Fractal Generator

Vlasis Gogousis

Contents

Aim of project

The purpose of this project was to develop an interface which allows the user to generate fractal visualizations and observe how parameters affect the end result. It was inspired by the use of Mandelbrot patterns on objects morphing into different shapes in the feature film Doctor Strange (2017). The script can be used as a tool to see how fractal formulas behave and -with some modifications- could even be applied on surfaces and textures. Lastly, the script is broken down to modules, so that additional fractals can be easily added in the future.

Planning

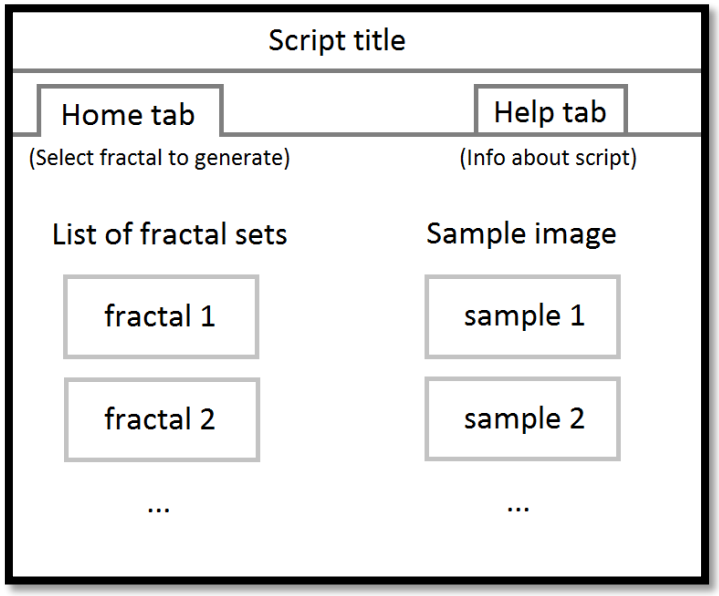

Figure 1: Fractal generator main window

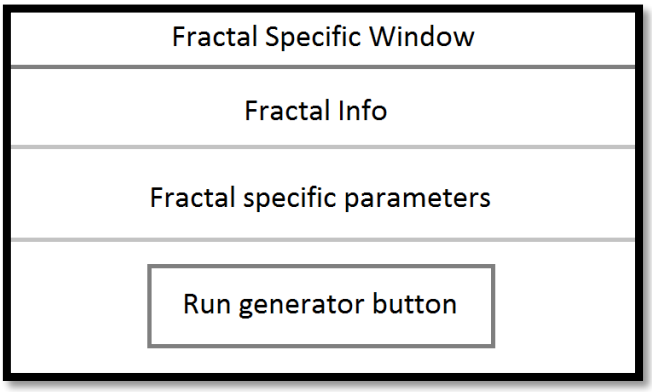

Figure 2: Fractal specific window to setup experiments

Instructions

1. Copy all contents of project folder into Maya's script directory
2. Execute Maya
3. Open up your script editor (either shortcut in lower right corner of Maya environment or typically: Widnows> General Editors> Script Editor)
4. Run the following command:

```
import fractal_generator
```

5. From "Fractal Generator" window you can:
    a. Go to the "Home" tab to select a fractal set from the list and setup parameters from popup window to run experiment

