Saturn Guide

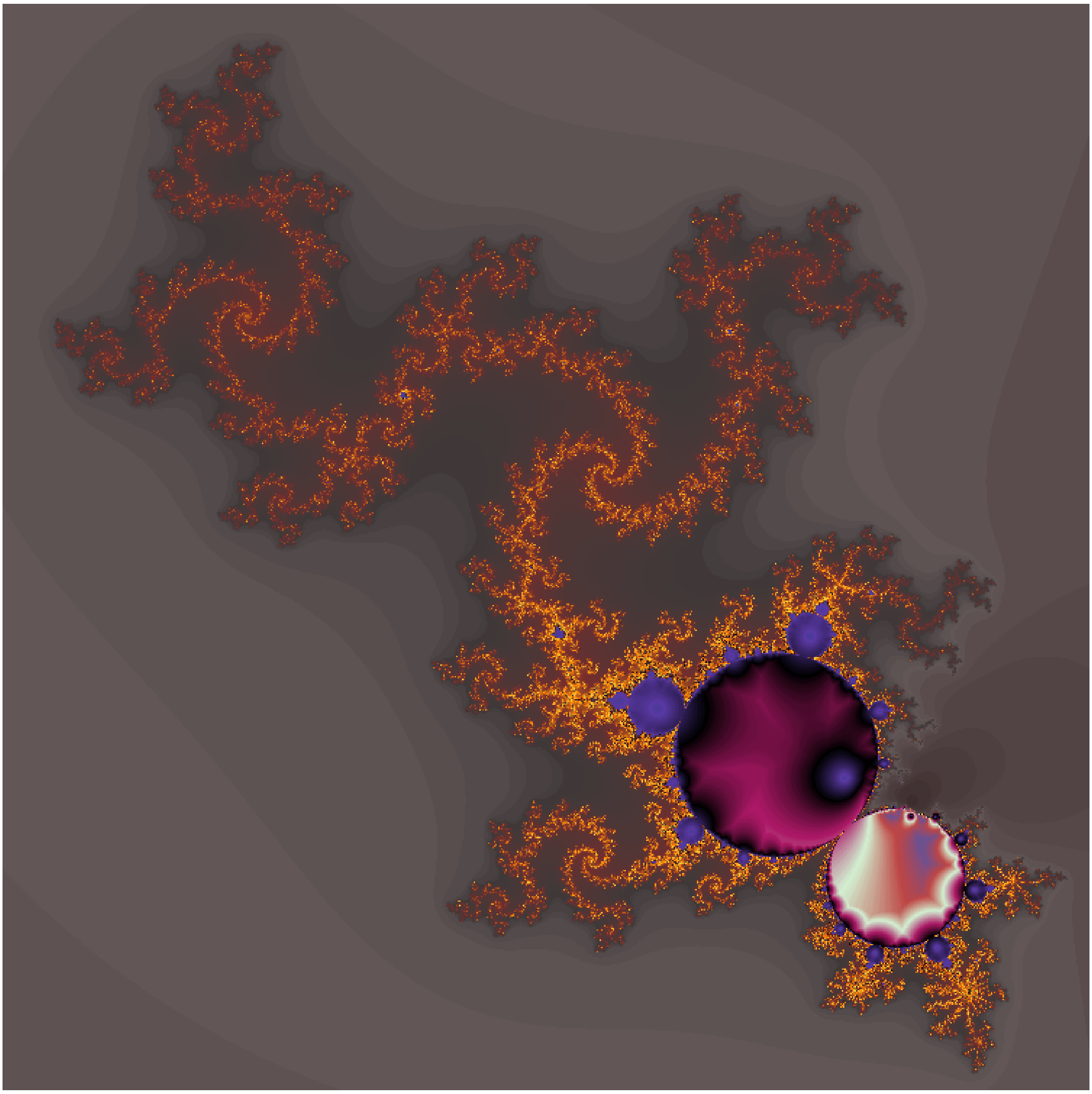


Table of Contents

1 Introduction

Not all fractal types available in Saturn will produce interesting pictures without some guidance. It can be very frustrating trying a fractal type and being presented with a black rectangle with no idea how to alter the parameters to get anything at at all.

The purpose of this guide is to give some starting points for exploring fractal types where some guidance is necessary.

In additional some well known fractal types are available but only via a fractal of a different name. The reason for this is that the escape time fractals can be modified by the use of transforms, so fractals such as the “Burning Ship” are produced by applying a transform to the Mandelbrot set.

There are example images in this document for which there are parameter files available for download at http://element90.wordpress/software/downloads.

The quality of the images shown in the PDF are degraded and are used to give an impression of the images that can be generated by Saturn.

The parameter files are provided to help you in your exploration of the fractal types implemented in Saturn. Please DO NOT simply change the colouring and pass off the images as your own.

2 Definitions

parameter file	a .spf (Saturn Parameter File) which contains all necessary parameters to generate a fractal image in XML. These files can be opened by either using Saturn's File menu or by dropping them on the area where the fractal image is drawn. Parameter files can only be open by Saturn version 2.0.x and later versions, to expand an image using Titan then Saturn can be used to save the seed file required by Titan.
z	the iterating complex variable, usually initialised to zero except when the initial iteration will result in either infinity, not a number or zero. It can also be initialised to a non-zero fixed complex number constant or to c (either transformed or untransformed). z_0 is the initial value of z.
c	this value corresponds to the location in the complex plane it is commonly used in the Mandelbrot type escape time algorithm formulae. It is also used to initialise z for the Julia type escape time algorithm.
i	The imaginary unit number such that i^2 is equal to -1.
suffix .r	the real part of a complex variable e.g. z.r is the real part of z
suffix .i	the imaginary part of a complex variable e.g. z.i is the imaginary part of z.
subscript r	the real part of a complex variable, this may used the formula section of the fractal tab (see fractal settings).
subscript i	the imaginary part of a complex variable this may used the formula section of the fractal tab (see fractal settings).
Greek letters	The Greek lower case letters α, β,γ, δ, ε, ζ represent the variables used as parameters in many fractal formulae shown in the formula section of the fractal tab, they are also used with .r and .i suffixes in the parameters section of the fractal tab.

3 Fractal Guide

This is the first edition of this guide, the aim is to eventually cover every fractal type implemented in Saturn. For this edition there will be general advice and a selection of fractal types will be covered.

The general advice covers the escape time fractals where some guidance is required, for Pickover Popcorn and Lyapunov fractals guidance isn't necessary.

At some point the user will be presented with a single colour rectangle. First of all check the inner colouring method, if it is set to “Fixed Colour” and the colour is the same as the rectangle then all the points in the image are being selected as inner. Some fractal types will only produce inner points and others will also only produce inner points when transforms are applied. Set the inner colour method to something other than “Fixed Colour”, a good choice is “Magnitude” with the “Exponential Sum” option as this colour method reveals inner structures.

If the single colour does not match inner colouring of “Fixed Colour” then there are two possibilities:

1. If calculation has completed quickly then every point in the image has met the “Bailout Condition”. Try increasing the limit, adjusting parameters (if any) or changing the initial value of z.
2. If calculation is very slow then the calculation of each iteration is resulting in “not a number” usually caused by division by zero. Try changing the initial value of z.

Some transformations can change the image in such way that its dimensions are greatly increased so you may see a portion of the image that has little or no detail. Try zooming out using either right clicking on the image or increasing the “Width” in the “Position Tab”.

If at any point you feel that you've lost your way in the fractal use the revert option in the edit menu. Revert will reset the display area to the starting display area, it will not change colouring methods, colour method selection or transforms.

As Saturn stores the current state of all of the fractals when it is closed and reloads the state of all the fractals when it is started you may get an unexpect image of a given fractal type and as it may be some time since it was last visited you may have forgotten how you left it. Use revert and rest colouring and transforms.

You can find posts on the Element 90 Fractals blog http://element90.wordpress.com on Saturn fractals, any fractals covered on the blog that aren't currently featured in this document will be incorporated into future editions.

