

An Artistic Approach to Julia Sets Through Computer Programming

Eduardo Adam Navas-López

(Mathematics Department, University of El Salvador, El Salvador, Central America)

Abstract: This article mentions how the resurgence of the study of Julia Sets was given thanks to the diffusion of personal computers and then proposes the artistic pathway is proposed to increase and enrich the mental image on Julia Sets. This pathway includes the mathematical, the playful, the artistic and the computational approach; and it is argued that they are not disjointed or dissociated despite general rejection by traditional academic communities and art critics communities. Some significant collections of Computational Art or Computer-Generated Mathematical Art are mentioned, and finally four artistic creations based on Julia Sets are presented as an example using the CFDG language.

Key words: computer graphics, fractals, Julia Sets, CFDG, computer generated mathematical art

1. Introduction: Julia Sets, Mathematics, Art and Computers

As Bergweiler (1993) explains, the study of iteration theory is fundamental in mathematics, and its classic problem is the study of the iterative behavior of a family of functions that depend on a parameter. In this sense appears the study of what we now know as Julia Sets in the early twentieth century. Of course, at the time there were no computers and the study of these sets was very difficult. However, Julia sets play a critical role in the understanding of the dynamics of families of mappings (Aarts & Oversteegen, 1993).

In recent decades, as Hitt (2003) points out:

“Technological advancement has significantly influenced the development of theoretical notions that were previously taken into account but were not considered crucial in terms of explaining the learning of mathematical concepts. These theoretical aspects are the basis for understanding the study of the different representations of mathematical objects and their role in the construction of concepts. Now, with technology, it is important to study the different representations of mathematical objects in environments very different from those that were followed in the past” (p. 214).

The study of many areas of mathematics and mathematics education have been modified with the popularization of different technologies such as personal computers, including Fractal Geometry. So building multiple computer-generated images to form a richer mental image (“mental image” in the sense of Vinner, 1983) of fractals like Julia Sets is quite affordable for the students of our day.

One way to implement this construction to achieve a better understanding of Julia Sets (and other types of fractals) is by artistic means, motivating the learner in a purely playful process to build (in the sense of creating) images not only aesthetically pleasing but also endowed with some meaning through computer programs.

Eduardo Adam Navas-López, Master, Mathematics Department, University of El Salvador; research areas: computer graphics, mathematics education, algorithmic thinking, computer science education. E-mail: eduardo.navas@ues.edu.sv.

This pathway is not widely used in our time in mathematics in general, because although mathematics and art have been very close since the first manifestations of rationality of the human species, unfortunately, we have seen that these two areas of knowledge have distanced in school programs (Zaleski Filho, 2017). In his book, D'Amore (2015) makes a brilliant exposition of the presence of mathematics at the dawn of humanity.

The playful dimension is more common. Bishop (1998) discusses the role of games in mathematics education and notes:

“Educators in mathematics have discovered through their experience, and they have supported with theoretical research, that playing can be an integral part of learning. This has made the act of playing and the idea of gaming a much more widespread teaching and learning activity than it had been before” (p. 21).

For further reading, in the literature review made by González Peralta, Molina Zavaleta & Sánchez Aguilar (2014), one can find possible research lines about gaming in mathematics education.

There is also the computational dimension, which is becoming more common. For example, Hoffmann (2001) presents a sixth grade primary experience using a Monte Carlo simulation for introducing the concept of area of a unit circle (which is the approximation of the number π). There is also the Experience of DeJarnette (2016) in which students use the Scratch programming environment to help themselves understanding of how distances traveled by certain objects are functions of time.

So we have the seemingly strange conjunction of the four dimensions: mathematics, artistic, playful and computational. This conjunction is not directly welcomed by mathematicians, artists and art critics, or traditional computer scientists. In fact, pioneers in Computational Art faced rejection by the mathematical community, as Mumford, Series & Wright (2002) emphasizes:

“What to do with the pictures? [...] they were unpublishable in the standard way. There were no theorems, only very suggestive pictures. They furnished convincing evidence for many conjectures and lures to further exploration, but theorems were the coin of the realm and the conventions of that day dictated that journals only publish theorems” (p. viii).

A similar marginalization (and/or misunderstanding) occurs to Rani & Kumar (2004) whose article was published under Series D journal, “Research in Mathematical Education” and not under Series B journal, “The Pure and Applied Mathematics” of the same Organization¹, despite their article is not about education at all, but about Superior Mandelbrot Set. This paper is clearly about Pure Mathematics.

There are also problems with artistic acceptance according to Franke (1986):

“[Images] were considered [only] drawings from the plotter, the main problem was the uncertainty of the experts, the art historians and the critics, and above all the gallery owners. The problem was that the computer can produce an arbitrary number of equally good ‘originals’, which can be a detriment in the business world of art” (p. 186).

