

## The application of Fractal images based on the Mandelbrot set and its variation

Chao Yang <sup>a</sup>, Zhong Zhou <sup>b,\*</sup>

College of Science, Zhongyuan University of Technology, Zhengzhou, China

<sup>a</sup>2252007606@qq.com, <sup>b</sup>southmonarch@qq.com

---

*Abstract: The fractals images produced from the famous M (Mandelbrot) set are usually shown for the beauty of the mathematics. However, with the popularization of 4K monitors, the more fined fractal details and it's application are investigated in this paper. A factor of the M set is introduced for the variation, 71 colormaps developed by matplotlib 2.2 are using for the rendering of the fractals. These amazing images may be enlarged on a local region without losing clarity, meanwhile, a 4K fractal bitmap file with megabytes only needs a few bytes to be stored. We have designed such a python program base on the matplotlib graphic module, an arbitrary region can be selected by the mouse to zoom it, a series of beautiful fractal images can be selected for costume designing, wallpaper decoration and so on.*

*Keywords: fractal images, colormap, matplotlib, costume designing, decoration.*

---

### 1. INTRODUCTION

The fractal images from the M set may not only show the beauty of the mathematics, but also can be applied to common life, for example, the patterns for costume designing and wallpaper decoration. Due to the development of technique of computer vision art [1], the method of color-mapping designing has made a big progress [2-4]. Until now, 71 kinds patterns of colormap has been packed in the newest released version of Matplotlib, the famous graphic module in Python ecosystem. Besides the application on visualization of big data, the color-mapping patterns may be used to rendering the Mandelbrot fractals, the same scenery using different patterns of color-mapping will produce utterly different fractal images, it will enrich greatly the patterns of fractal images from the M set. In addition, the variation factor  $\alpha$  takes positive integers, the fractal images are normal and beautiful with diversity of pattern, if  $\alpha$  takes positive fractions, the fractal images include breaking partly, however, the pretty part can be easily found. All the fractal images with optional colormap patterns could be produced by a Python program presented in this paper, the local region can be zoomed without losing clarity. The beautiful fractal images may be selected by user for logo or costume designing, wallpaper decoration and other design material.

### 2. THE M SET AND ITS VARITION

#### 2.1 The iteration system and six parameters

The M set comes from the following complex iteration [5],

$$W_{n+1} = W_n^\alpha + c \tag{2.1}$$

where  $W_0 = x + y*i$  is a point on the plane, after several iterations, if  $|W_{n+1}| < d$  holds, we call  $W_0 \in M$  (Mandelbrot set). Where,  $d$  takes a fixed number 10, the number of iteration can take 30 for calculation. If  $\alpha$  takes 2 for the normal M set, other positive real number for the variation of the M set. The definition of  $\alpha$  is to enlarge the library of fractal images of the M set. We found that if  $\alpha \geq 2$  and  $\alpha \in N$ , the images show no breaking and perfect. (See Fig.1)

Six parameters determine a fractal image,  $V_i = [x_{min}, x_{max}, y_{min}, y_{max}, \alpha, cm]$ , where  $(x_{min}, y_{max})$  is the left-top coordinates of a given rectangle region,  $(x_{max}, y_{min})$  is the right-bottom of it,  $\alpha$  is the variation factor,  $cm$  is a string variable from the 71 colormaps defined in matplotlib. For example, in Fig.1,  $V_i = [0.1979, 0.43469, 0.47142, 0.55255, 2, 'gist\_rainbow']$ , we can reproduce fig.1 by  $V_i$  in the computer program, also save it as a png file.

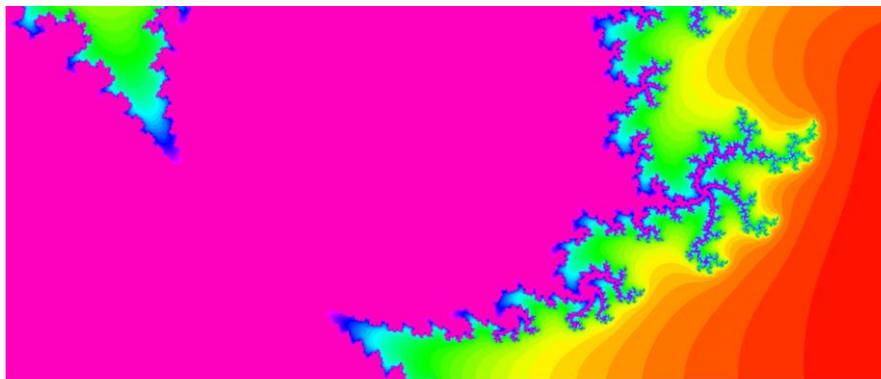


Fig.1 Six parameters determine a fractal image.