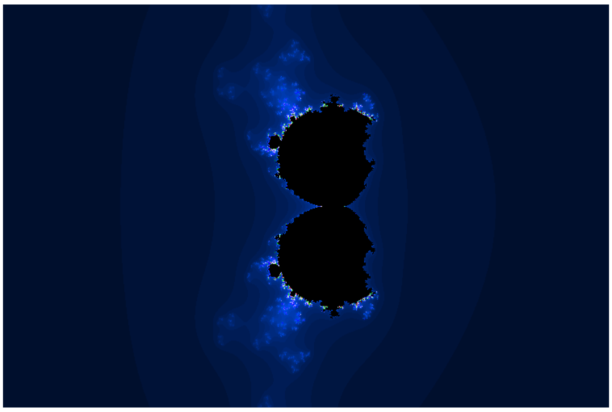
3.1 Escape Time Fractals

3.1.1 Almost Cubic & Cubic

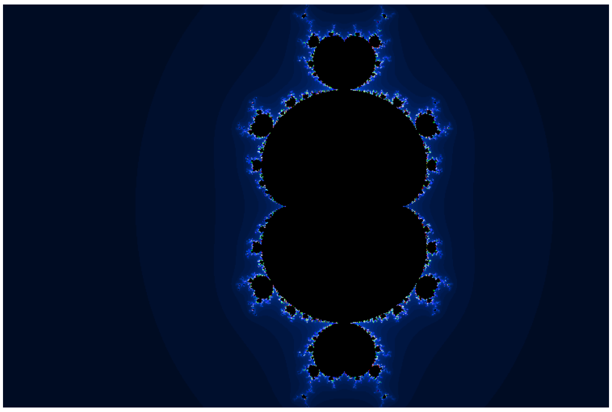
These two fractal types are essentially the same except that Almost Cubic is the result of an error in implementing the formula using pairs of real numbers instead of using complex numbers.

The formula for both these fractals is:

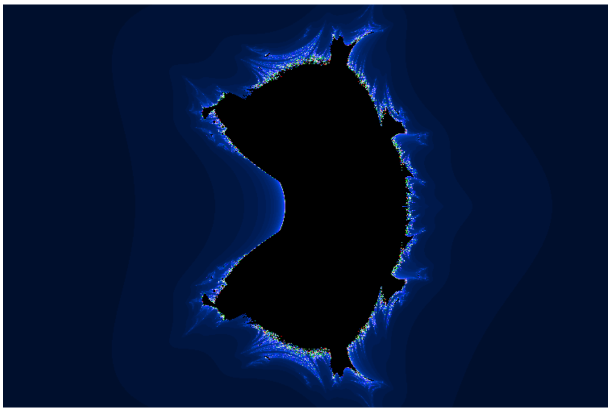
$$z \leftarrow \alpha z^3 + \beta z^2 + \gamma z + c$$



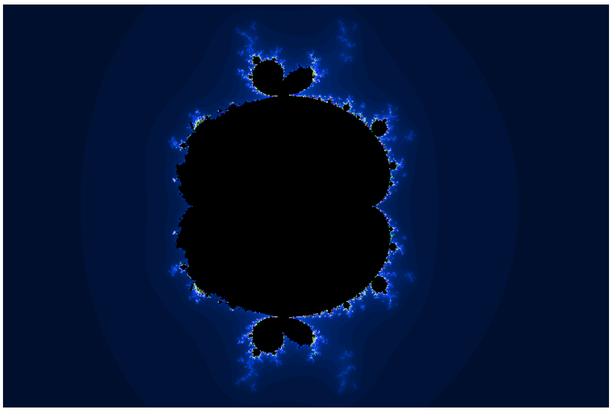
$\alpha = 1$, $\beta = 1$, $\gamma = 1$ the image is the same for both fractals.



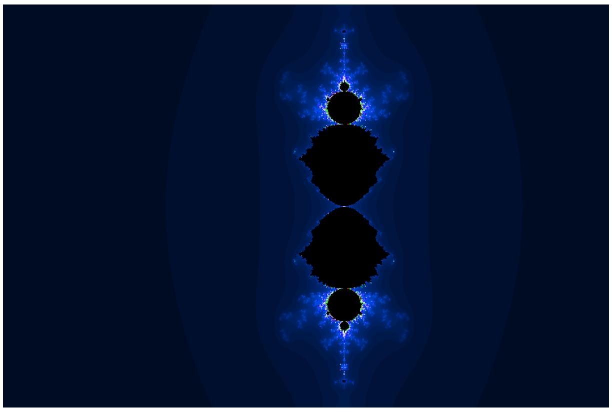
$\alpha = 1$, $\beta = 0$, $\gamma = 0$ Cubic



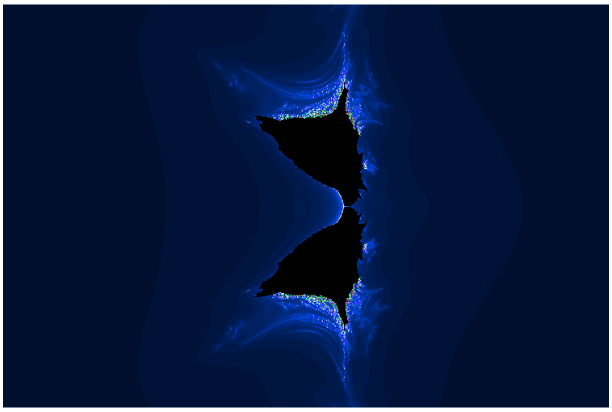
$\alpha = 1$, $\beta = 0$, $\gamma = 0$ Almost Cubic



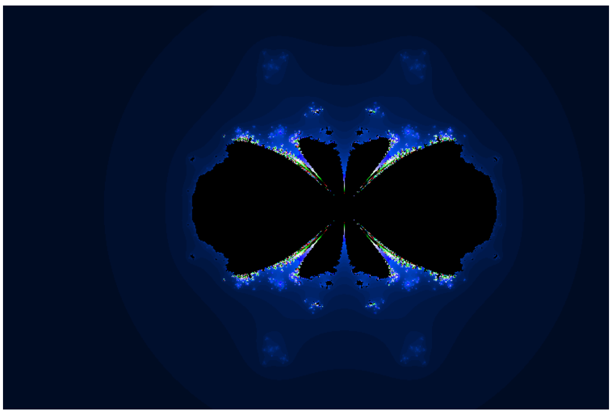
$\alpha = 1$, $\beta = 0$, $\gamma = 0$ this image is the same for both fractal types.



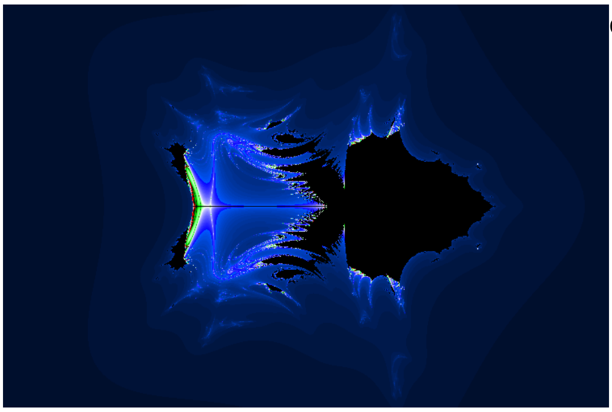
$\alpha = 1$, $\beta = 0$, $\gamma = 1$ Cubic



