ISTD 50.017 Graphics & Visualization

Spring Semester 2017

FINAL PROJECT PROPOSAL

By:

Zhexian Zhang | 1001214

William Koh | 1000929

Dhanya Janaki | 1001288

TABLE OF CONTENTS

(Click on each item to navigate to page)

1. INTRODUCTION

Background

Still remember your childhood kaleidoscope toy? A kaleidoscope is able to create mesmerizing images by simply rotating a long tube with reflective surfaces and colored, translucent objects inside. Light waves enter the tube by first passing through colored objects and then are reflected inside the tube to create colorful patterns.

We enjoy kaleidoscope for the special visual pleasure in viewing the random and complex patterns. After all, the word “kaleidoscope” itself come from the term “beautiful form to see” in Greek[1]. However, it is hard to fine-tune the physical kaleidoscope, or even capture and share the patterns generated. Wouldn’t it be exciting if we are able to achieve all this in a digital kaleidoscope, with a similar realistic viewing experience brought by virtual reality?

Aim

We aim to create:

- A basic digital kaleidoscope with two viewing mode: analogue mode and zoomed-in (fullscreen) mode
- Adjustable object count, size, color, and mirror angles
- Special conditions and kaleidoscopic objects to simulate special effects that are achievable in real life
- Virtual reality viewing capability

[1] Kaleidoscope etymology: https://en.wikipedia.org/wiki/Kaleidoscope

2. LITERATURE REVIEW

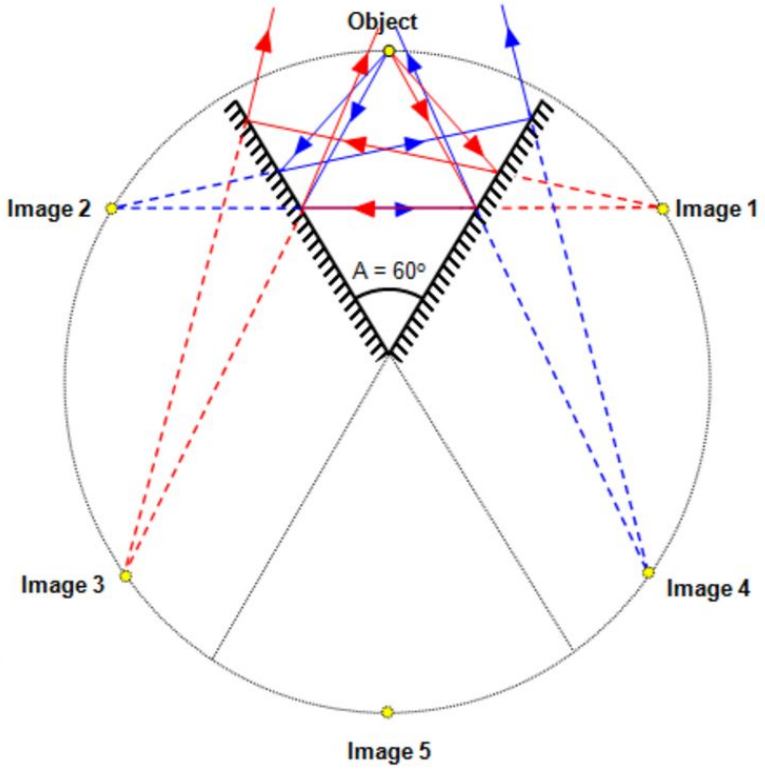
How does Kaleidoscope Work

The kaleidoscope operates using two principles of science: the law of reflection, and the colour composition of light.

The Law of Reflection

The law of reflection states that the angle of incidence equals the angle of reflection. In a kaleidoscope, the light waves reflect back and forth inside the tube allowing the creation of multiple images.

If the image of an object is viewed in two plane mirrors that are inclined to each other more than one image is formed. The number of images depends on the angle between the two mirrors, as shown below[2]:



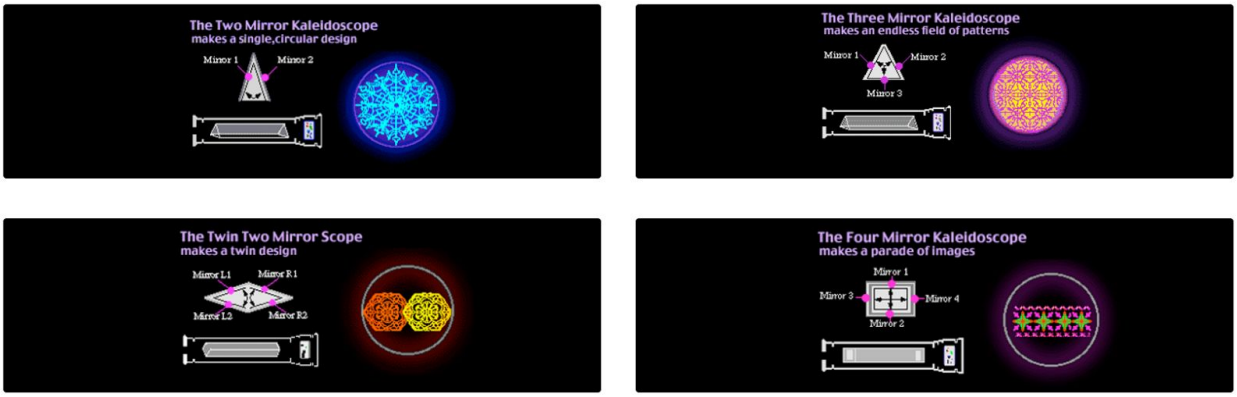
[2] Image source: http://www.schoolphysics.co.uk/age16-19/Optics/Reflection/text/Reflection_/index.html

The Colour Composition of Light

When white light passes through translucent coloured objects, most of the colours are absorbed by the object, except the colour of the object. Thus, light in the colour of the object is allowed to pass through. With various colored objects in a kaleidoscope, different colors are transmitted into the tube.

