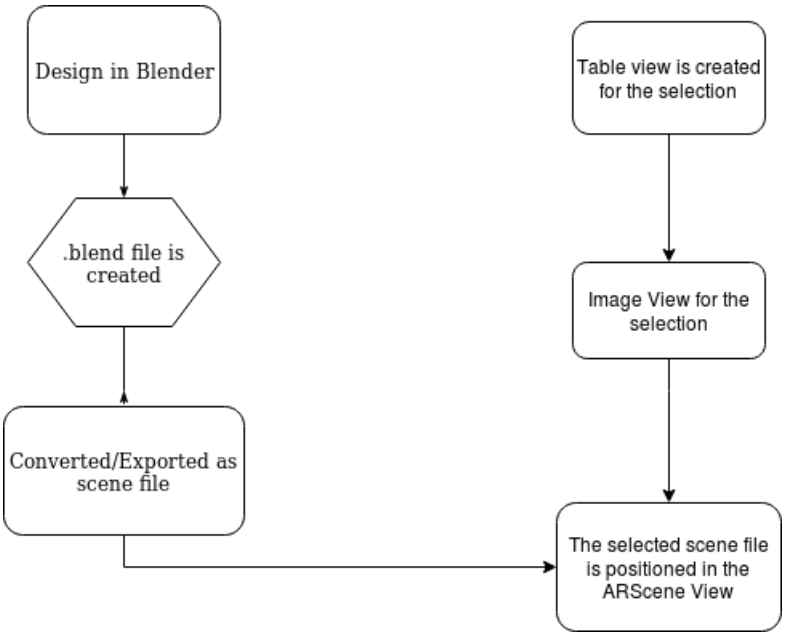

Figure 4.2: ARCHITECTURE DIAGRAM

and scenes used in an animation. Therefore, a BLEND file is a project file rather than a basic 3D image or animation file.

Exporting as scene file:

Typically, you obtain a scene from a file created using external 3D authoring tools. If you include scene files in your app's bundle resources directory, Xcode compresses them for optimal SceneKit loading performance.

Table view is created:

Table views are versatile user interface objects frequently found in iOS apps. A table view presents data in a scrollable list of multiple rows that may be divided into sections.

Image view is created:

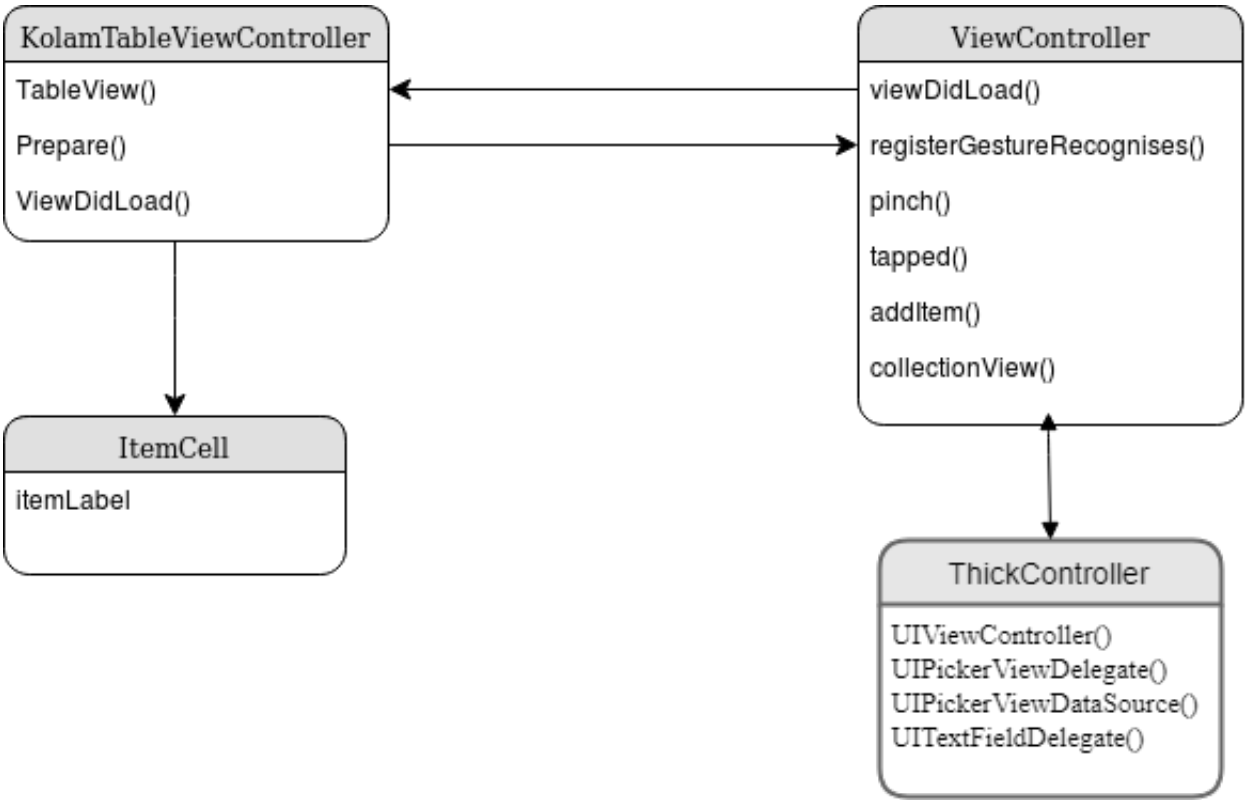

Figure 4.3: CLASS DIAGRAM

The UIImageView class provides the easiest way for developers to implement animations. All you need to do is to create an UIImageView object with a series of images for animating. Once you trigger the animation, the image view will automatically render the individual frames.