On the apparent incompatibility between science and art, and referencing the book of Peitgen and Richter (1986), Eilenberg (1986) offers a conciliation:

“It is rather unusual for natural [physical and mathematical] scientists to endeavor with such tenacity to bring their results and insights to the general public, [...] Instead of giving an abstract presentation in so many dry words, they have chosen pictures with a direct, universal appeal — a combination of mathematics and art!” (p. 175).

¹ <http://www.ksme.info/eng/>

That is, scientific art (whether mathematical art or computational art) can be used primarily to publicize results to the general public.

On the confusion between Computational Art and Standard Art, Franke (1986) proposes the following reflection:

“The art of every age has used the means of its time to give form to artistic innovation. [...] Why shouldn't the computer, that universal medium of information and communication which has even invaded our private homes [and our lives], be used as a medium and instrument of art?” (p. 187).

Moreover, Zaleski Filho (2017) tells us: “Thus, true art, which has no end in any of its external realizations, has as its identification a spiritual principle that enlivens and surpasses all of them”. So, after all, computational art, mathematical art and scientific art in general, as well as all other types of art, need no more justification than their ability to encourage human beings to rejoice in the artworks themselves.

This is how various collections and producers of Computer Generated Mathematical Art have proliferated, such as the Peitgen & Richter (1996) collection which includes many fractal graphics of complex dynamic systems, such as The Bridges Organization (2010) which annually holds an international exhibition and competition for mathematical art (not just computer-generated), such as Aslaksen's (2005) collection of university courses about art and mathematics, such as the computer-generated mathematical art collection presented by Navas-López (2015) which includes various types of fractals and other types of computer graphing techniques, etc. (explained in their catalog, Navas-López, 2016). There are also other less formal but not less impressive collections, such as Nylander's (n/d), The Context Free Art (n/d) community gallery, and Math Munch's (n/d) active blog which includes not only mathematical art but many interesting things. There are some collections that are very specific like Ross' (2007) mathematical analyses from the Sacks number spiral.

Speaking now of mathematical art in the classroom and in the curriculum in general, I support the reasoning of Figueiras, Molero, Salvador & Zuasti (2000):

“In the Mathematics that are taught, those that in Obligatory Education it is said that they will serve to acquire what is needed in life, Who can deny a place to beauty? Do we intend to let ourselves be carried away only by the dubious pragmatism of a mathematics cut short both in the time available for its teaching and in the potentiality of its values?” (p. 46).

Moreover, “The study of fractals is a motivating element in students, due to the implicit aesthetics in their constructions and the suggestive that their designs can be” (Figueiras et al., 2000, p. 47).

For Bosque, Segovia & Lupiáñez (2017):

“[...] Aesthetics transcends the Philosophy of Art finding a place in the Philosophy of Science. In this way, if aesthetics is a component of knowledge and also has a role in scientific and mathematical activity, it makes sense to study its influence on the teaching and learning of mathematics” (p. 20).

The proposed Fractal Geometry Activities in the Redondo & Haro (2004, 2005) High School Classroom are an excellent source of ideas for planning outreach activities to different types of fractals, including Julia Sets (see Redondo & Haro, 2005, p. 17). However, in this vast and general overview, the artistic approach and colors are scarce.

The presentation of the Mandelbrot and Julia Sets by Varona (2003) explains technical details about how to graph them in Mathematica(R), which is proprietary software, and proposes the inclusion of color palettes to

enhance the images although it can only display them in gray-scale due to the type of publication. This is an example that although there is acceptance of the subject, not all journals, editors or publishers are interested or prepared to accept “mathematics with colors”.

So, this paper then aims to expand the part of Redondo & Haro’s (2004, 2005) proposal on the Julia Sets topic, using computer programming as in the Varona’s (2003) paper, but with the following differences: (a) emphasizing the artistic approach and not the mathematical one, (b) not reducing to the aesthetic dimension but incorporating the communicative and didactic dimensions, (c) adding color, and (d) using free software, in this case the ContextFree software².

2. Theoretical Aspects

2.1 Julia Sets

We can take the definition of Julia Set from Bergweiler (1993):

“Let $f: \mathbb{C} \rightarrow \hat{\mathbb{C}}$ be a meromorphic function, where \mathbb{C} is de complex plane and $\hat{\mathbb{C}} = \mathbb{C} \cup \{ \infty \}$. [...] we shall always assume that f is neither constant nor a linear transformation. Denote by f^n the n th iterate of f , that is, $f^0(z) = z$ and $f^n(z) = f(f^{n-1}(z))$ for $n \geq 1$.