The Mirror Arrangement

A kaleidoscope may contain two or more mirrors or reflective surfaces positioned at an angle to each other, usually forming a V-shape or a triangle. However, different mirror arrangements are possible, and they create distinct patterns as shown[3]:



[3] Kaleidoscope mirror arrangements and image:
http://www.kaleidoscopesusa.com/about/how-kaleidoscopes-work/

Existing Projects

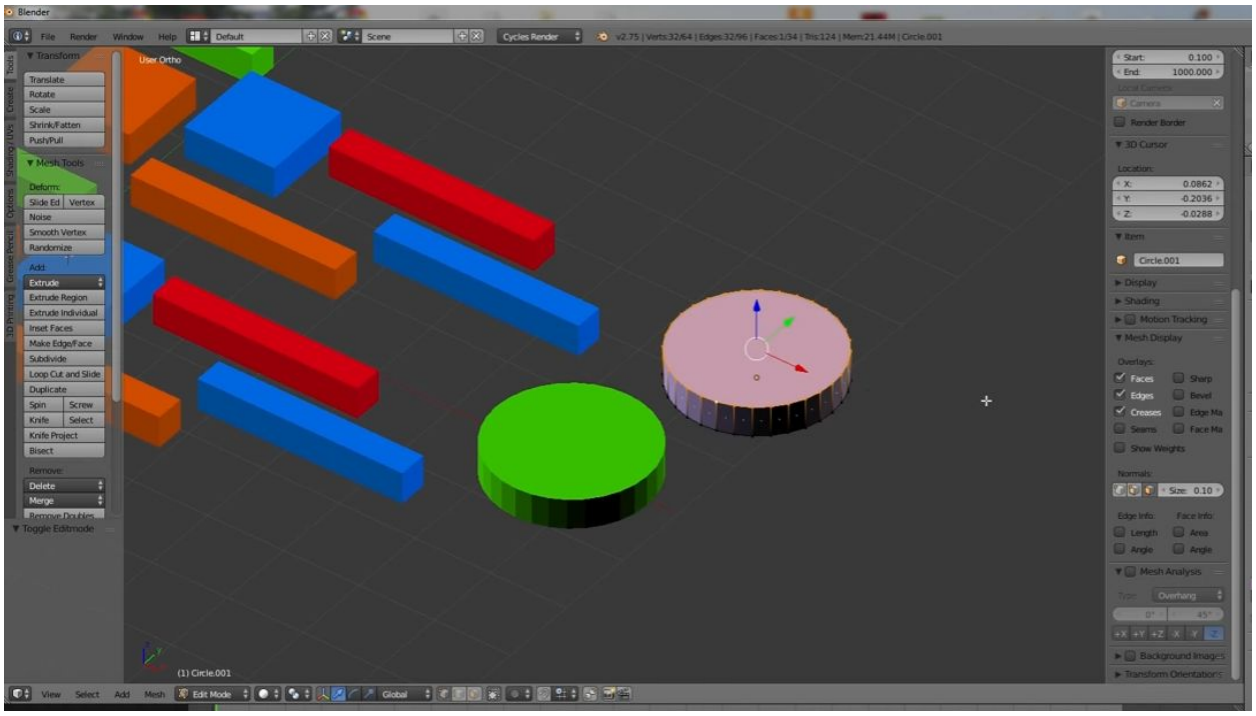
Although there are research studies related to kaleidoscope, such as the SIGGRAPH study on measuring bidirectional texture reflectance with a kaleidoscope[4], there is rarely any investigation on computer-simulated kaleidoscopic graphics. The most related work is a hobbyist's attempt to code for kaleidoscope shader[5]. Thus, we foresee great potential for our digital, interactive, and virtual reality-enabled kaleidoscope viewer.

We searched across Unity, the game engine chosen for the project, for existing kaleidoscope related work. The only results found are the for-profit kaleidoscope patterns in asset store[6]. Despite the lack of available resources in Unity, we found a related tutorial[7] based in Blender, a graphics platform that shares certain similar characteristics with Unity.