ARScene view:

The ARSCNView class provides the easiest way to create augmented reality experiences that blend virtual 3D content with a device camera view of the real world. When you run the view's provided ARSession object: The view automatically renders the live video feed from the device camera as the scene background.

CLASS DIAGRAM

VIEW CONTROLLER

This class carries functions to load,recognize the plane using feature points ,scale , rotate and position the kolam object.

THICK CONTROLLER

This carries the function to adjust the width of the kolam object.

KOLAM TABLE VIEW CONTROLLER

Used to select the type of the kolam object.

ITEM CELL

Displays the selected object.

CHAPTER 5

IMPLEMENTATION

5.1 FUNCTIONS AND ALGORITHM

SWIFT language is used.

Algorithm for plane detection.

Step 1: Start.

Step 2: Configure the session as ARWorldTrackingConfiguration().

Step 3: Add plane detection property to the configuration.

Step 4: Set the value as horizontal.

Step 5: The renderer method is invoked.

Step 6: An anchor of type ARAnchor is passed to the method.

Step 7: Check if the passed anchor is a plane anchor.

Step 8: If it is true go to step 9,else return and go to step 16.

Step 9: Convert the type of anchor from ARAnchor to ARPlaneAnchor.

Step 10: Create SCNPlane with width and height.

Step 11: Create a node to attach this plane using SCNNode().

Step 12: Fix the position of the node.

Step 13: Since this SCNPlane is vertical rotate it by 90 degree clockwise using SCNMatrix4MakeRotation().

Step 14: Add material to the created plane.

Step 15: Attach this plane to the node.

Step 16: End.

Algorithm for adding kolam in the AR scene view

Step 1:Start.

Step 2:First the user is asked to select the thickness of the kolam based on their desire using a drop down list.

Step 3:Then various categories of kolam is displayed as table view and the user is allowed to select the category they desire.

Step 3:Then every kolam which is selected will be displayed according to the thickness selected.

Step 4:After the detection of the plane the position were the kolam is to be placed is chosen by the user.

Step 5:Based on the selected position the kolam is appended to the AR scene.

Step 6:End.

Algorithm for scene deletion

Step 1: Start.

Step 2: Creation of array for SCNNode

Step 3: The selected scenes are appended in the array for every selection.

Step 4: To delete the trash button is selected

Step 5: This inturn calls the trash function

Step 6: All the scenes are deleted from the parent node using removeFromParentNode().

Step 7: End.

Algorithm to Resize the Virtual object

Step 1:Start.

Step 2: pinchGestureRecogniser is used to resize the selected object.
Step 3:The location where resizing has to be done is identified and stored.
Step 4:Now the nodeis redefined as the node which has to resized.
Step 5:Then the action which is to be performed is stored and implemented.(i.e.,Whether it is to be pinched to a larger size or whether it is to be made smaller).
Step 6:End.

Algorithm for Virtual object rotation

Step 1: Start.
Step 2: panGestureRecogniser is used to rotate the image to a different orientation.
Step 3: First it recognizes whether the view is ARscenview or not.
Step 4:Then it identifies the ocation of the position where the touch is made.
Step 5: Then the selected node is applied with the new angles.
Step 6: End.

Algorithm for playing the video

Step 1:Start.
Step 2:The location of the video file is identified.
Step 3:The height and width of the video that is to be played is mentioned.
Step 4:Then the position where the video is to be played is set.
Step 5:The geometry of the plane is set(i.e.,tvPlane)
Step 6:Then the node is appended.
Step 7:End.

CHAPTER 6

CONCLUSION AND FUTURE WORK

There are large numbers of NRIs out there who are having desire to draw kolam but cant because of lack of guidence. Our app will surely help out those people. There is no need to worry about the space because the app itself will take care of it. JUST PROJECT USING THE PHONE AND START DRAWING!!!

6.1 References

1.https://developer.apple.com/documentation/arkit
2.https://developer.apple.com/swift/
3.https://www.youtube.com/watch?v=c8BGQ3CfPBs
4.https://fieldguide.gizmodo.com/everything-you-need-to-know-about-augmented-reality-now
-1809069515
5.http://indianexpress.com/article/technology/gadgets/magic-leap-one-everything-
you-need-to-know-about-augmented-reality-glasses/
6.https://stackoverflow.com/questions/13467029/how-to-create-dropdown-in-xcode
7.https://www.appcoda.com/arkit-horizontal-plane/
8.https://digitalleaves.com/blog/2017/10/augmented-reality-with-arkit-detecting-planes/
arkitlessons.com/how-to-detect-planes-in-arkit-and-spritekit/
9.https://machinethinks.com/arkit-detecting-planes-and-placing-objects/