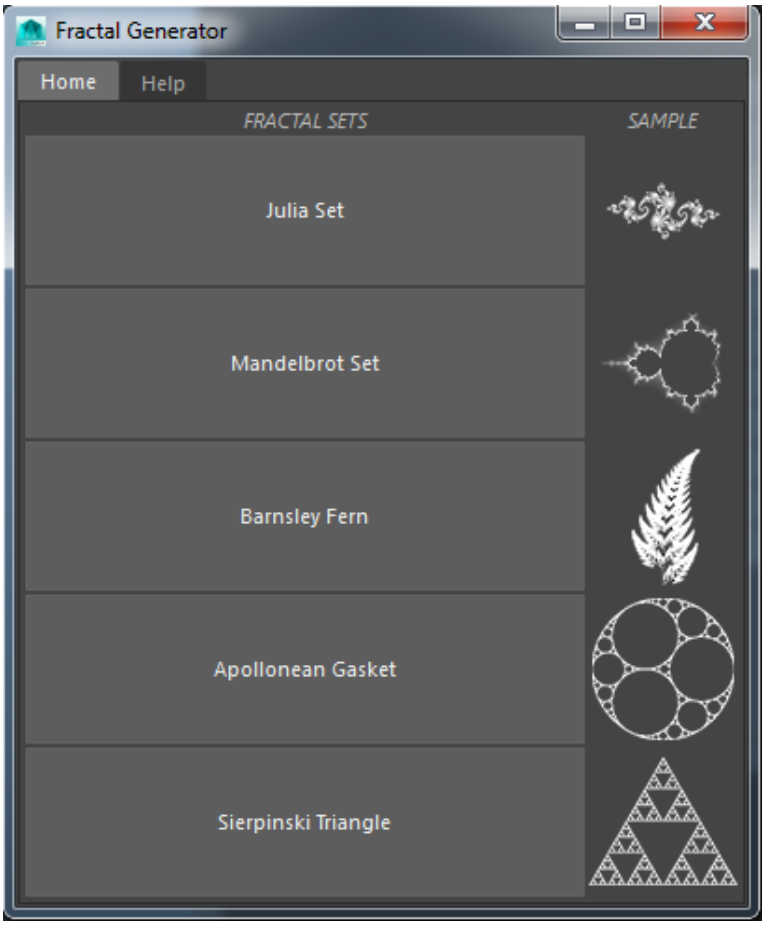

Figure 3: Home tab

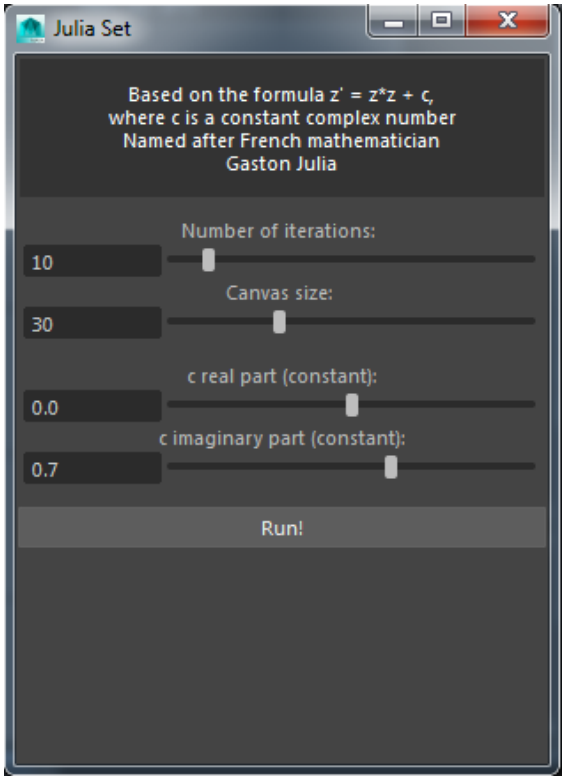

Figure 4: Fractal setup window

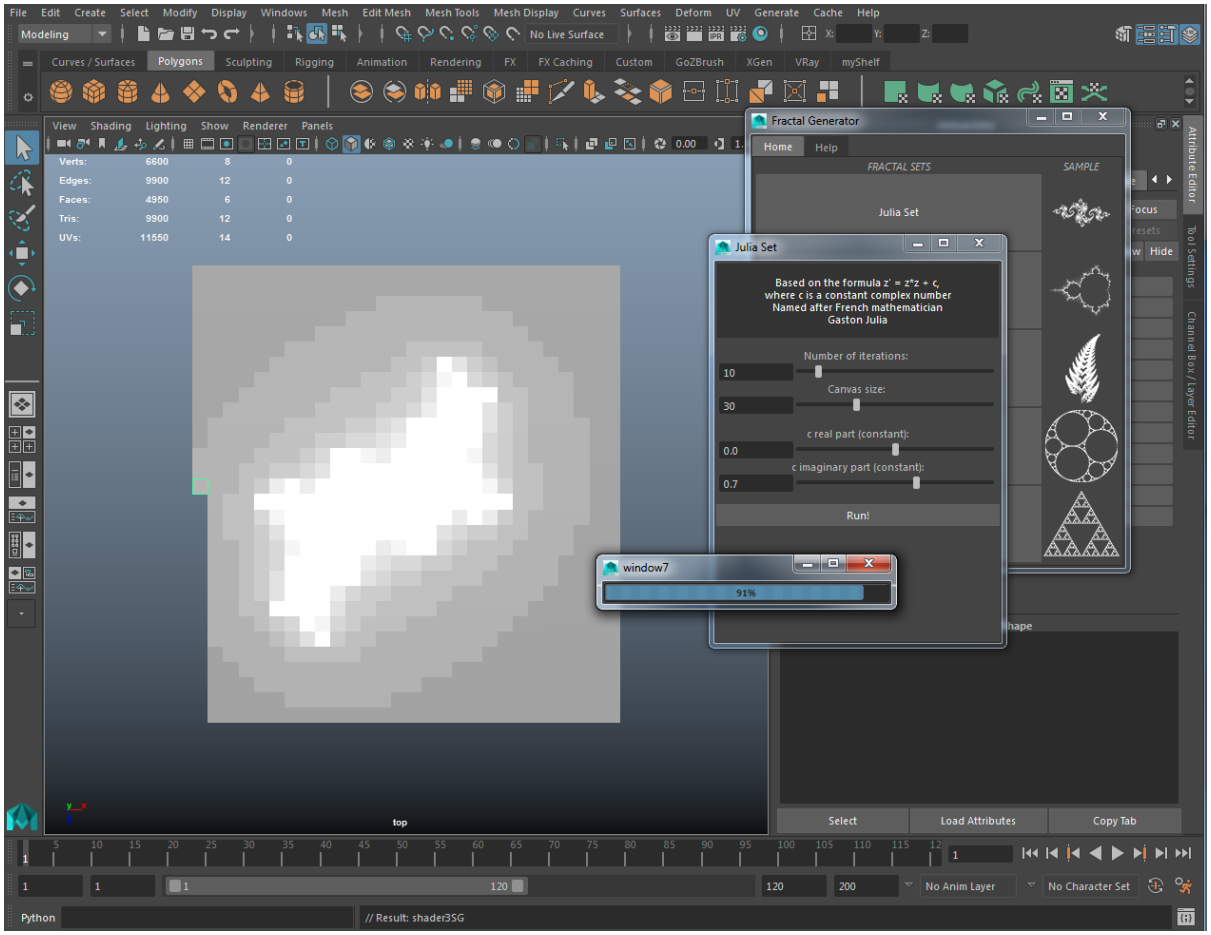

Figure 5: Example fractal visualization

b. Go to the “Help” tab to view information about fractals and the script

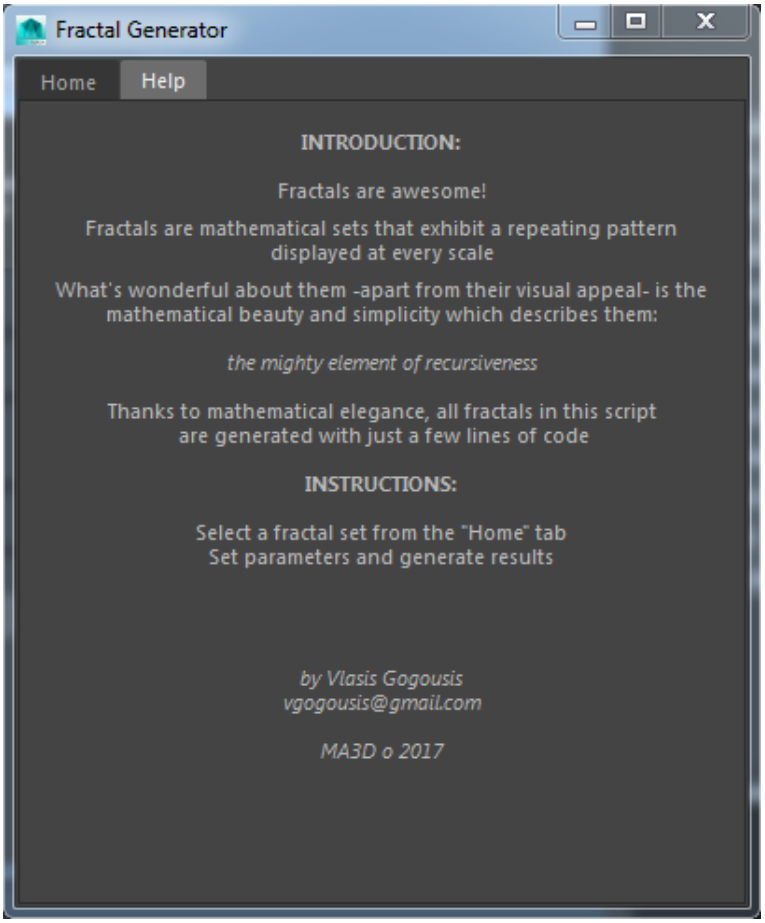

Figure 6: Help tab

6. Troubleshooting: If unsure where your Maya’s script directory is, you can run the following lines of code from Maya’s script editor to view all the directories in Maya's path:

```
import sys
print sys.path
```

   Copy contents of project folder into any of the returned directories and follow steps above
7. Scripts have been formatted so that if imported help is available through script editor. Example:

```
import barnsley_fern
help(barnsley_fern)
```

Algorithms

-The following scripts were re-adjusted from online sources in order to run in Maya:

a. **mandelbrot_set.py** (Borini 2010)
b. **julia_set.py** (Burke 2013)