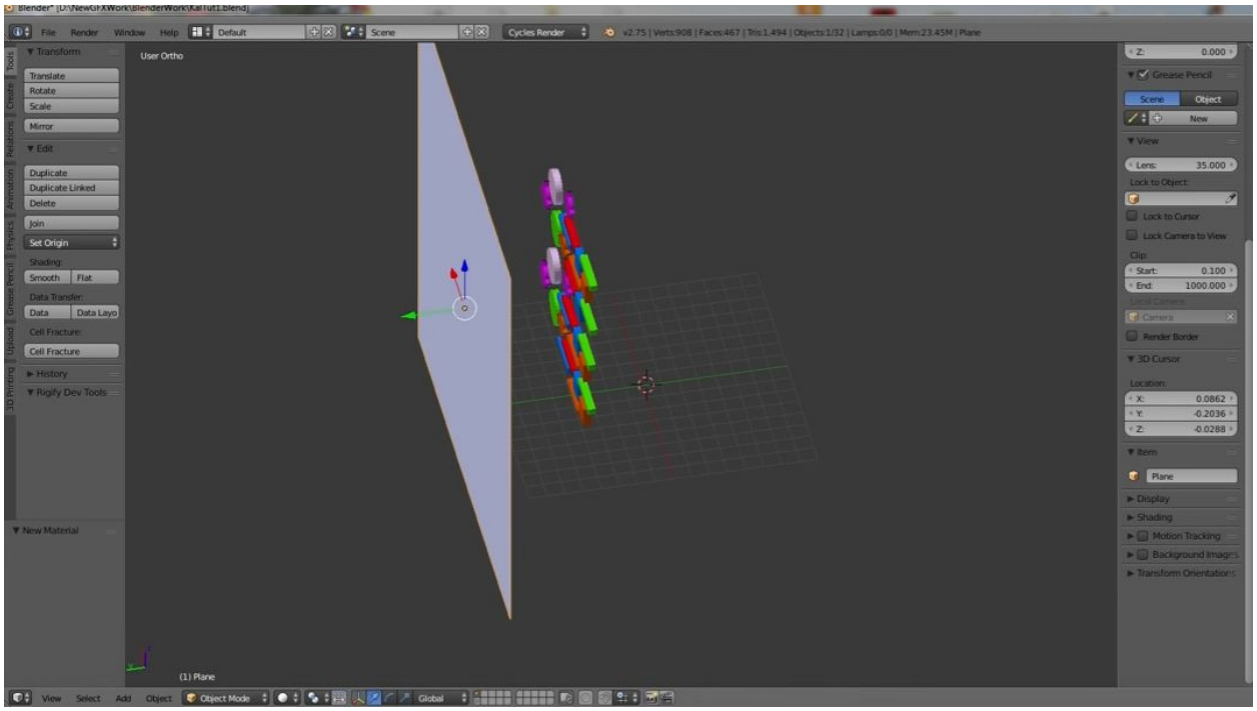
The tutorial video may be summarized as the following steps (image source: screenshots taken from the tutorial video):

[4] Measuring bidirectional texture reflectance with a kaleidoscope: https://nyuscholars.nyu.edu/en/publications/measuring-bidirectional-texture-reflectance-with-a-kaleidoscope
[5] Kaleidoscope shader: http://sizecoding.blogspot.sg/2007/11/kaleidoscope-shader.html
[6] Sample kaleidoscope project in Unity asset store: https://www.assetstore.unity3d.com/en/#!/content/41467
[7] Blender kaleidoscope tutorial: https://www.youtube.com/watch?v=Qm24ep6-i4Y

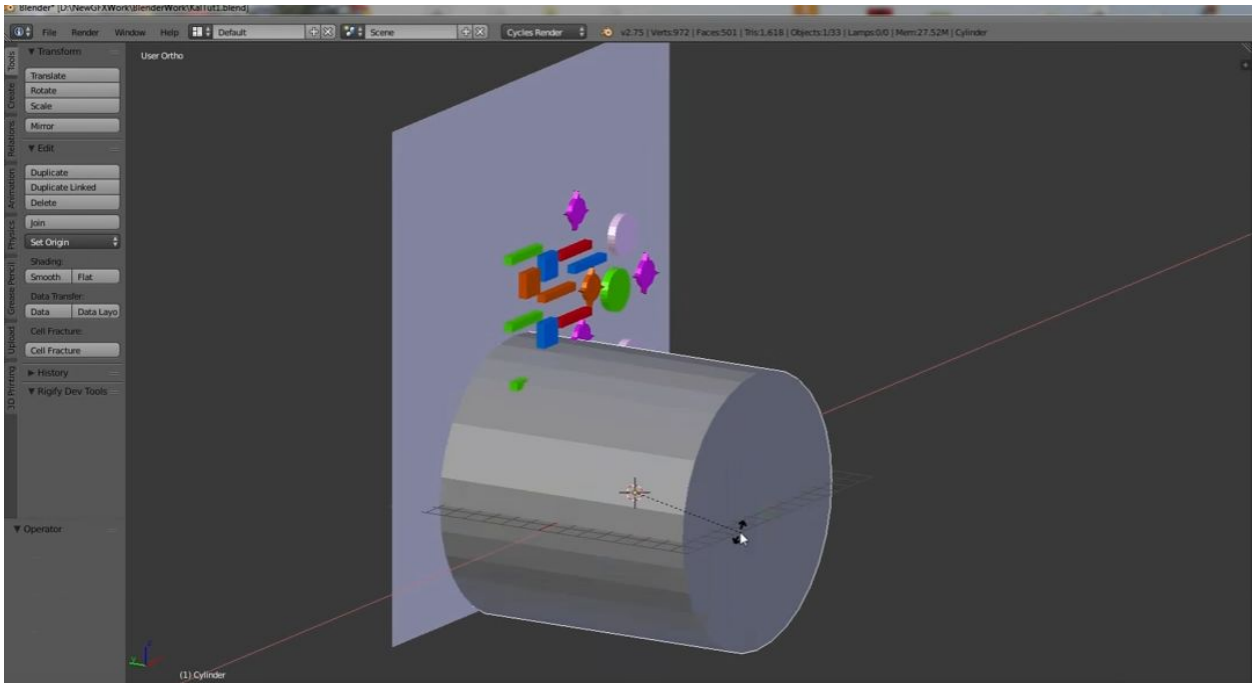
Step 1. Creating colourful 3D objects in basic geometries