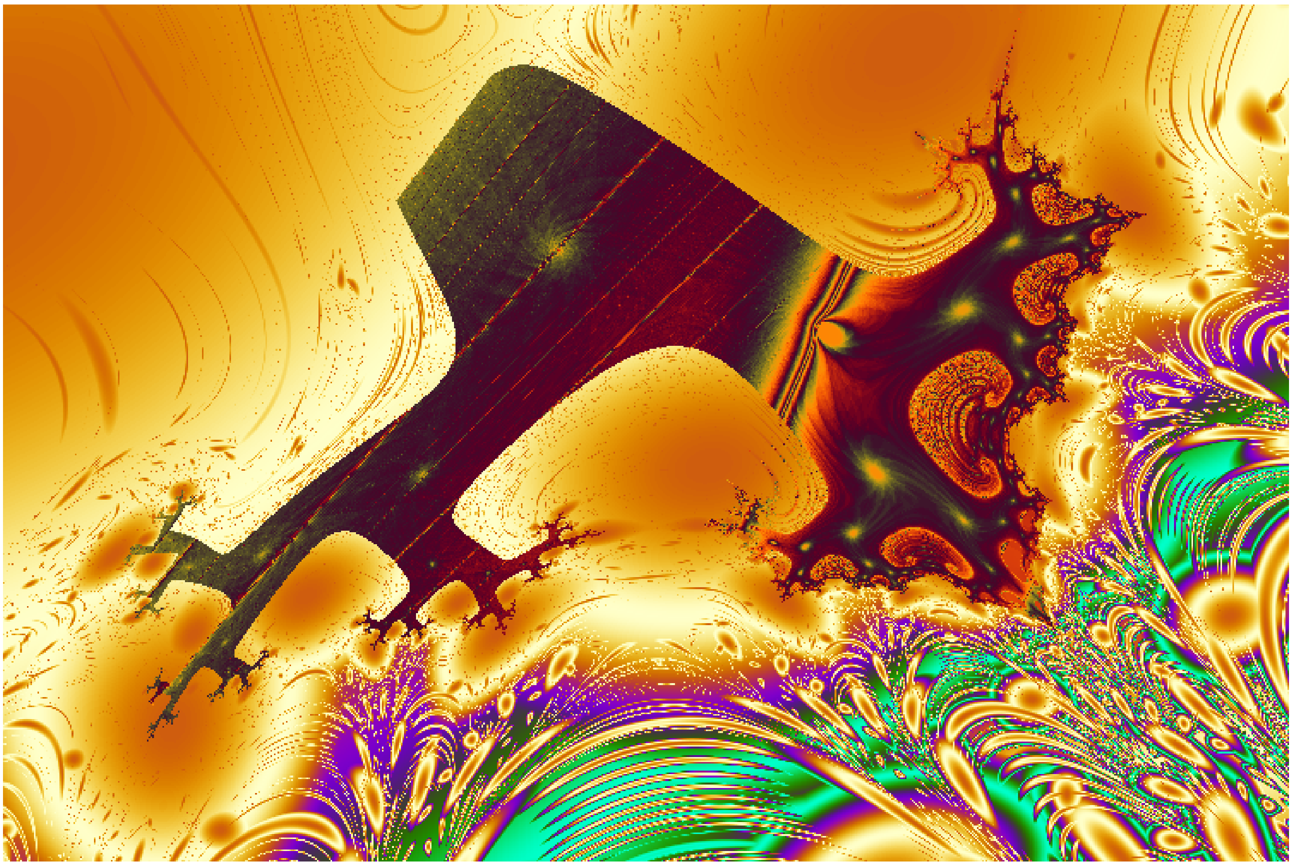
$\alpha = 1$, $\beta = 0$, $\gamma = 1$ Almost Cubic

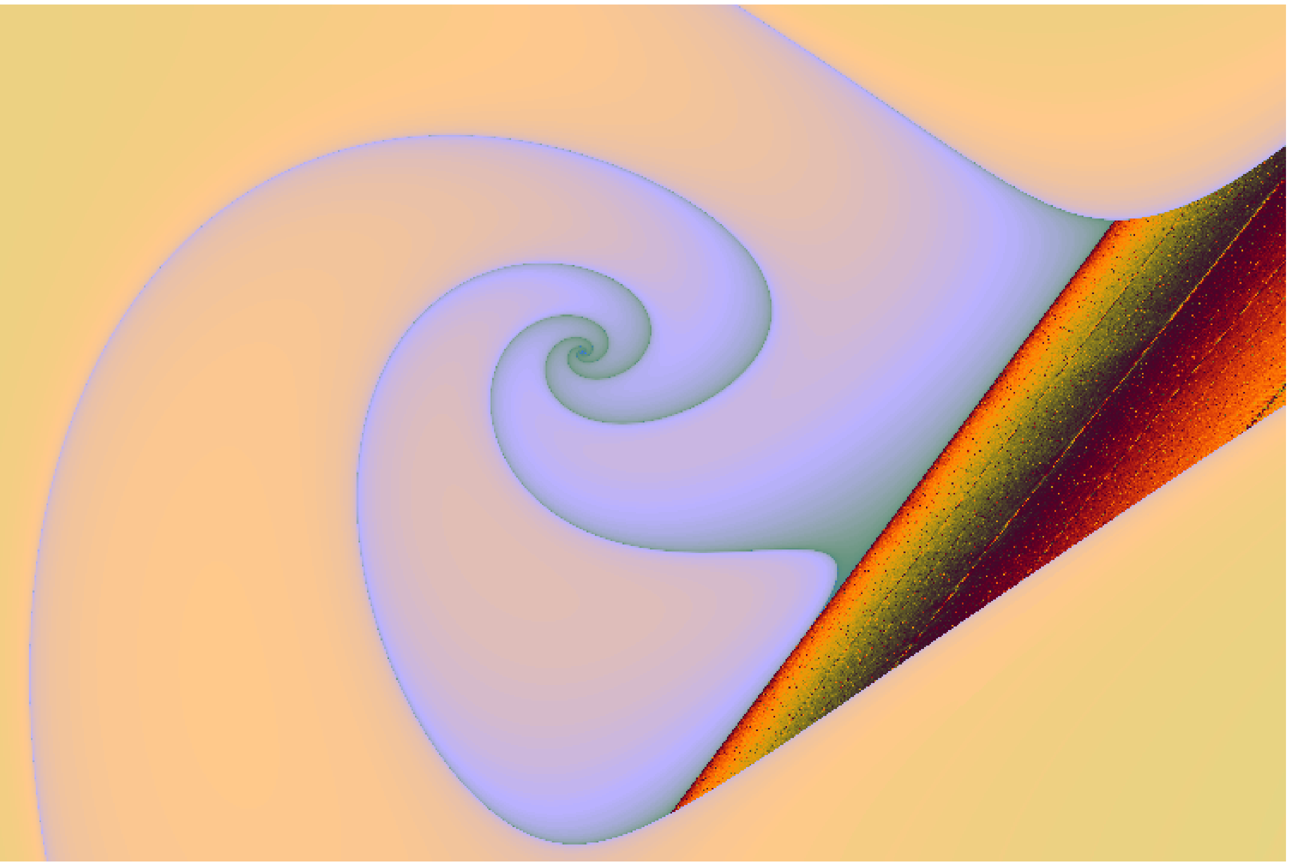


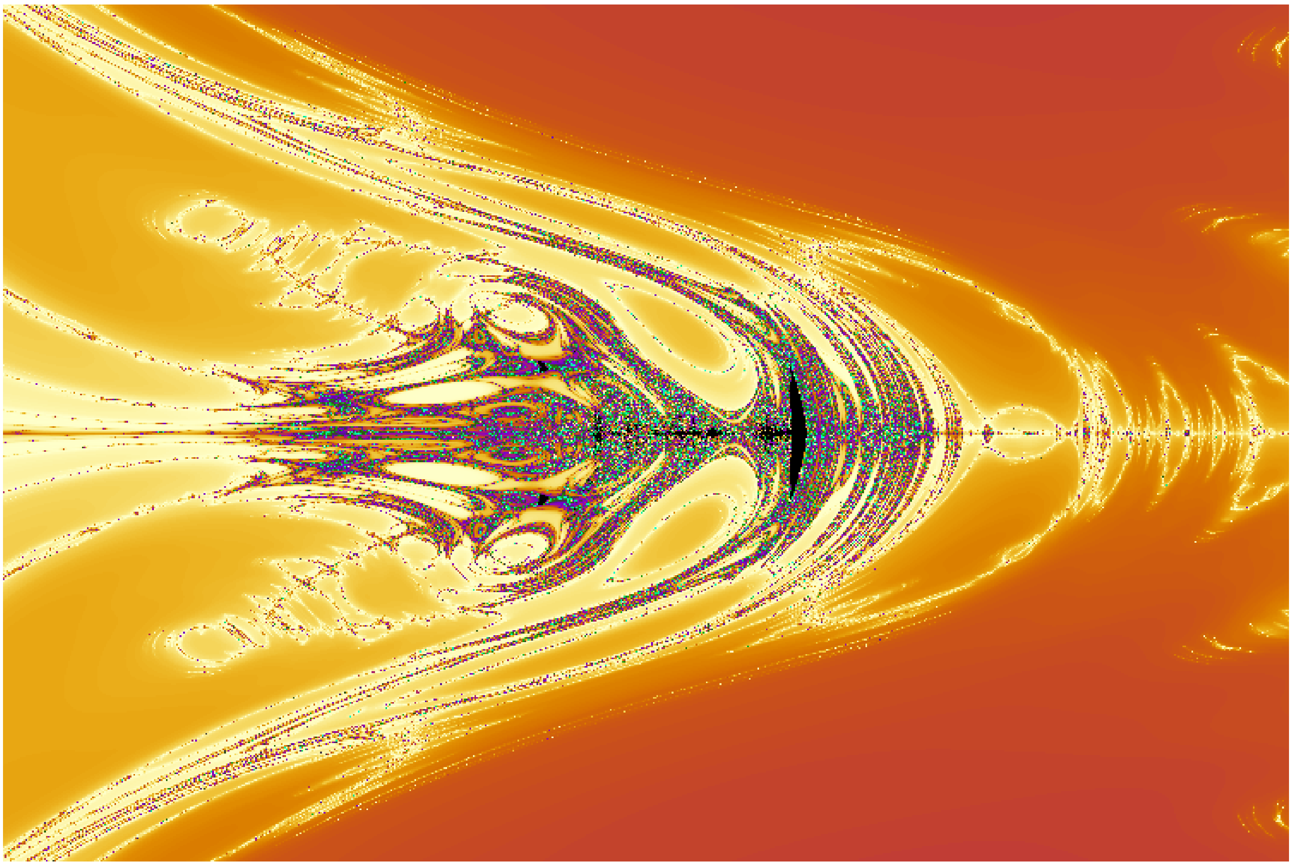
$\alpha = 1$, $\beta = 0$, $\gamma = -1$ Cubic