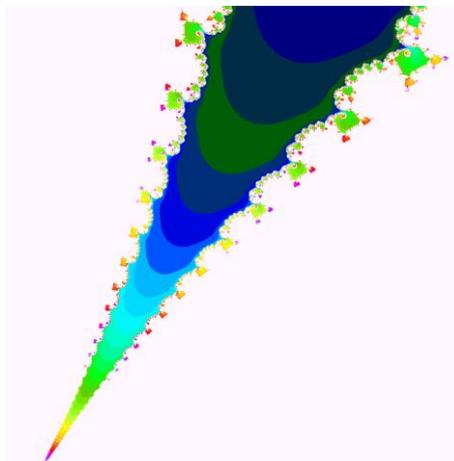


Fig.2 The fractal image with  $\alpha=7$

Fig.2 shows the case of  $\alpha$  takes 7, the variation of M set, have still a regular or normal fractal, no breaking found. In fact, a lot of experiment have been done, an empirical result is that if  $\alpha \geq 2$  and is nature number, the fractal is usually normal, elsewise, if  $\alpha$  is a fraction or irrational number, the breaking is always coming.

Fig.3 shows the case of  $\alpha = \frac{7}{3}$ ,  $vi = [-0.04439, 0.024878, 0.72164, 0.77388, \frac{7}{3}, 'jet']$ , the breaking makes the image not pretty, so letting  $\alpha \geq 2$  and taking the nature number is reasonable.

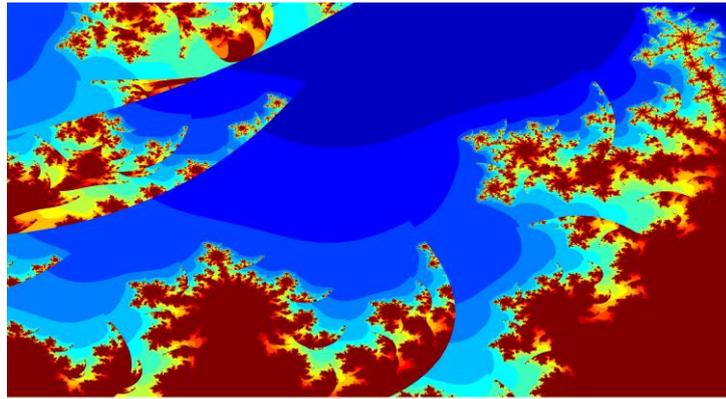


Fig.3 The “ugly” fractal breaking for  $\alpha = 7/3$

## 2.2 How to make the zoomed part clearly

As well known, the bitmap would become blurred if the part of it was zoomed several times. However, by using of the self-similarity of M fractal set, the image displayed on the screen will hold the extra definition when the local region is enlarged. This property would remain unchanged, if the two points were considered.

First, the following python codes should be executed to get the maximum resolution of the current monitor, it supports the 4K monitors and platforms of windows, mac and linux.

```
from PyQt5.QtWidgets import QWidget, QApplication, QDesktopWidget
import numpy as np
screen = QDesktopWidget().screenGeometry()
y0 = np.linspace(ymin,ymax,screen.height())
x0 = np.linspace(xmin,xmax,screen.width())
x,y = np.meshgrid(x0,y0)
```

For an arbitrary rectangular region in fig.4, the discretization should be done by the maximum resolution screen.width() and screen.height().