The basic objects studied in iteration theory are the Fatou set $F = F(f)$ and the Julia set $J = J(f)$ of a meromorphic function f . Roughly speaking, the Fatou set is the set where the iterative behavior is relatively tame in the sense that points close to each other behave similarly, while the Julia set is the set where chaotic phenomena take place. The formal definitions are:

$F = \{z \in \mathbb{C}: \{f^n: n \in \mathbb{N}\} \text{ is defined and normal in some neighborhood of } z\}$

and

$J = \hat{\mathbb{C}} - F$ ” (p. 153)

However, here will be chosen the most simplified version of Navas-López (2010, p. 263): “Julia set of a function $f_c(z)$ with seed $c \in \mathbb{C}$, denoted by $J_c(f)$, is the set composed by all $z \in \mathbb{C}$, such that the sequence $\{z_k\}_{k=0}^{\infty}$ is bounded, where $z_0 = z$ and $z_{n+1} = f_c(z_n)$ [for $n \geq 0$]. Where $J_c(f)$, in its simplest form, uses $f_c(z) = z^2 + c$.”

However, many other different functions can be used that provide interesting results such as those presented by Entwistle (1989), Garijo, Jarque & Villadelprat (2017), Liu, Zhu, Wang & Zhu (2001), Peitgen & Richter (1996), Pickover & Khorasani (1985), Rochon (2000) and Rani & Kumar (2004).

The criterion used to determine whether the sequences diverge is if z_k , for some $k \geq 0$, has a modulus greater than 2, that is $|z_k| \geq 2$. Since it cannot be evaluated to infinity, a bound is used: N . If the sequence does not “diverge” before reaching the N th term (when $k \leq N$), it is considered not divergent, that is, it is bounded. The larger N , the greater the precision of the set (Navas-López, 2010, p. 263).

2.2 Implementation

The CFDG language, version 3, of Context Free³ software and adapted computer graphics techniques from

² www.contextfreeart.org

³ <https://www.contextfreeart.org/>

Baker (1995) and some from Foley et al (1996) (from their respective chapters on fractal graphing) are used for implementation.

The CFDG language is not a programming language properly. It is actually a language in which can be defined a particular type of context-free grammars whose terminal symbols are primitive figures: squares, circles and triangles. Different related transformations (displacement, scaling, rotation, etc.) can be applied to these figures. However, CFDG language supports Functional Programming when it is needed, specially for numerical algorithms. For further reading, visit the Context Free Art (n/d) documentation page.

The basic geometric object to use in the examples is the square. This is constructed using the following primitive:

```
SQUARE [
x<x_offset>
y<y_offset>
size<width><height> # Can be abbreviated as s
hue<tone>           # Can be abbreviated as h
saturation<saturation> # Can be abbreviated as sat
brightness<bright>   # Can be abbreviated as b
]
```

Where $\langle x_offset \rangle$ and $\langle y_offset \rangle$ indicate the displacement from the origin, $\langle width \rangle$ and $\langle height \rangle$ determine the size of the figure, $\langle tone \rangle$ is an angle between 0° and 360° indicating the color of the figure according to the HSV color model, and $\langle saturation \rangle$ and $\langle bright \rangle$ indicate the corresponding. For more information, see HSV Color Model (2021).

The function required to determine the convergence of a point (z_r, z_i) in the complex plane is:

```
MAXSTEPS = 40
steps(numSteps, z_r,z_i, c_r,c_i) =
    if((numSteps < MAXSTEPS) && (z_r*z_r+z_i*z_i<4),
        steps(numSteps+1,
            z_r*z_r - z_i*z_i + c_r, 2*z_r*z_i + c_i, c_r, c_i),
        numSteps)
```

where MAXSTEPS is the value of N in the criterion described in section 2.1. The initial call must be in the form `steps (0, z_r, z_i, c_r, c_i)`, where (c_r, c_i) is the seed value. This call returns the number of iterations executed from which the sequence diverges, or returns MAXSTEPS if the sequence has not yet diverged at the N th iteration.

Then a figure must be implemented, determined by the seed, with the following structure:

```
LIMIT = 1000 # Image resolution

# Borders of the complex plane to show:
LIMLEFT = -1.5
LIMRIGHT = 1.5
LIMTOP = 1.5
LIMBOT = -1,5
```

```

# Width and height of the boxes that will discretize the image:
SIZEX = (LIMRIGHT-LIMLEFT)/(LIMIT-1)
SIZEY = (LIMTOP-LIMBOT)/(LIMIT-1)

shape julia(c_r,c_i) {
    loop i = (LIMIT) [] {
        z_i = (LIMTOP-LIMBOT)*i/(LIMIT-1) + LIMBOT # y
    loop j = LIMIT [] {
        z_r = (LIMRIGHT-LIMLEFT)*j/(LIMIT-1) + LIMLEFT # x

        numSteps = steps(0, z_r, z_i, c_r, c_i)
        if (numSteps==MAXSTEPS){
            # Black
            SQUARE[x z_r y z_i size SIZEX SIZEY b 0]
        } else {
            # White
            SQUARE[x z_r y z_i size SIZEX SIZEY b 1]
        }
    }
}
}

```

The startshape directive is used to indicate which is the generating figure:
startshape julia(-0.381966, 0.618034)

3. Artworks Raised as Examples

It is recommended to take these proposals and experiment with the parameters (seed, drawn interval, colors, etc.).