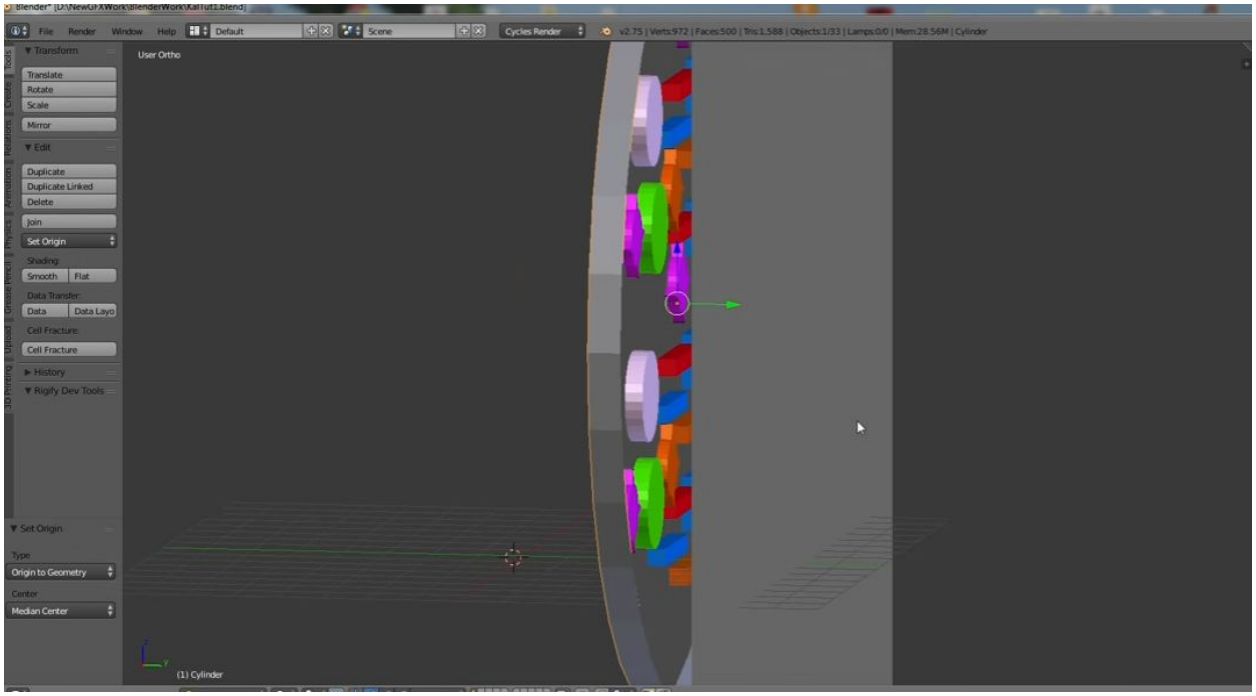
Step 2. Create a background plane behind the objects



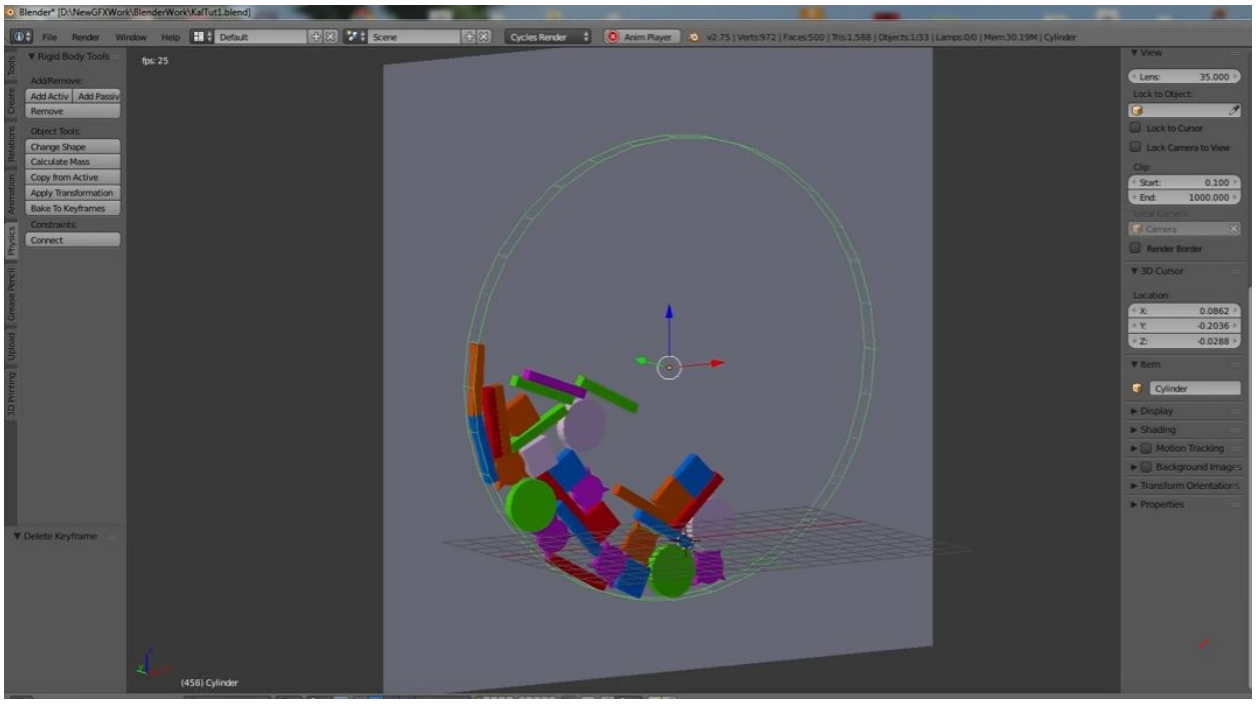
Step 3. Create a cylinder to enclose all objects