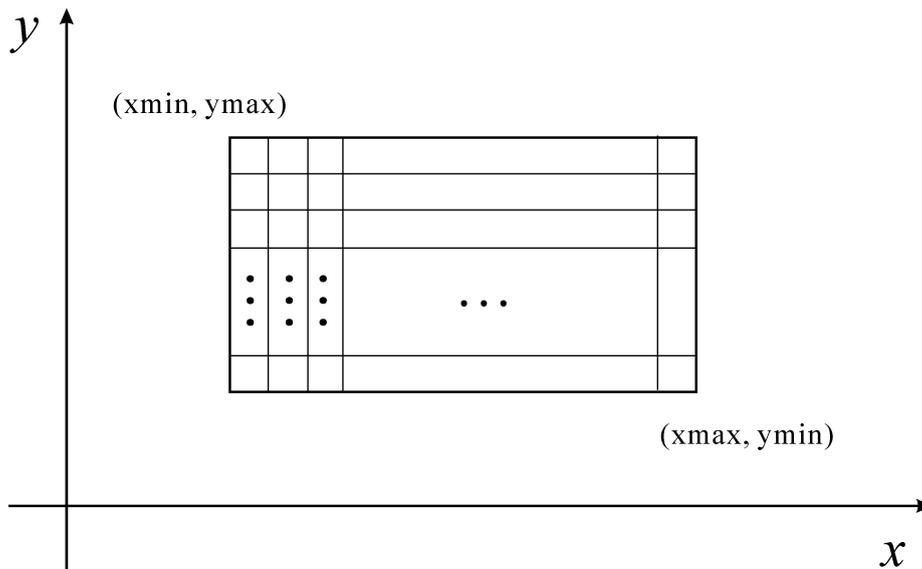


Fig. 4 Discretization of the rectangular region

Second, if  $y_{max} - y_{min} < 2^{-52}$  or  $x_{max} - x_{min} < 2^{-52}$  hold, the zooming action would fail, the so-called unlimited enlargement is terminated at this step for the limitation of the accuracy in double

precision. In order to solve this problem, the famous C++ high precision Gmp and Ntl library [] should be used. In python environment, the similar module gmpy2 is recommended to import.

### 2.3 The rich color mapping patterns defined in matplotlib

The boundary of the M set is needed to show by the color-mapping patterns. For examples, 18 kinds colormap patterns are defined in Matlab. However, due to the contribution of the artists and experts from the computer vision, more and more colormap patterns are designed for computer graphic art. In table 1, 71 kinds patterns of colormaps defined in matplotlib are listed.

Table 1 71 kinds patterns of color mapping

1	Accent	13	PRGn	25	RdPu	37	afmhot	49	gist_ncar	61	ocean
2	Blues	14	Paired	26	RdYlBu	38	autumn	50	gist_rainbow	62	pink
3	BrBG	15	Pastel1	27	RdYlGn	39	bone	51	gist_stern	63	plasma
4	BuGn	16	Pastel2	28	Reds	40	brg	52	gnuplot	64	prism
5	BuPu	17	PiYG	29	Set1	41	bwr	53	gnuplot2	65	rainbow
6	CMRmap	18	PuBu	30	Set2	42	cool	54	gray	66	seismic
7	Dark2	19	PuBuGn	31	Set3	43	coolwarm	55	hot	67	spring
8	GnBu	20	PuOr	32	Spectral	44	copper	56	hsv	68	summer
9	Greens	21	PuRd	33	YlGn	45	cubehelix	57	inferno	69	terrain
10	Greys	22	Purples	34	YlGnBu	46	flag	58	jet	70	viridis
11	OrRd	23	RdBu	35	YlOrBr	47	gist_earth	59	magma	71	winter
12	Oranges	24	RdGy	36	YlOrRd	48	gist_heat	60	nipy_spectral		

Fig. 5 shows the fractal images for  $vi=[0.3668,0.37112,0.51847,0.52115,2,cm]$ , cm takes string corresponding 1 to 71 from table 1 respectively.

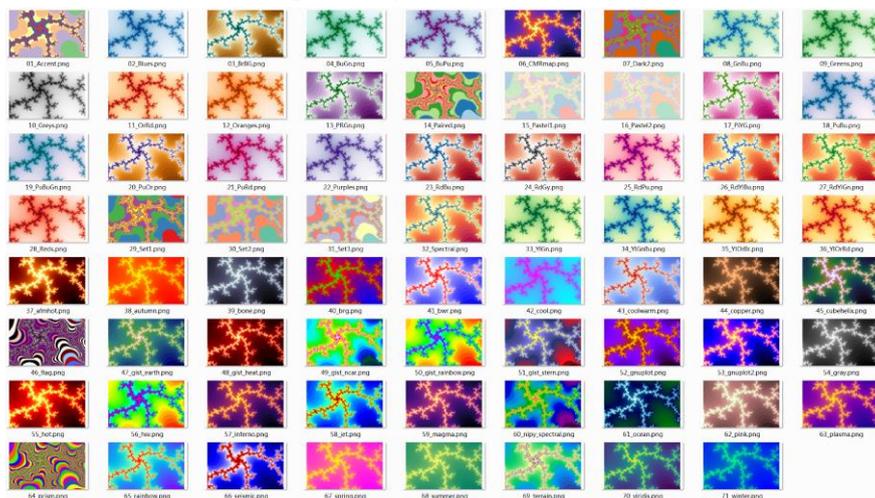


Fig. 5 The same fractal images with different rendering effect by the 71 kinds patterns of color mapping

### 3. The APPLICATIONS OF FRACTAL IMAGES

Fig.6 shows an amber fractal image for  $vi=[0.13253447, 0.13269070418, 0.77098768, 0.7711221435, 3, 'Oranges']$ ; Fig.7 may be selected for wallpapers on children's rooms,  $vi=[-0.4242426, -0.4241919, 0.68147, 0.681504, 6, 'Pairs']$ ; Fig.8 shows the fractal art of wallpaper,  $vi= [-0.735913, -0.633978,$

-0.3886267, -0.320670127 ,2,'nipy\_spectral'].