$\alpha = 1$, $\beta = 0$, $\gamma = -1$ Almost Cubic

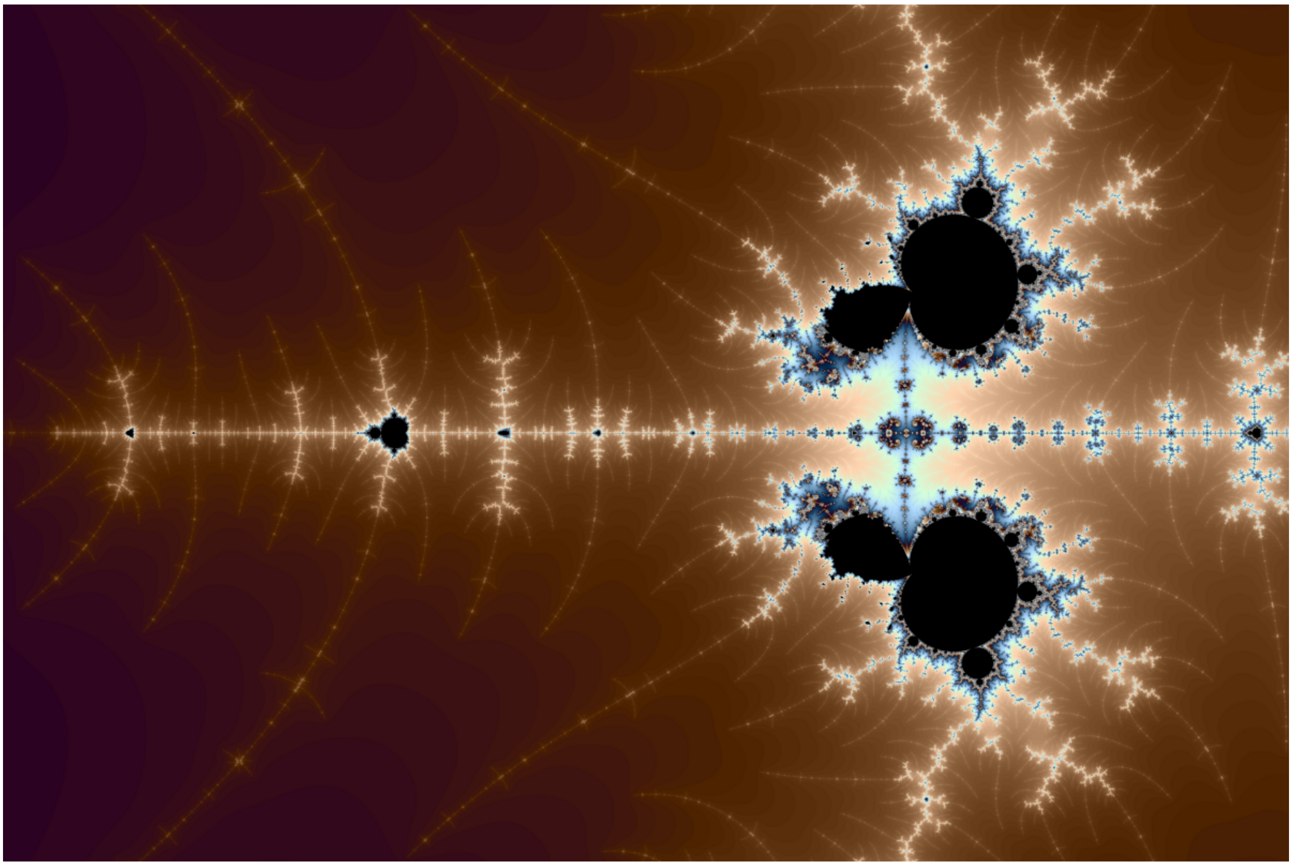
As can be seen varying the parameters α, β γ provides a wide variety of images. The examples so far have used black inner colouring and iteration outer colouring, making use of the various colouring methods and colour maps available in Saturn can enhance the images produced.