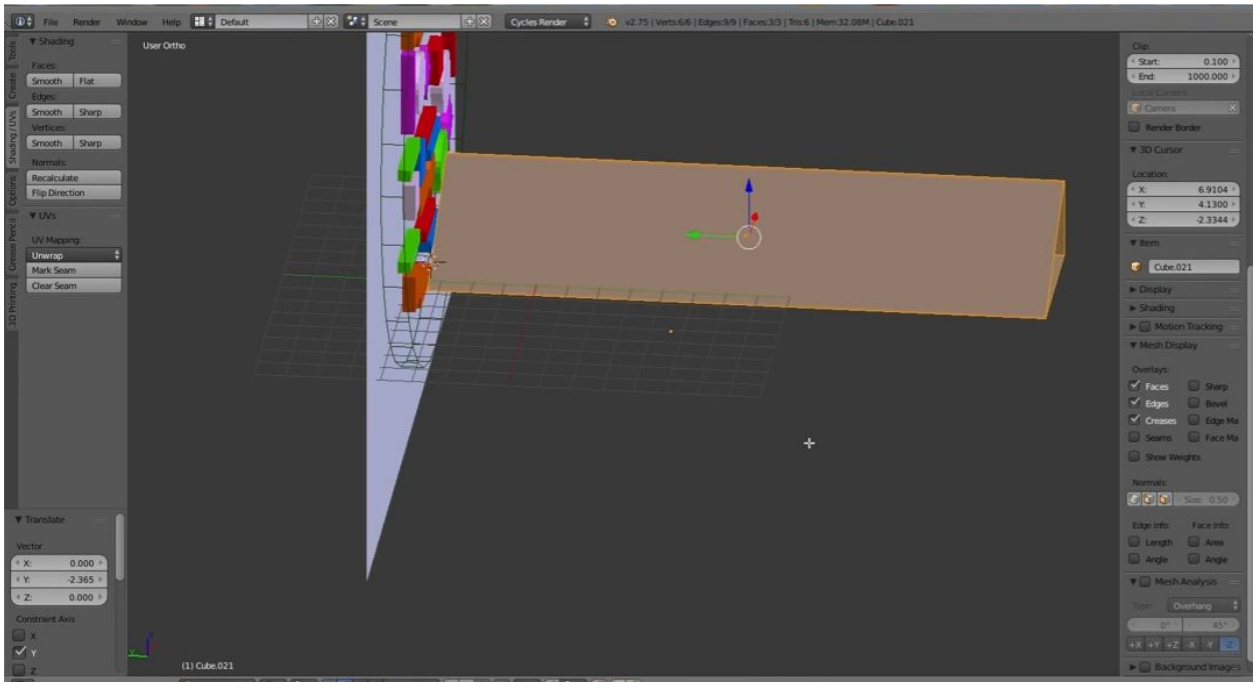
Step 4. Adjust the cylinder so it has the same width as the objects inside



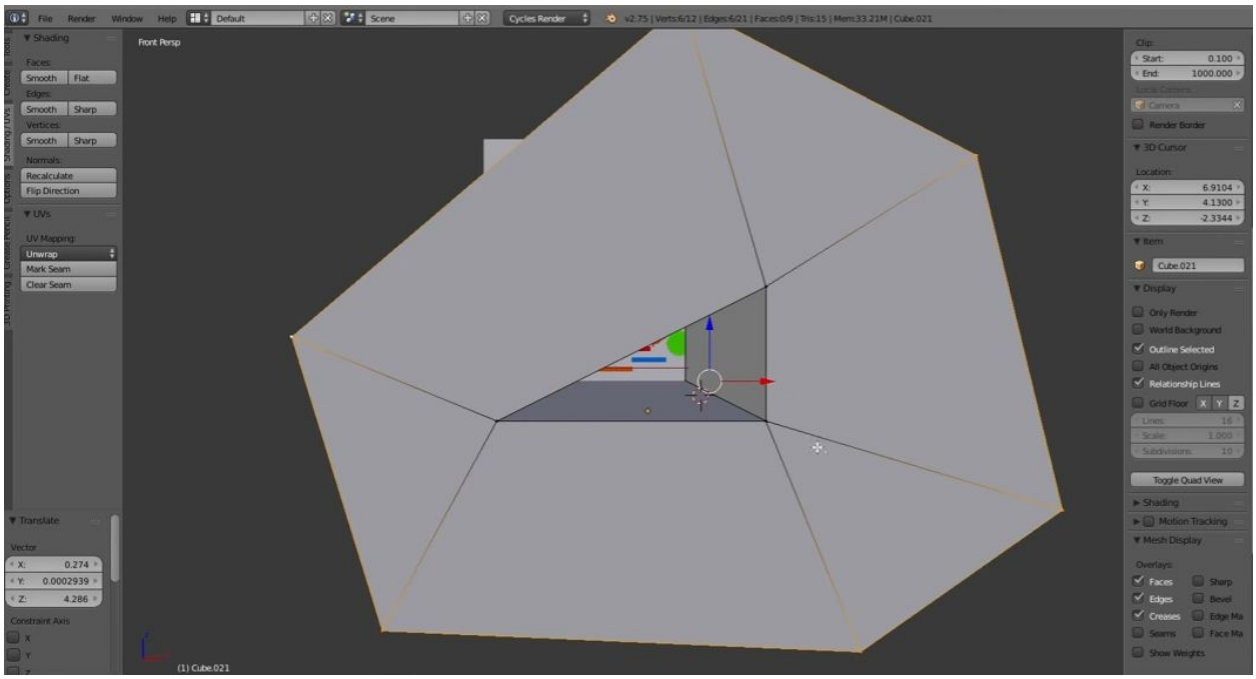
Step 5. Apply gravity to objects so they fall to the bottom of the cylinder