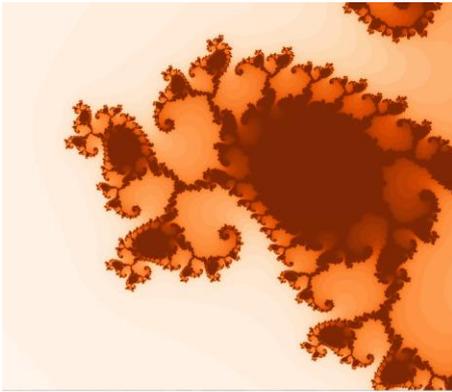


Fig. 6 Amber fractal



Fig. 7 Wallpaper for children

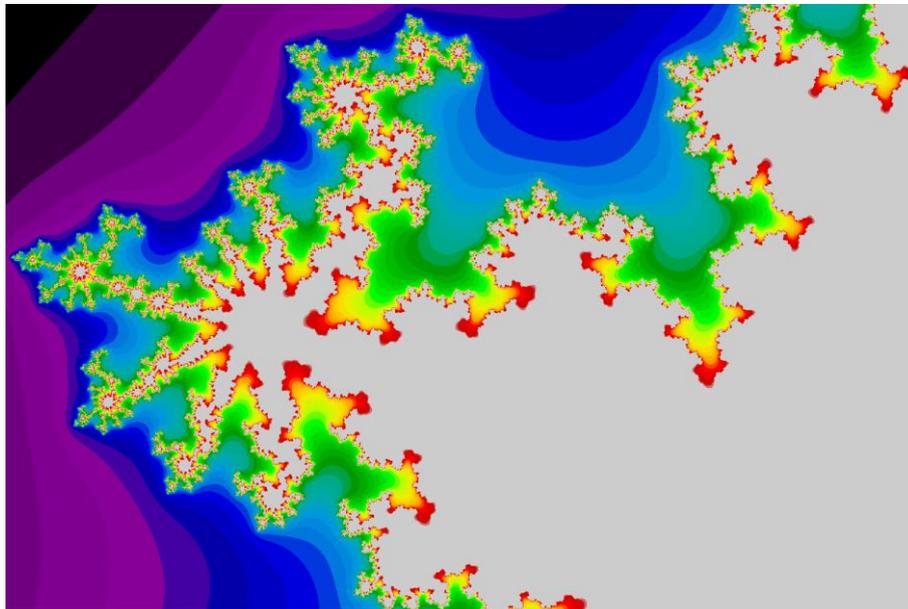


Fig. 8 The fractal art of wallpaper

Here, we only presented several fractal images. The variation factor  $\alpha$  and rich patterns of color mapping will lead to the plentiful pretty fractal maps from the M set for a given region, in addition, the users could select an arbitrary rectangle domain to zoom unlimitedly, so more and more pretty images may be produced and classified into some image libraries ,such as wallpaper library for children and decorators, picture material library for logo or costume designers, sometimes, the poets and artists may have drawn their inspiration from the fractal image libraries.

#### 4. CONCLUSION

Today, using computer program to get elegant fractal images of the M set is based on the following reasons. First, the 4K monitors are popular and should be used to get more higher definition of the pictures; second, the variation factor  $\alpha$  and 71 kinds patterns of color mapping make a dramatic increase on the number of pretty fractal images of the M set. The corresponding wallpaper library and picture material library are useful for many people.

At present, the pretty fractal images from the M set can be obtained by the computer program, while the manual intervention is needed. The ugly or exquisite fractal images should be distinguished

according to the esthetics viewpoint. How to make an intelligent automatic selection for the pretty pictures from the computer program, is a touchy and interesting issue, however, it will become more and more feasible due to the development of deep learning, people should let the computer know what patterns correspond to beautiful or ugly pictures.

#### **REFERENCES**

- [1] K. Peter., “Good Colour Maps: How to Design Them.”, *Computer Science*, 2015, Vol. 6 (12), p511-523
- [2] B. David and Russell M. Taylor, “Rainbow color map (still) considered harmful”, *IEEE Computer Graphics and Applications*, 2007, Vol. 27(2),14
- [3] A. Cynthia, Brewer. “Spectral schemes: Controversial color use on maps”, *Cartography and Geographic Information Systems*, 1997, Vol. 24(4), 203-220
- [4] G.-A., Paul, “A colour alphabet and the limits of colour coding”, *Colour: Design & Creativity*”, 2010, Vol. 5(1),23
- [5] K.J. Falconer, “Techniques in Fractal Geometry”, [M] New York:John Wiley and Sons Press,1997