

# Geometric Texture Synthesis by Example

*Ishan Mehta, Tushar Singh*

Our project is on geometric texture synthesis. We are given an input analogy which consists of 2 figures. One is a simple 3d object and another is the same object with some 3d textures mounted on it (like thorns, pits etc). Then we find out the type of texture from the difference in coordinates of both the figures and store it in some other structure. Then we create another 3d object with the same textures pasted on its surface, obtained from the differences in coordinates.

All types of 3d objects, whether textured or not, are represented in separate classes.

Firstly, we assign an offset to every object used, to keep them separate in world space. Then we set other parameters like radius, height etc. for rendering of the object through its parametric equations.

In the case of input analogy objects, we use variable mode in function `fill_points()`, which generates coordinates of the object for rendering. If mode value is 0, then we create the object by just using its parametric equation. And if its value is 1 then we modify the coordinates in a determined way to create a geometric texture on it.

Then we call the function `get_analogy()`, which measures the differences in the coordinates in the input objects. Then it converts them into corresponding values of  $u$  and  $v$  for mapping into a vector analogy. That is our input analogy vector.

Finally, we create our output objects and set offsets and parameters like radius, height etc. The first one is rendered only by its parametric equation. The mode

value is 0. The other one is rendered by mapping. When the mode value is 2 then, we convert the u,v values of input analogy to the coordinates of the output object to get the same texture on it as given on the input object.

#### MILESTONES:

1. Rendering different shapes from their parametric equations(plane, sphere, cylinder)
2. Created equations to map all 3d points of the shapes to 2d planes.
3. Captured the 3d geometric analogy of input objects
4. Converted the same analogy to 3d points of the dummy object.
5. Tried and tested on different sets of unique geometrical patterns and objects.
6. Applied lighting and shading to reflect the actual dynamics of the rendered object.

REPO link:

[https://github.com/CSE-333-Computer-Graphics-2022/Texture\\_synthesis](https://github.com/CSE-333-Computer-Graphics-2022/Texture_synthesis)