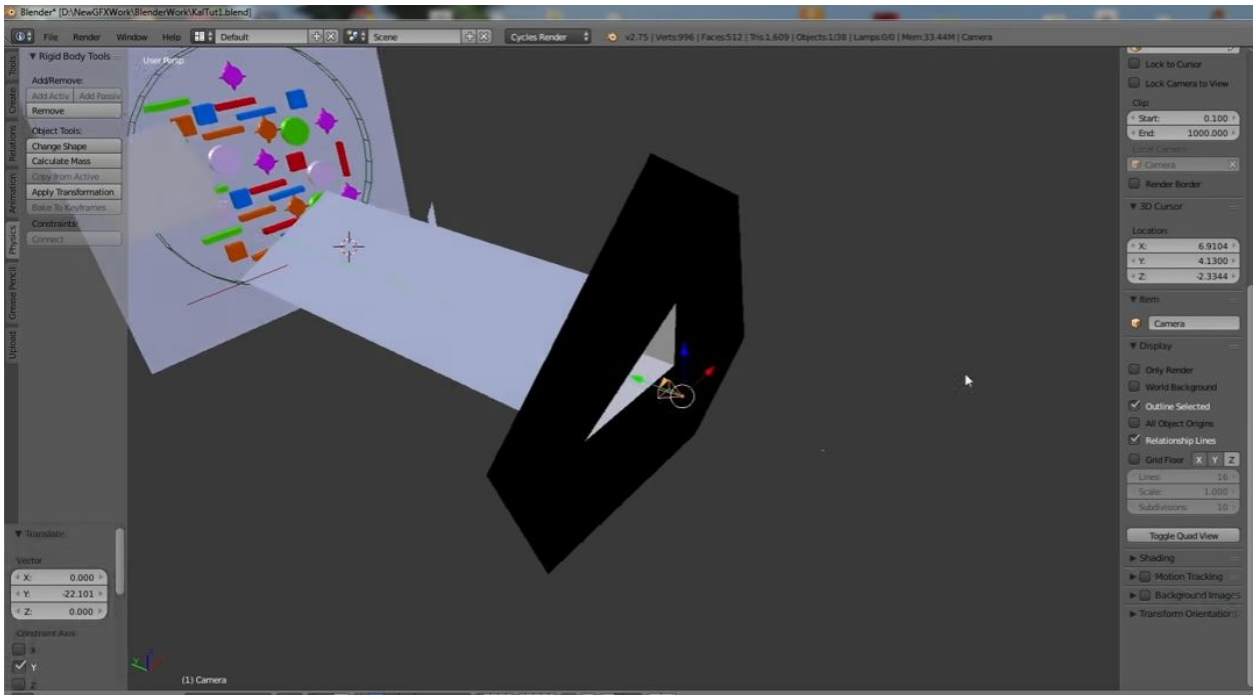
Step 6. Create a prism on top of the cylinder



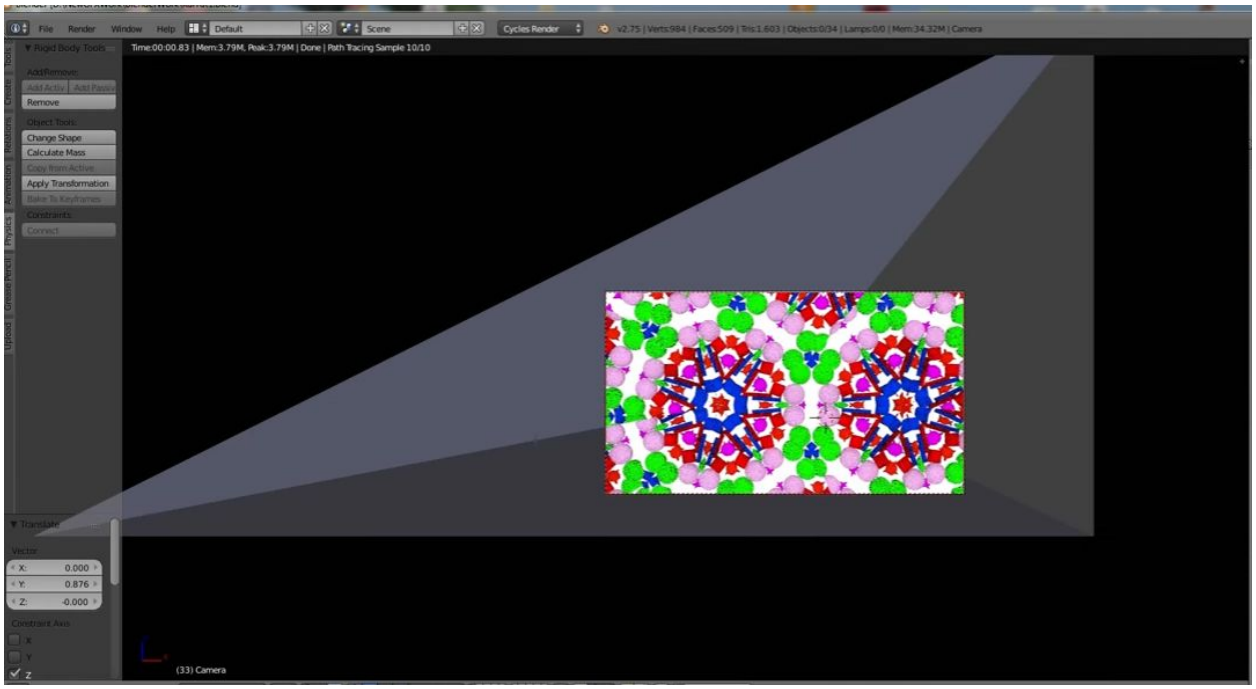
Step 7. Extend on side of the prism so that it covers the objects behind