3.1 Frozen Fjords

This artwork (Figure 1) shows an aerial view of snow-capped fjords, its thin dark sand shores and the deeply blue sea. The motivation is that fjords have a natural fractal shape.

From a technical point of view, it is a view of a Julia set in the range of $[0.01,0.09] \times [0.02i, 0.10i]$, with seed $c = -1.384286 + 0.004286i$.

The coloration of this artwork has a constant hue as well as saturation, but the brightness is variable depending on the number of steps in which it is determined that the point belongs or does not belong to the set.

Code:

```

# This file is Free Software released under the GNU GPLv3 license or
# its latest version:
# http://www.gnu.org/licenses/gpl.html

# To generate the image run the following line:
# $ cfdg -b 0 -s 1000 fjords.cfdg fjords.png

```

```
startshape fjords(-1.384286,0.004286)
```

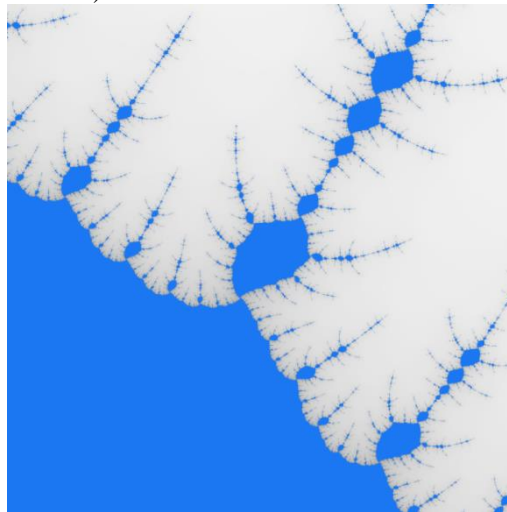


Figure 1 Frozen Fjords

```
LIMIT = 1000 # Image resolution
```

```
MAXSTEPS = 300
```

```
# Borders of the complex plane to show:
```

```
LIMLEFT = 0.01
```

```
LIMRIGHT = 0.09
```

```
LIMTOP = 0.10
```

```
LIMBOT = 0.02
```

```
# Width and height of the boxes that will discretize the image:
```

```
SIZEX = (LIMRIGHT-LIMLEFT)/(LIMIT-1)
```

```
SIZEY = (LIMTOP-LIMBOT)/(LIMIT-1)
```

```
steps(numSteps,z_r,z_i,c_r,c_i) =
```

```
  if((numSteps < MAXSTEPS) && (z_r*z_r+z_i*z_i<4),
```

```
    steps(numSteps+1,
```

```
    z_r*z_r - z_i*z_i + c_r, 2*z_r*z_i + c_i, c_r, c_i),
```

```
    numSteps)
```

```
shape fjords(c_r,c_i) {
```

```
  FILL[h 214 sat 0.89 b 0.95] # Blue ocean
```

```
  loop i = (LIMIT) [] {
```

```
    z_i = (LIMTOP-LIMBOT)*i/(LIMIT-1) + LIMBOT # y
```

```
    loop j = LIMIT [] {
```

```
      z_r = (LIMRIGHT-LIMLEFT)*j/(LIMIT-1) + LIMLEFT # x
```

```

numSteps = steps(0, z_r, z_i, c_r, c_i)
if(numSteps<MAXSTEPS){
    SQUARE[x z_r
        y z_i
        size SIZEX SIZEY
        h 30 sat 0
        b (1+(1-numSteps)/(MAXSTEPS-1))]
    }
}
}
}

```

3.2 The Wail of the Pripiat Forest

This artwork (Figure 2) shows a sick forest around the city of Pripiat. This ghost town is known for being affected by the worst accident in nuclear power history on April 26, 1986, when the Chernobyl Nuclear Power Plant reactor number 4 was overheated and exploded during a shutdown test. The motivation came after seeing a detailed documentary about the nuclear disaster at the Chernobyl nuclear power plant.



Figure 2 The Wail of the Pripiat Forest

From a technical point of view it is a view of a Julia set in the range $[-0.052857, 0.188571] \times [-0.105714i, 0.135714i]$ with seed $c = -0.381966 + 0.618034i$.

The coloration of this artwork has a random variable hue between 60 and 74, a variable random saturation between 0.41 and 0.66, and a variable random maximum brightness between 0.32 and 0.35 for points that do not belong to the set.