Almost.Cubic.1.spf

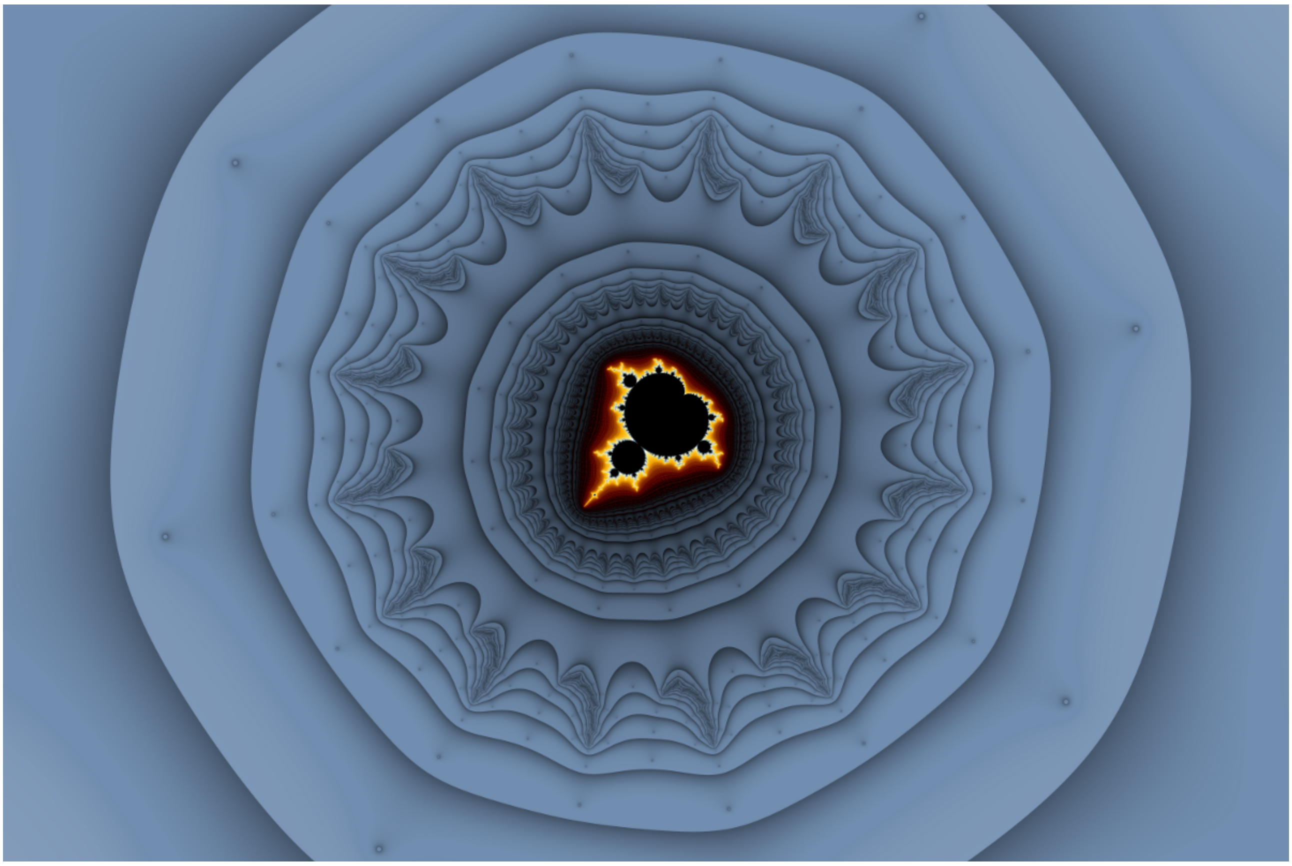

Almost.Cubic.2.spf

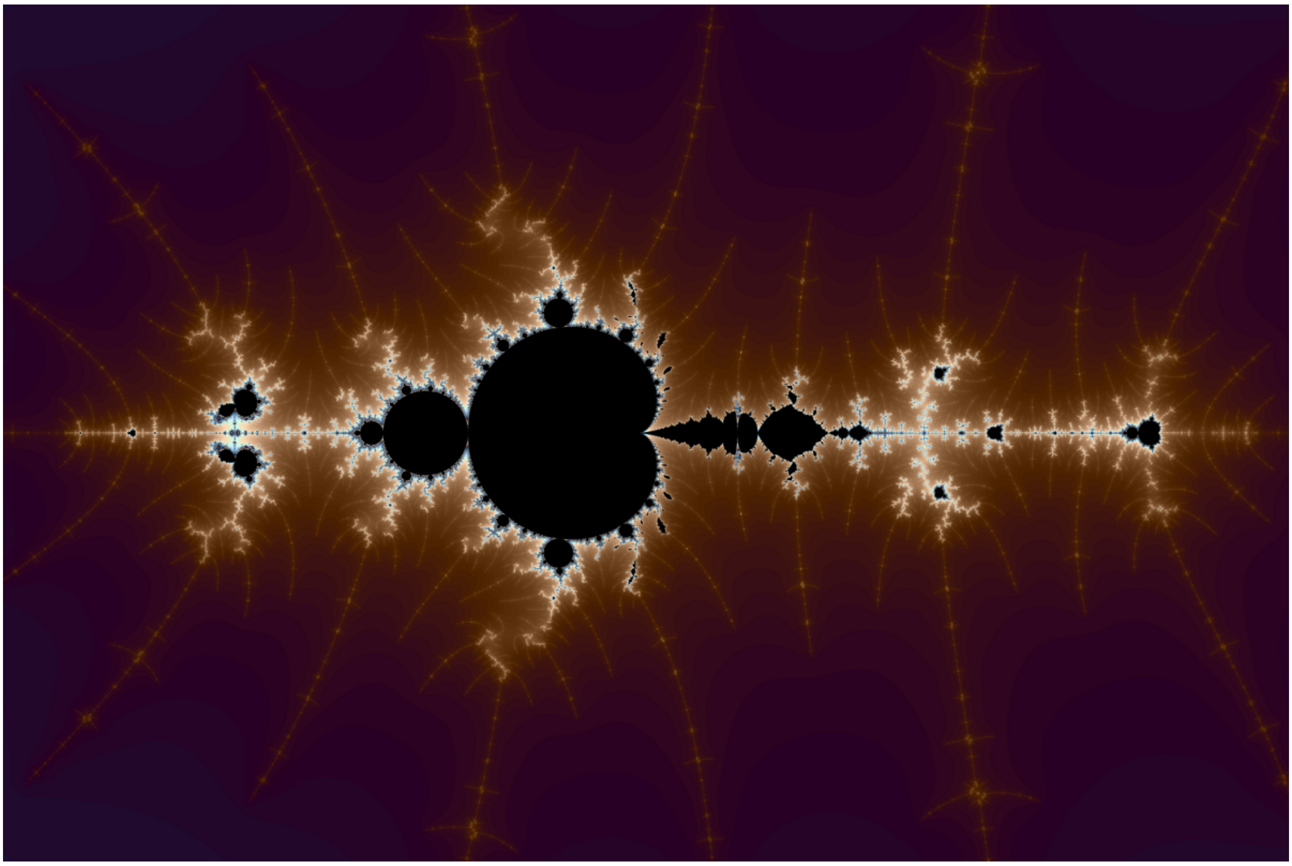

Almost.Cubic.3.spf

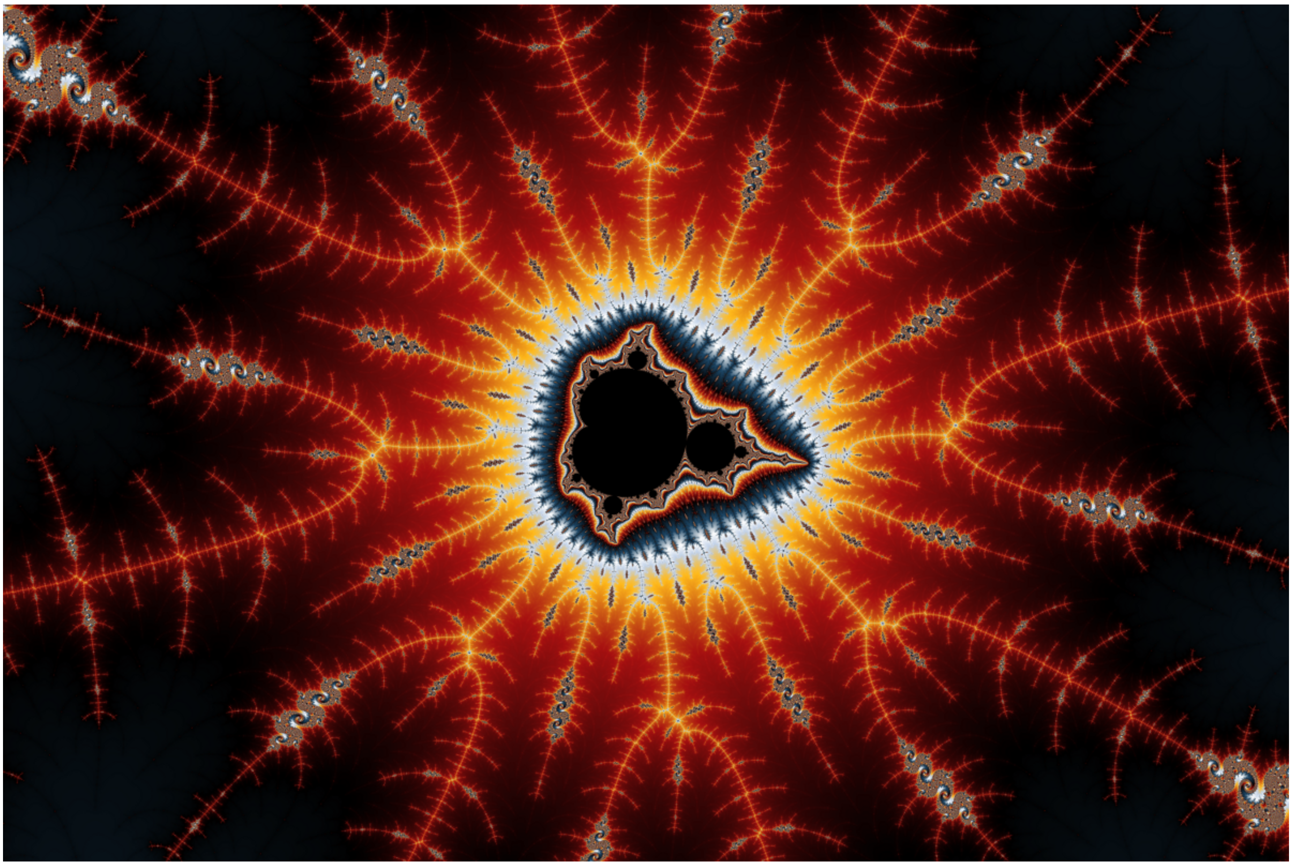
Since issue 1 of this document I've come across parameters for Cubic which result in Mandelbrot shape appearing sometimes distorted and often not.



Above, In.a.Cubic.3.spf, below In.A.Cubic.4.spf.