Step 8. Position the camera in front of the prism opening



Step 9. Check the result from the viewport

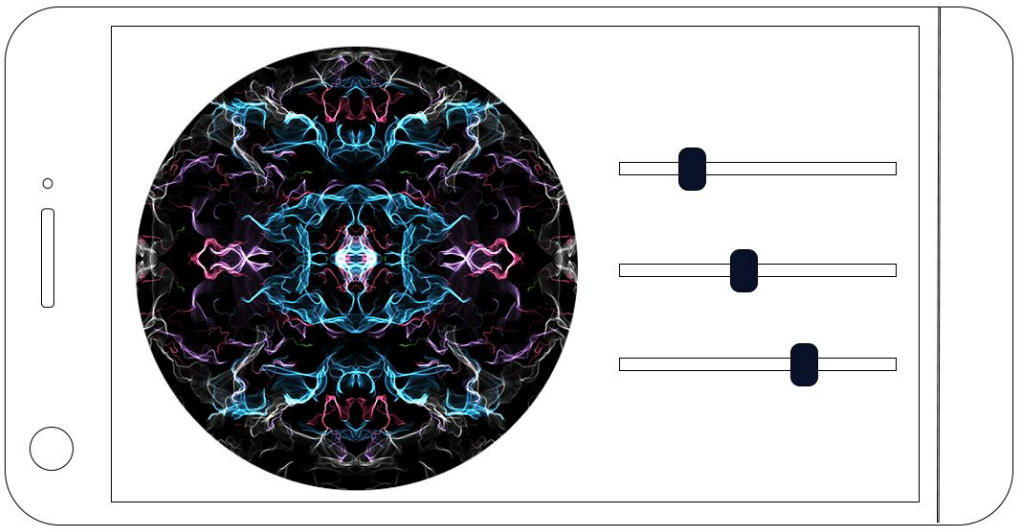


3. PROPOSED APPROACH

User Interface: 2-Dimensional

We will be using Unity and Blender to do this project, and the end product will be an app. There will be two modes- a normal mode, where the user can change parameters and look at the kaleidoscope by turning the phone, and a zoomed in mode (for using with VR glasses) where the kaleidoscope will auto-rotate.

Normal mode will look something like this:

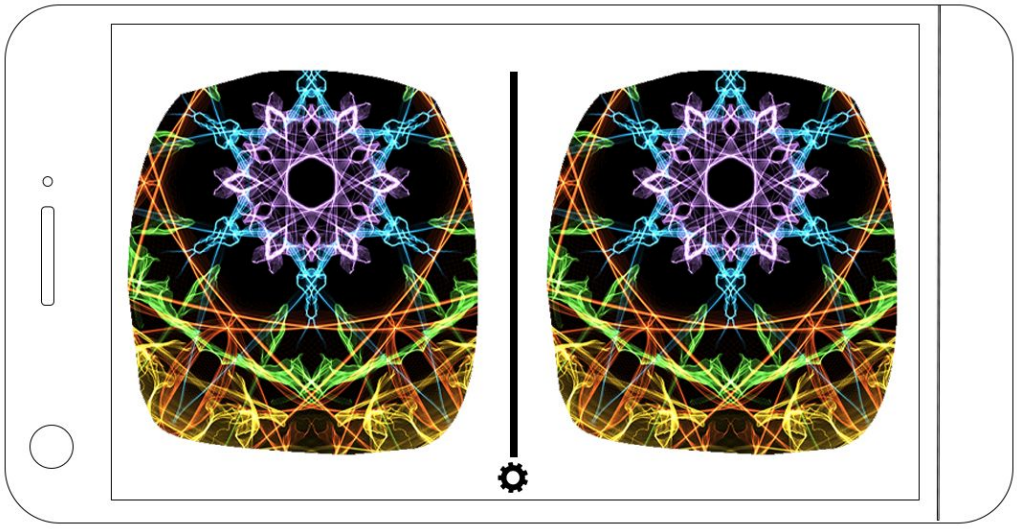


These images are a rendering, and the finished product will have more solid objects, and hence the shapes will look more realistic, and not as flow-y.
There are sliders that can change the number, color and geometry of the objects. We intend to have 2, 3, 4 and more mirrors, more than 3 colors and 4 or more geometry of the objects (spheres, flat pieces, prisms, etc).

A zoomed in view of the app will look like this with auto rotate on for the kaleidoscope view itself.

User Interface: 3-Dimensional

We will also design a 3D user interface for users.There will be a toggle between 2D and 3D interfaces. The 3D interface will be presented as the sketches below:



In this mode, users can either use Google cardboard or any 3D display devices to view the kaleidoscope's patterns where the autorotation is activated so users don't need to manually rotate the lenses. Furthermore, users are free to tilt their head to different direction to check out the inner structure of kaleidoscope.