Code:

```

# This file is Free Software released under the GNU GPLv3 license or
# its latest version:

```



```

# http://www.gnu.org/licenses/gpl.html

# To generate the image run the following line:
# $ cfdg -b 0 -s 1000 -v PAJBHA forest.cfdg forest.png

LIMIT = 1000 # Image resolution
MAXSTEPS = 200
MINMAXBRIGHT = 0.32
MAXMAXBRIGHT = 0.68

startshape forest

# Borders of the complex plane to show:
LIMLEFT = -0.052857
LIMBOT = -0.105714
LIMRIGHT = 0.188571
LIMTOP = 0.135714

# Width and height of the boxes that will discretize the image:
SIZEX = (LIMRIGHT-LIMLEFT)/(LIMIT-1)
SIZEY = (LIMTOP-LIMBOT)/(LIMIT-1)

steps(numSteps,z_r,z_i,c_r,c_i) =
  if((numSteps < MAXSTEPS) && (z_r*z_r+z_i*z_i<4),
    steps(numSteps+1,
      z_r*z_r - z_i*z_i + c_r, 2*z_r*z_i + c_i, c_r, c_i),
    numSteps)

NUMBLOCKS = 8 # The number of columns and rows

shape forest{
  loop i = NUMBLOCKS []{
    loop j = NUMBLOCKS [] {
      julia2(
        -0.381966, 0.618034,
        j*(LIMRIGHT-LIMLEFT)/NUMBLOCKS+LIMLEFT,
        i*(LIMTOP-LIMBOT)/NUMBLOCKS+LIMBOT,
        rand(MINMAXBRIGHT,MAXMAXBRIGHT)
      ) [h rand(60,74)
        sat rand(0.41,0.66)
        b rand(0.32,0.35)]
    }
  }
}

```

```

    }
  }
}

shape julia2(c_r,c_i, xi,yi, maxBright) {
  xf = xi+(LIMRIGHT-LIMLEFT)/NUMBLOCKS
  yf = yi+(LIMTOP-LIMBOT)/NUMBLOCKS
  loop i = LIMIT/NUMBLOCKS [] {
    z_i = (yf-yi)*i/(LIMIT/NUMBLOCKS-1) + yi
    loop j = LIMIT/NUMBLOCKS [] {
      z_r = (xf-xi)*j/(LIMIT/NUMBLOCKS-1) + xi

      numSteps = steps(0, z_r, z_i, c_r, c_i)
      SQUARE[x z_r y z_i size SIZEX SIZEY
        b ( maxBright+maxBright*(1-numSteps)/(MAXSTEPS-1) )]
    }
  }
}

```

3.3 Ragnarök

This artwork (Figure 3) shows a rough, sharp and nordic scenario illustrating the cataclysm of the Ragnarök, which is the Viking apocalypse. The motivation comes from the story of the violent Viking apocalypse.

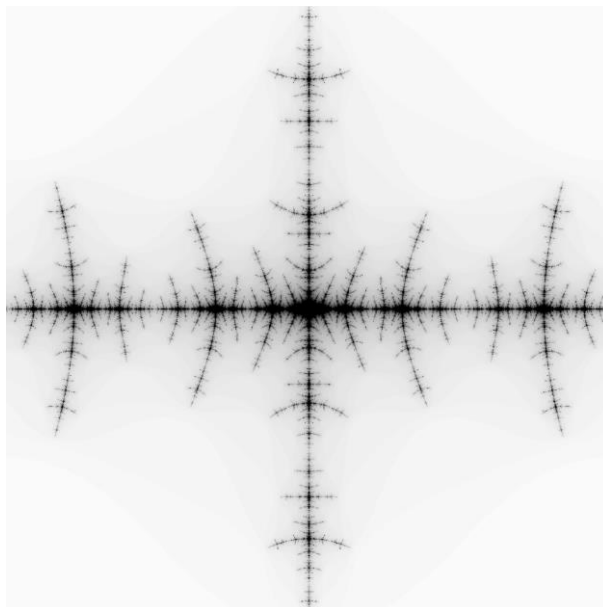


Figure 3 Ragnarök

From a technical point of view, it is a view of a Julia set in the range of $[-0.6, 0.6] \times [-0.6i, 0.6i]$, with seed $c = -1.4 + 0.0i$.

The coloration of this artwork has a higher variable white brightness the faster it is determined that the dots do not belong to the set. The points that do belong to the set are black.