In.a.Cubic.5.spf

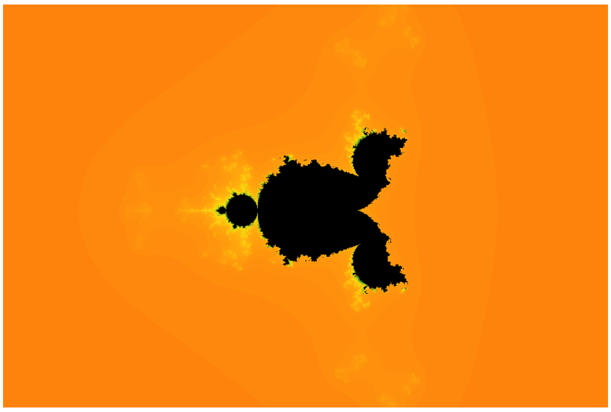

In.a.Cubic.7spf

3.1.2 Quartic

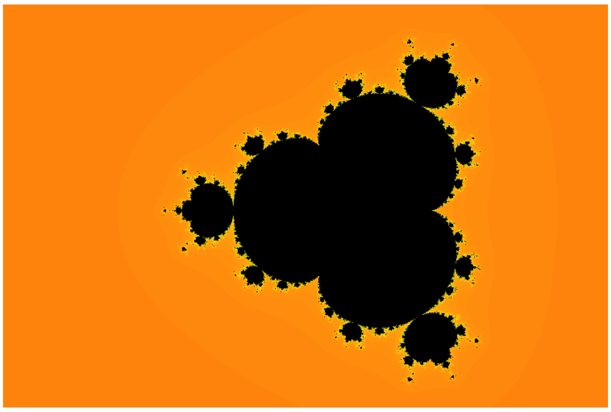
The quartic formula:

$z \leftarrow \alpha z^4 + \beta z^3 + \gamma z^2 + \delta z + c$

The default values for α, β, γ & δ are 1.0.

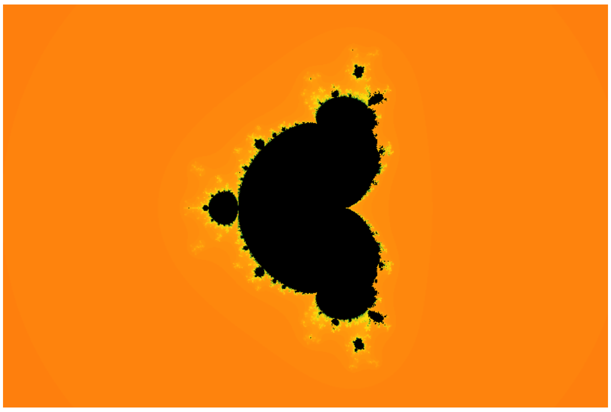


Defaultsand $z_0 = 0.0$



$\alpha = 1.0$, β, γ & $\delta = 0.0$, $z_0 = 0$

This is the 4th power Mandelbrot.



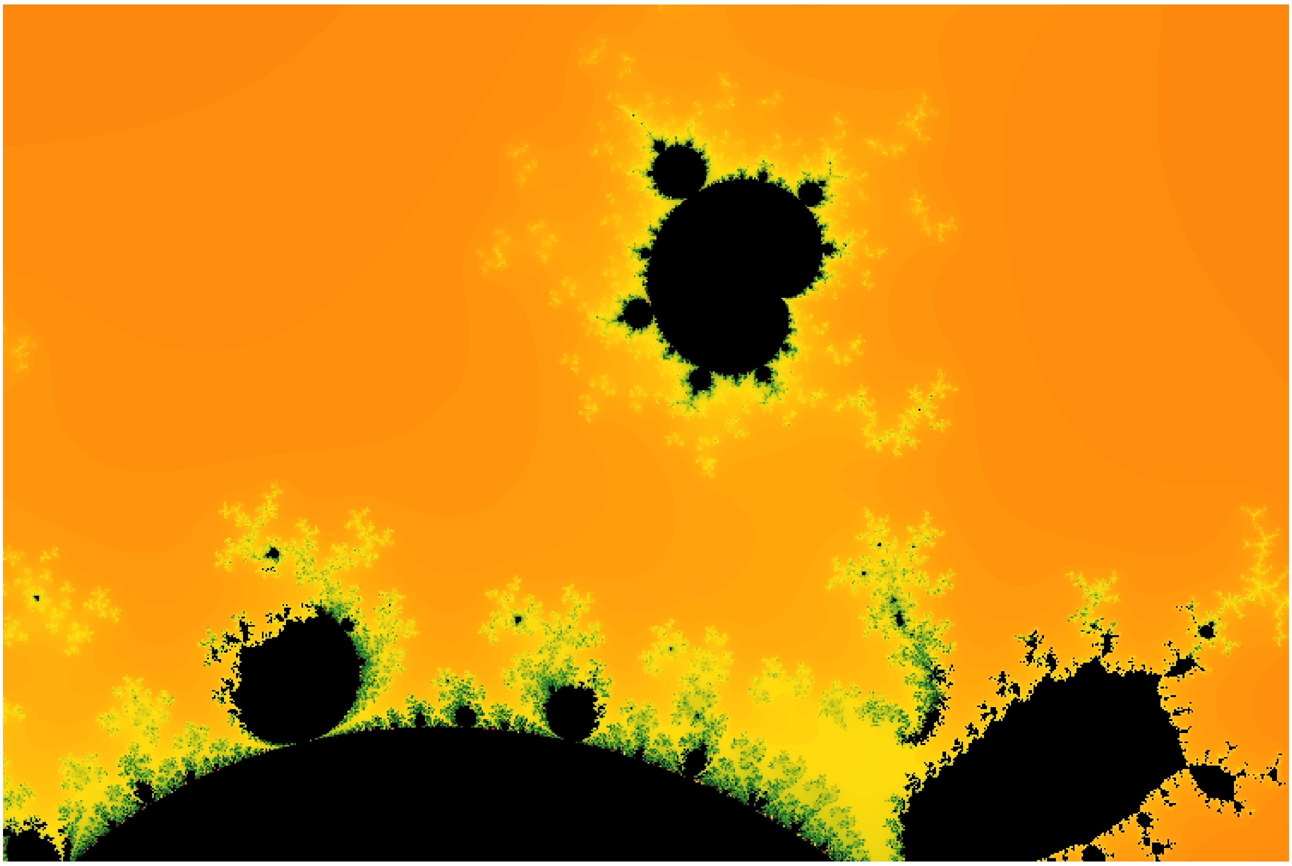
$\alpha = 0.25$, $\beta = 0.0$, $\gamma = 0.5$, $\delta = 0.0$