Components and Parameters

The main components and parameters can be summarized as below:

Viewport

Our program will have 2 view options for the users to toggle, which include zoom in and zoom out view. The difference between this 2 views is the way of users interact with the program.

Manual / auto rotation mode

For zoom out view, users are able to manually rotate the kaleidoscope by tapping on the phone screen and drag in clockwise or anti-clockwise direction to see the changes on image patterns.
For zoom in view, auto-rotation mode will be activated. At first, users will see a smooth transition from zoom out view to zoom in view. Then users will see the auto rotation that's happening inside kaleidoscope which cover up the entire screen.

Objects inside the kaleidoscope

We have considered a few aspects of the design on the objects in kaleidoscope which includes number, color and geometry of the objects.

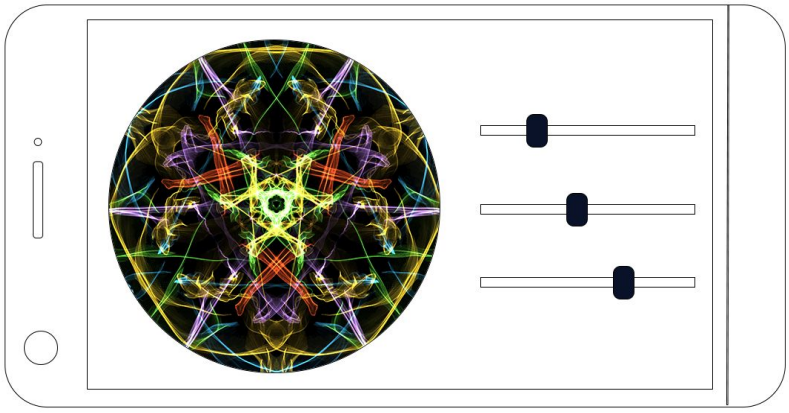
Special objects in kaleidoscope

In reality, we can't find a kaleidoscope that contains an object in between the two end of the scope because it is against the law of physics. This is where the fun part is. We planned to place a special object could it be a prism, jelly, liquid, etc. In virtual world this will give us an unexpected scene which we can never witnessed in real world and it definitely will bring a whole new experience to the user.

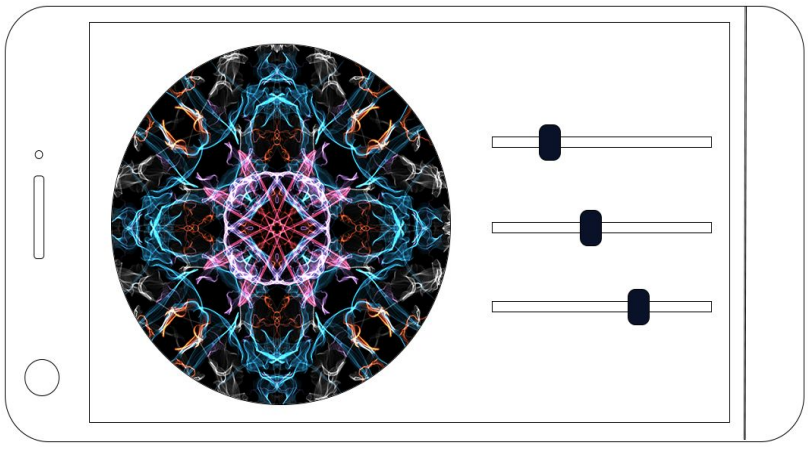
Number and angle of mirror

The number and angle of mirror are changeable where the users are allowed to adjust according to their preferences.

Just to show the difference between number of mirrors, here is a rendering of a kaleidoscope with 3 mirrors and another one with 4 mirrors. We hope to recreate this in the project.



3 mirrors



4 mirrors

Shape of mirror

This is also one of the interesting parameters that we want to introduce. In reality, we can only find flat mirror inside kaleidoscope. Here we would like to show users a different effect of kaleidoscope that is different from the reality by using a curved mirror could it be a concave or a convex mirror.

Light source

This is another interesting parameters of this project where we planned to provide lights source with different option of colors for the users to choose. With this different color of light sources, we aim to create a whole new effect that cannot be created in real life.

System Requirements

For the development of this project, we are going to use 2 softwares which are Blender for modelling and Unity 3D For designing and creating the project. For users, they will need a PC or an Android devices to run the project. On top of that, VR scope is also required for 3D effects.

4. **MILESTONES**

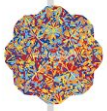
START

27 Mar  Basic kaleidoscope structure

30 Mar  Virtual reality settings for project

08 Apr  Adjustable variables implementation

10 Apr  Transition between view modes

13 Apr  Generate innovative patterns

X

END

REFERENCES

- Kaleidoscope etymology: https://en.wikipedia.org/wiki/Kaleidoscope
- Reflection mages: http://www.schoolphysics.co.uk/age16-19/Optics/Reflection/text/Reflection_/index.html
- Kaleidoscope mirror arrangements and image: http://www.kaleidoscopesusa.com/about/how-kaleidoscopes-work/
- Measuring bidirectional texture reflectance with a kaleidoscope: https://nyuscholars.nyu.edu/en/publications/measuring-bidirectional-texture-reflectance-with-a-kaleidoscope
- Kaleidoscope shader: http://sizecoding.blogspot.sg/2007/11/kaleidoscope-shader.html
- Sample kaleidoscope project in Unity asset store: https://www.assetstore.unity3d.com/en/#!/content/41467
- Blender kaleidoscope tutorial: https://www.youtube.com/watch?v=Qm24ep6-i4Y
- Kaleidoscope icons for milestone: Google images