c. **apollonian_gasket.py** (FB 2012)

-The following scripts were developed by following the original algorithms

a. **barnsley_fern.py**
   1. Set starting point $(x, y) = (x_0, y_0) = (0,0)$
   2. Randomly calculate one of four standard affine transformations:

$$f_1(x, y) = \begin{bmatrix} 0.00 & 0.00 \\ 0.00 & 0.16 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$f_2(x, y) = \begin{bmatrix} 0.85 & 0.04 \\ -0.04 & 0.85 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.00 \\ 1.60 \end{bmatrix}$$

$$f_3(x, y) = \begin{bmatrix} 0.20 & -0.26 \\ 0.23 & 0.22 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.00 \\ 1.60 \end{bmatrix}$$

$$f_4(x, y) = \begin{bmatrix} -0.15 & 0.28 \\ 0.26 & 0.24 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.00 \\ 0.44 \end{bmatrix}$$

   3. Plot the resulting point in space
   4. Update (x, y) to resulting point of step 3
   5. Repeat above steps for any number of iterations

b. **sierpinski_triangle.py**
   1. Set three points in space to form an equilateral triangle
   2. Set the current point to $(x_0, y_0) = (0,0)$
   3. Randomly calculate the midpoint of the current point and one of the triangle's vertices
   4. Plot the midpoint
   5. Set the current point to the midpoint
   6. Repeat above steps for any number of iterations

-numpy library was used to help compute above algorithms

References

Borini, S., 2010. *For The Science* [online]. Available from: http://forthescience.org/blog/2010/07/12/the-mandelbrot-set-in-python/ [Accessed 13 February 2017]

Burke, T., 2013. *Batchloaf* [online]. Available from: https://batchloaf.wordpress.com/2013/02/10/creating-julia-set-images-in-python/ [Accessed 13 February 2017]

Doctor Strange, 2016. [film, DVD]. Directed by Scott Derrickson. Marvel Studios, Walt Disney Studios Motion Pictures

FB36, 2012. *Active State* [online]. Available from: http://code.activestate.com/recipes/578016-apollonian-gasket-fractal-using-ifs/ [Accessed 13 February 2017]