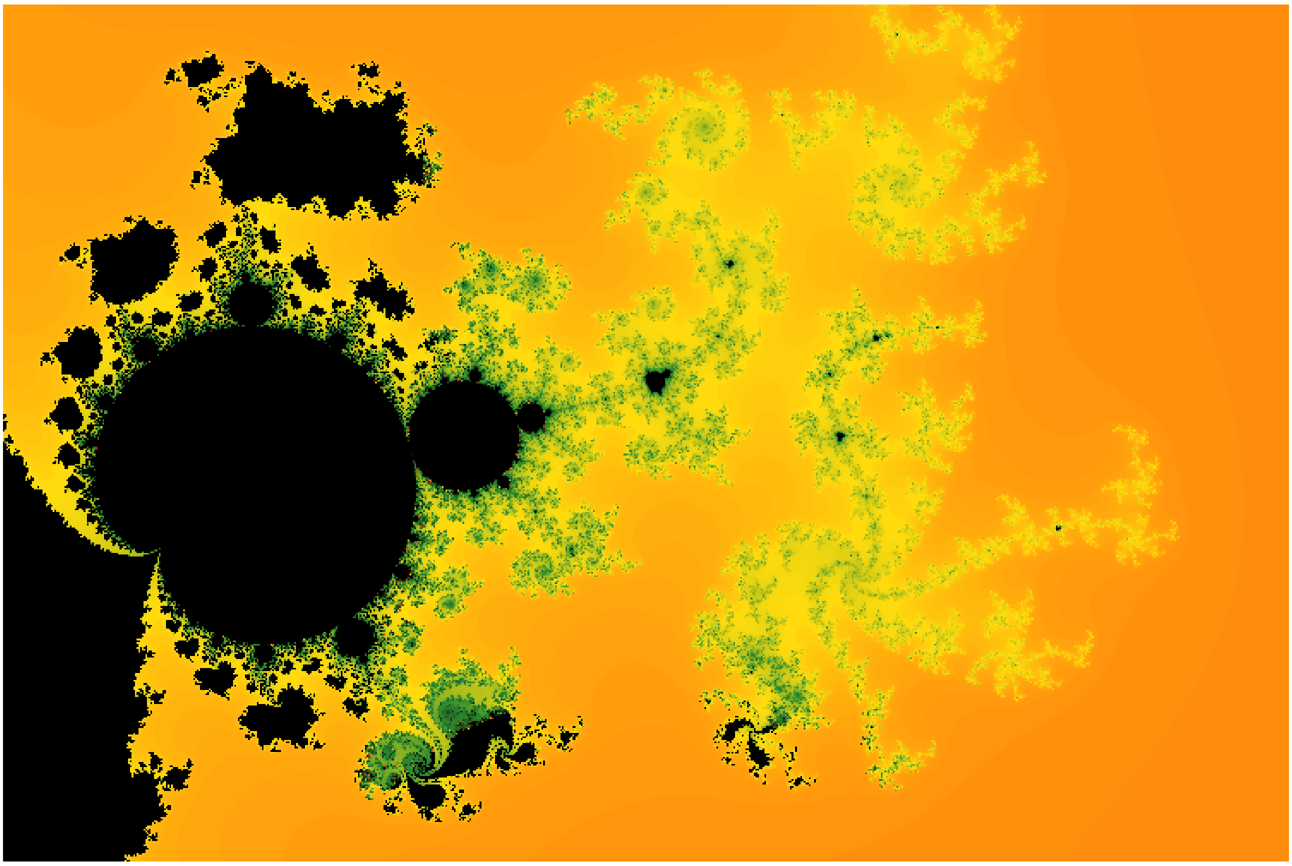
$z_0 = 0.0$

Reducing α to 0.5 shrinks the 4th power Mandelbrot and adding a 2nd power scaled by 0.5 mutates the image.

There are four parameters to play with to mutate the fractal image, you are effectively combining three different types of Mandelbrot, the standard, the cubic (3rd power) and the quartic (4th power) and varying the influence of each them with an extra z thrown into the mix.

The third picture has features of the standard Mandelbrot, some of the islands are serverely distorted others less so. This is illustrated in the next two pictures which zoom in at the top of image and on the right hand side.

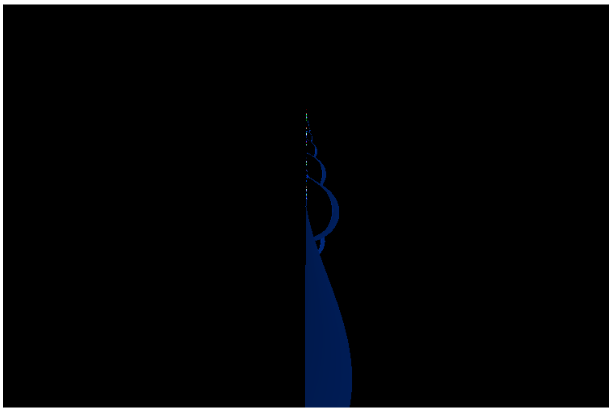

In.a.Quartic.1.spf

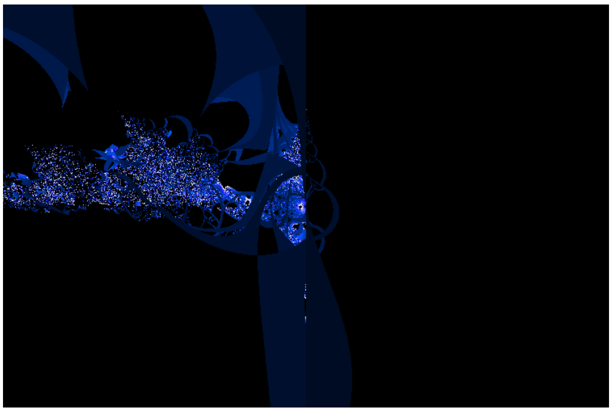

In.a.Quartic.2.spf

3.1.3 Bad Complex Powers & Bad Complex Power Julias

These fractals use incorrect implementations of complex numbers to a complex power. The formula in both cases is:

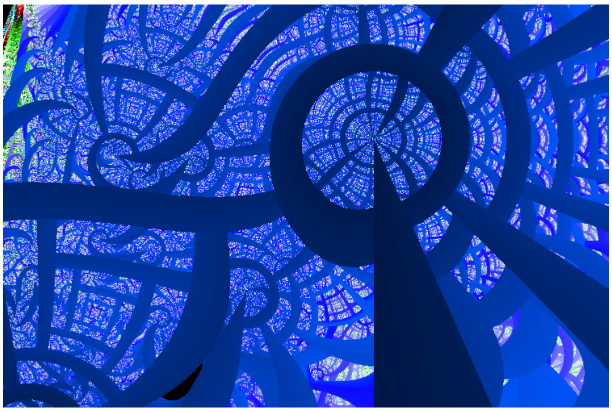
$z \leftarrow z^{\alpha} + c$



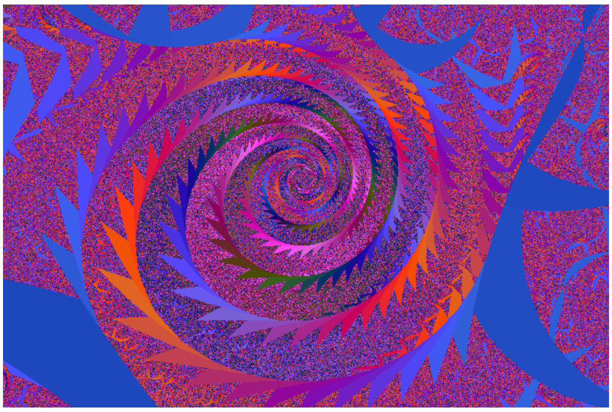


Bad Complex Power on the left and Bad Complex Power 2 on the right, both use $\alpha = i$ and default position and size.

Things get much more interesting when you zoom in.



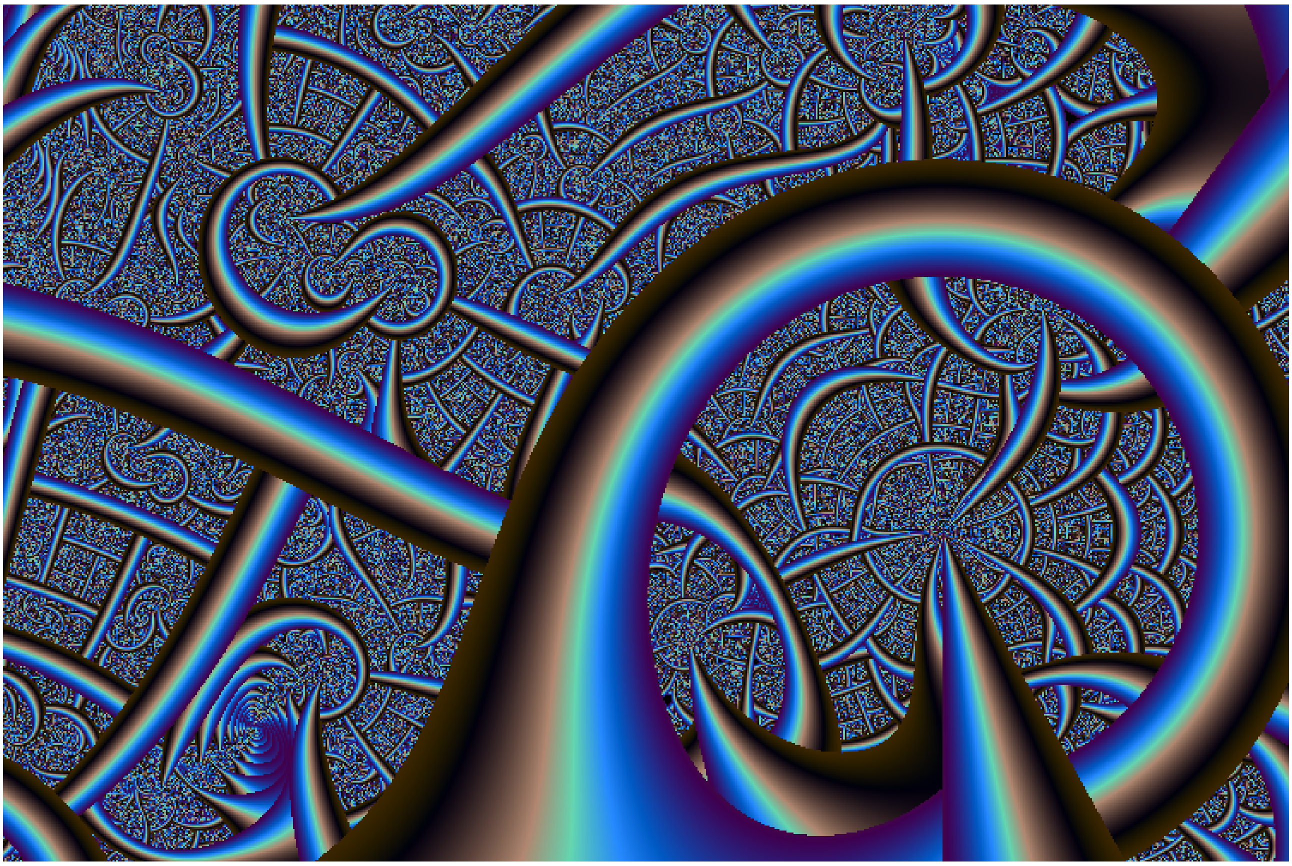
Bad Complex Power with $\alpha = 1.2i$.