Code:

```
# This file is Free Software released under the GNU GPLv3 license or
# its latest version:
# http://www.gnu.org/licenses/gpl.html
```

```
# To generate the image run the following line:
# $ cfdg -b 0 -s 1000 ragnarok.cfdg ragnarok.png
```

```
# Ragnarok seed possibilities:
```

```
# -1.384286,0.004286
# -1.3, 0.00525
# -1.3, 0.0
# -1.5, 0.0
# -1.4, 0.0
```

```
startshape ragn(-1.4, 0.0)
```

```
LIMIT = 1000 # Image resolution
MAXSTEPS = 100
```

```
# Borders of the complex plane to show:
```

```
LIMLEFT = -0.6
LIMRIGHT = 0.6
LIMTOP = 0.6
LIMBOT = -0.6
```

```
# Width and height of the boxes that will discretize the image:
```

```
SIZEX = (LIMRIGHT-LIMLEFT)/(LIMIT-1)
SIZEY = (LIMTOP-LIMBOT)/(LIMIT-1)
```

```
steps(numSteps,z_r,z_i,c_r,c_i) =
  if((numSteps < MAXSTEPS) && (z_r*z_r+z_i*z_i<4),
    steps(numSteps+1,
      z_r*z_r - z_i*z_i + c_r, 2*z_r*z_i + c_i, c_r, c_i),
    numSteps)
```

```
shape ragn(c_r,c_i) {
```

```

loop i = LIMIT/2 [] {
  z_i = (LIMTOP-LIMBOT)*i/(LIMIT-1) + LIMBOT # y
  loop j = LIMIT/2 [] {
    z_r = (LIMRIGHT-LIMLEFT)*j/(LIMIT-1) + LIMLEFT # x

    numSteps = steps(0, z_r, z_i, c_r, c_i)
    bright = (1+(1-numSteps)/(MAXSTEPS-1))
    # Symmetry when imag part of seed is zero:
    SQUARE[x z_r    y z_i    size SIZEX SIZEY b bright]
    SQUARE[x (-z_r) y z_i    size SIZEX SIZEY b bright]
    SQUARE[x z_r    y (-z_i) size SIZEX SIZEY b bright]
    SQUARE[x (-z_r) y (-z_i) size SIZEX SIZEY b bright]
  }
}
}

```

3.4 The Battle for Smolensk

Here (Figure 4) what I want to represent is a blood bath on the icy ground of Smolensk at the end of 1941, early winter. The motivation comes from a series of documentaries about World War II, particularly about Operation Barbarossa and how the facts of the German advance on Soviet ground developed in the period 1941-1943.

From a technical point of view it is a view of a Julia set in the range of $[-0.21, 0.63] \times [-0.865714i, -0.025714i]$, with center $0.21 - 0.445714i$ and seed $c = 0.39 - 0.252857i$.

The coloration of this artwork are red dots and maximum brightness with lower saturation for the points the faster it is determined that they do not belong to the set, and the higher the longer it takes to determine not belonging. The dots that do belong to the set are painted as intense red.

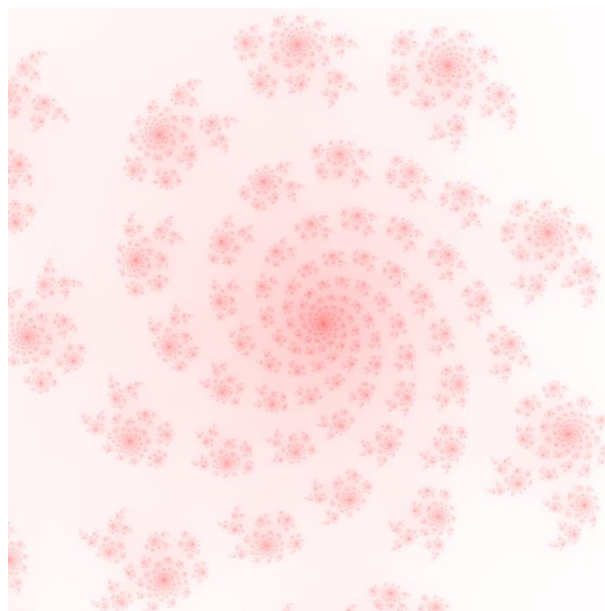


Figure 4 The Battle for Smolensk

Code:

```
# This file is Free Software released under the GNU GPLv3 license or  
# its latest version:
```

```
# http://www.gnu.org/licenses/gpl.html
```

```
# To generate the image run the following line:
```

```
# $ cfdg -b 0 -s 1000 batalla.cfdg batalla.png
```

```
LIMIT = 1000 # Image resolution
```

```
MAXSTEPS = 400
```

```
startshape julia3(0.39, -0.252857)
```

```
SIDE = 0.84 # side of the viewport square
```

```
CX = 0.21 # center x
```

```
CY = -0.445714 # center y
```

```
# Borders of the complex plane to show:
```

```
LIMLEFT = CX - SIDE/2
```

```
LIMBOT = CY - SIDE/2
```

```
LIMRIGHT = CX + SIDE/2
```

```
LIMTOP = CY + SIDE/2
```

```
# Width and height of the boxes that will discretize the image:
```

```
SIZEX = (LIMRIGHT-LIMLEFT)/(LIMIT-1)
```

```
SIZEY = (LIMTOP-LIMBOT)/(LIMIT-1)
```

```
steps(numSteps,z_r,z_i,c_r,c_i) =
```

```
  if((numSteps < MAXSTEPS) && (z_r*z_r+z_i*z_i<4),
```

```
    steps(numSteps+1,
```

```
      z_r*z_r - z_i*z_i + c_r, 2*z_r*z_i + c_i, c_r, c_i),
```

```
    numSteps)
```

```
shape julia3(c_r,c_i) {
```

```
  loop i = LIMIT [] {
```

```
    z_i = (LIMTOP-LIMBOT)*i/(LIMIT-1) + LIMBOT # y
```

```
    loop j = LIMIT [] {
```

```
      z_r = (LIMRIGHT-LIMLEFT)*j/(LIMIT-1) + LIMLEFT # x
```

```

numSteps = steps(0, z_r, z_i, c_r, c_i)
SQUARE[x z_r y z_i size SIZEX SIZEY b 1
      sat ((numSteps-1)/(MAXSTEPS-1))]
    }
  }
}

```

6. Reflections and Discussion

As González, Molina & Sánchez (2014) say, there is certainly potential in the inclusion of playful activities in teaching but precautions must be taken to make the sessions useful for the purposes of the curriculum. So this type of activities should be done mainly in extracurricular spaces, since students have different levels of aptitude and artistic sensitivity.

Artistic activities in general are very enriched thanks to the computer offering us the possibility of experimentation, since one can check the influence of parameters on the results, one can check the result of the transformations, the limiting values of interactively applied calculations, etc. (Franke, 1986).

“Modern art studies have shown, however, that meeting the classical definition of beauty is not in itself sufficient to create a work of art. In addition there must be something to stimulate interest, demand involvement, and motivate further thoughts” (Franke, 1986, p. 184). So it is not enough to “create” complicated fractal images that are aesthetically beautiful, but they must have a more transcendent meaning.

It should be noted that the fine structures of these images are manifestations of the fact that the smallest variations (in this case the value of the seed) at the beginning of a procedure can result in huge differences later (the different Julia sets are very different from each other), and as Eilenberg (1986) tells us, the research of dynamic systems indicates that this is typical of natural processes.

Finally we can echo McCabe and Reisz’s (2015) documentary and say that we can extract art from mathematics since “Mathematics is an inherent part of nature”.

References

- Aarts J. M. and Oversteegen L. G. (1993). “The geometry of Julia sets”, *Transactions of the American Mathematical Society*, Vol. 338, No. 2, pp. 897–918.
- Aslaksen H. (2005). “A contrarian view of teaching mathematics and art”, in: *Proceedings of the Special Year in Art & Mathematics, A+M=X International Conference*, Univ. of Colorado.
- Baker P. and Hearn D. (1995). *Gráficas por Computadora*, Prentice Hall.
- Bergweiler W. (1993). “Iteration of meromorphic functions”, *Bulletin of the American Mathematical Society*, Vol. 29, No. 2, pp. 151–188.
- Bishop A. (1998). “El papel de los juegos en la educación matemática”, *Uno. Revista de Didáctica de las Matemáticas*, Vol. 18, pp. 9–19.
- Bosque B., Segovia I. and Lupiáñez J. L. (2017). “Exploración del papel de la estética en la enseñanza y el aprendizaje de las Matemáticas”, *PNA*, Vol. 12, No. 1, pp. 1–25.
- Context Free Art (n/d). “Art gallery using CFDG language”, available online at: <https://contextfreeart.org/gallery/>.
- D’Amore B. (2015). *Arte e Matematica: Metafore, Analogie, Rappresentazioni, Identità Tra Due Mondi Possibili*, Edizioni Dedalo.
- DeJarnette A. F. (2016). “Students’ use of a computer programming environment to represent distance as a function of time”, in: Wood M. B., Turner E. E., Civil M. and Eli J. A. (Eds.), *Proceedings of the 38th Annual Meeting of the North American Chapter of the International Group for the Psychology of Mathematics Education (PMENA 38)*, Tucson, AZ: The University of Arizona.
- Eilenberg G. (1986). “Freedom, science, and aesthetics”, in: Peitgen H. O. & Richter P. H., *The Beauty of Fractals: Images of*

- Complex Dynamical Systems*, Springer-Verlag: Berlin, Heidelberg, pp. 175–180.
- Entwistle I. D. (1989). “Julia set art and fractals in the complex plane”, *Computers & Graphics*, Vol. 13, No. 3, pp. 389–392, doi: 10.1016/0097-8493(89)90090-3.
- Figueiras L., Molero M., Salvador A. and Zuasti N. (2000). “Una propuesta metodológica para la enseñanza de la Geometría a través de los fractales”, *Suma*, Vol. 35, pp. 45–54.
- Franke H. W. (1986). “Refractions of science into art”, in: Peitgen H. O., & Richter P. H. *The Beauty of Fractals: Images of Complex Dynamical Systems*, Springer-Verlag: Berlin, Heidelberg, pp. 181–188.
- Foley J. D., Van Dam A., Feiner S. K., Hughes J. F. and Phillips R. L. (1996). *Introducción a la Graficación por Computador*, Addison-Wesley.
- Garijo A., Jarque X. and Villadelprat J. (2017). “An effective algorithm to compute Mandelbrot sets in parameter planes”, *Numerical Algorithms*, Vol. 76, No. 2, pp. 555–571, doi: 10.1007/s11075-017-0270-8.
- González Peralta A. G., Molina Zavaleta J. G., and Sánchez Aguilar M. (2014). “La matemática nunca deja de ser un juego: investigaciones sobre los efectos del uso de juegos en la enseñanza de las matemáticas”, *Educación Matemática*, Vol. 26, No. 3, pp. 109–133.
- Hitt F. (2003). “Una reflexión sobre la construcción de conceptos matemáticos en ambientes con tecnología”, *Edición Especial: Educación Matemática*, Vol. 213.
- Hoffmann R. (2001). “Computer simulations in mathematics education”, in: van den Heuvel-Panhuizen M. (Ed.), *Proceedings of the 25th Conference of the International Group for the Psychology of Mathematics Education (PME 25)*, Utrecht, The Netherlands. International Group for the Psychology of Mathematics Education.
- Liu X., Zhu Z., Wang G. and Zhu W. (2001). “Composed accelerated escape time algorithm to construct the general Mandelbrot sets”, *Fractals*, Vol. 9, No. 2, pp. 149–153, doi: 10.1142/S0218348X01000580.
- Math Munch (n/d). Available online at: <https://mathmunch.org/>.
- McCabe D., Reisz R. (producers) and McCabe D., and Reisz R. (directors). (2015). *The Great Math Mystery*, available online at: <http://www.imdb.com/title/tt4254160/>.
- HSV Color Model (2021). “Wikipedia, La enciclopedia libre”, May 30, 2021, available online at: https://es.wikipedia.org/w/index.php?title=Modelo_de_color_HSV&oldid=135310952.
- Mumford D., Series C. and Wright D. (2002). *Indra’s Pearls: The Vision of Felix Klein*, Cambridge University Press.
- Navas-López E. A. (2010). *Una Humilde Introducción a la Graficación por Computadora y Otras Yerbas*. Universidad Centroamericana “José Simeón Cañas”: El Salvador, available online at: <https://drive.google.com/file/d/1PwAy60amUzrEXRahY7VGS9wa2P2Beqv1/view>.
- Navas-López E. A. (2015). *Arte Matemático 2015*, Image Album, available online at: <http://flic.kr/s/aHsk8rHNRi>
- Navas-López E. A. (2016). *Catálogo de Arte Matemático Generado por Computadora, Colección 2014–2015*, doi: 10.13140/RG.2.2.34092.90243.
- Nylander P. (n/d). “Math artwork”, available online at: <http://bugman123.com/Math>.
- Peitgen H. O. and Richter P. H. (1986). *The Beauty of Fractals: Images of Complex Dynamical Systems*, Springer-Verlag: Berlin, Heidelberg.
- Pickover C. A. and Khorasani E. (1985). “Computer graphics generated from the iteration of algebraic transformations in the complex plane”, *Computers & graphics*, Vol. 9, No. 2, pp. 147–151, doi: 10.1016/0097-8493(85)90022-6.
- Rani M. and Kumar V. (2004). “Superior mandelbrot set”, *Research in Mathematical Education*, Vol. 8, No. 4, pp. 279–291.
- Redondo A. and Haro M. (2004). “Actividades de geometría fractal en el aula de secundaria (I)”, *Suma*, Vol. 47, pp. 19–28.
- Redondo A. and Haro M. (2005). “Actividades de geometría fractal en el aula de secundaria (II)”, *Suma*, Vol. 48, pp.15–21.
- Rochon D. (2000). “A generalized Mandelbrot set for bicomplex numbers”, *Fractals*, Vol. 8, No. 4, pp. 355-368. DOI: 10.1142/S0218348X0000041X.
- Ross M. M. (2007). “The sacks number spiral”, available online at: <http://www.naturalnumbers.org/sparticle.html>.
- The Bridges Organization. (2010). “Mathematical art galleries”, available online at: <http://gallery.bridgesmathart.org/>.
- Varona J. L. (2003). “Representación gráfica de fractales mediante un programa de cálculo simbólico”, *La Gaceta de la RSME*, Vol. 6, pp. 213–230.
- Vinner S. (1983). “Concept definition, concept image and the notion of function”, *International Journal of Mathematical Education in Science and Technology*, Vol. 14, No.3, pp. 293–305, doi: 10.1080/0020739830140305.
- Zaleski Filho D. (2017). *Matemática e Arte*. Coleção Tendências em Educação Matemática (1st ed.), Autêntica: Belo Horizonte.