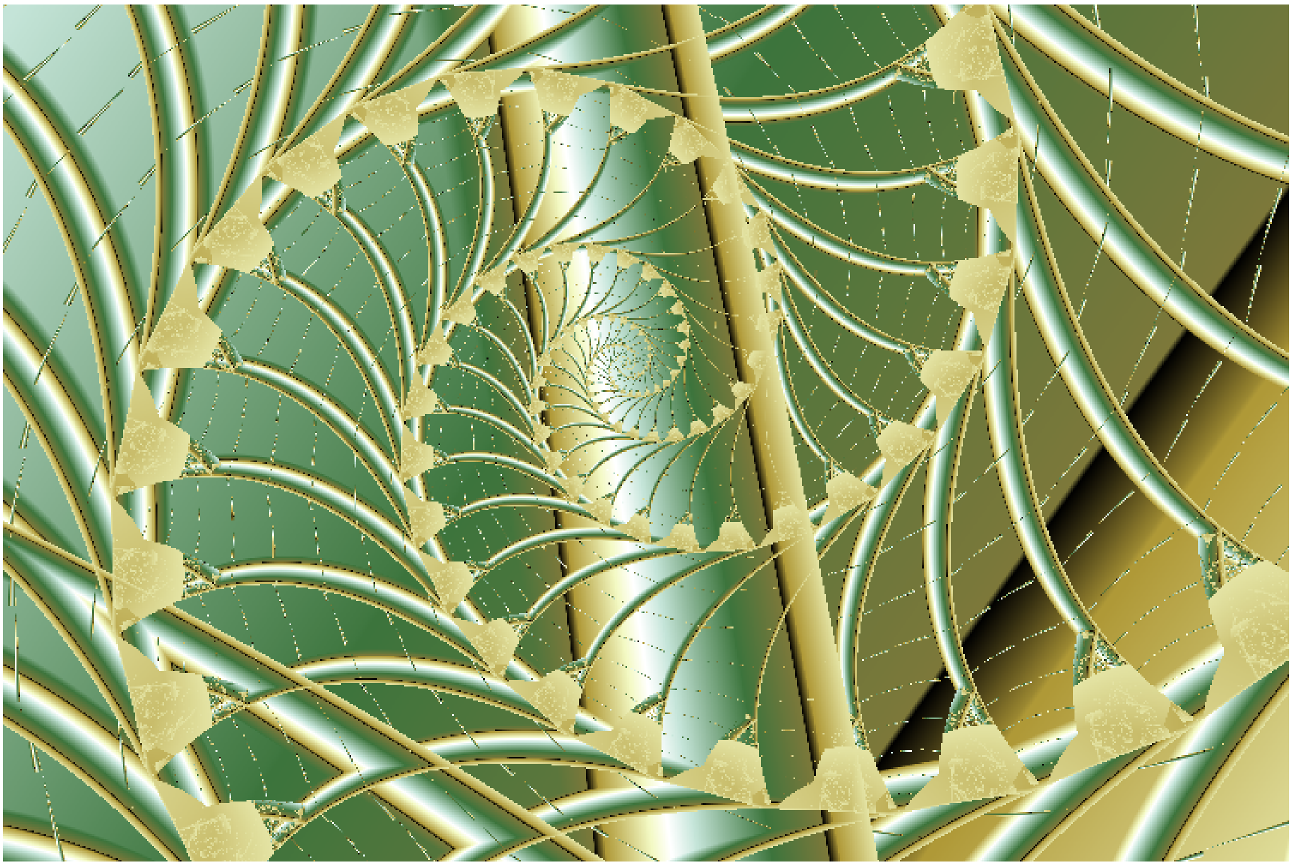


Bad Complex Power 2 with $\alpha = i$.

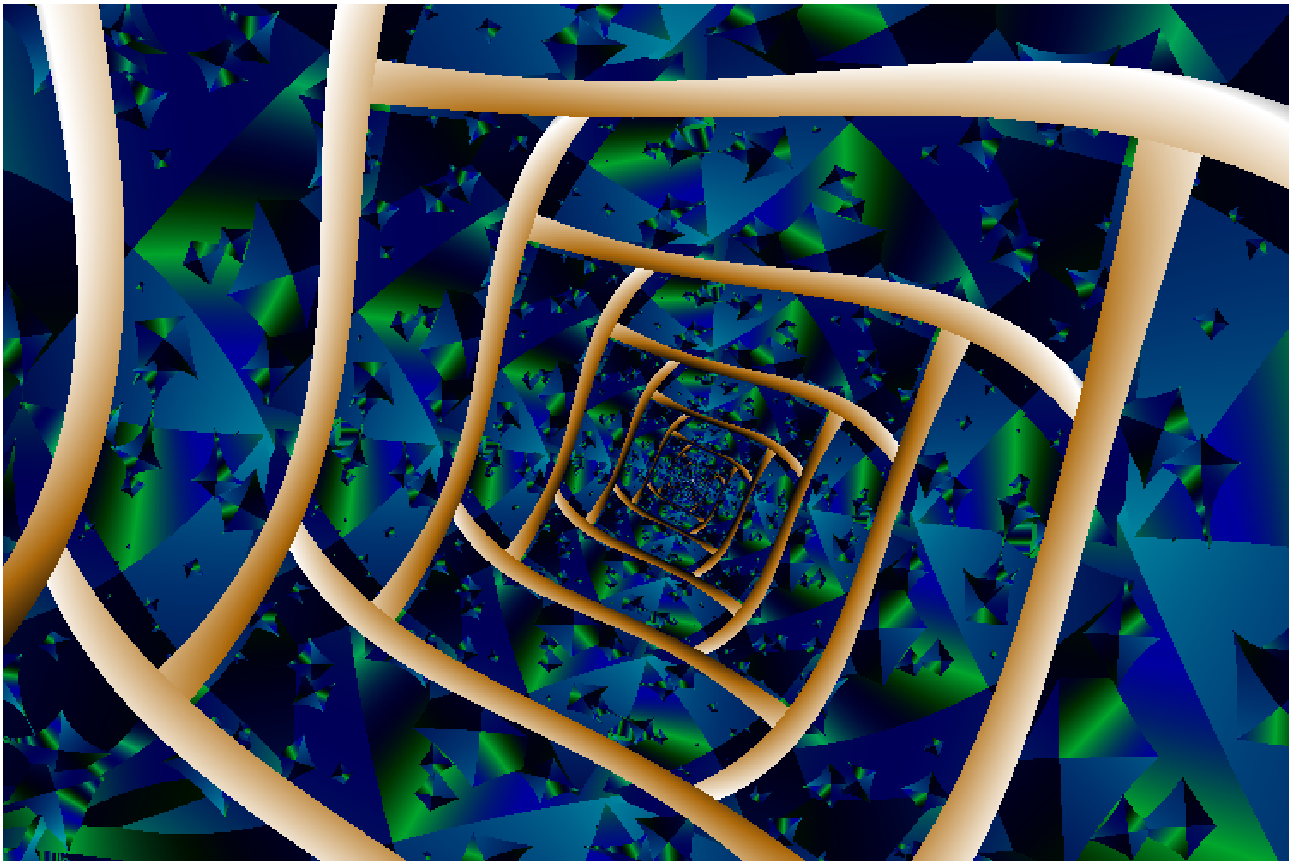
A good range of values of α is i to 2i, there may of course be others but these are good values for starting your exploration of these fractals.

Experimentation with different colouring methods and colour maps is also worthwhile.


Bcp1.spf


Bcp2.spf

Two examples of Bad Complex Power Julias



Bcp.Julia.1.spf

Bcp.Julia.2.spf

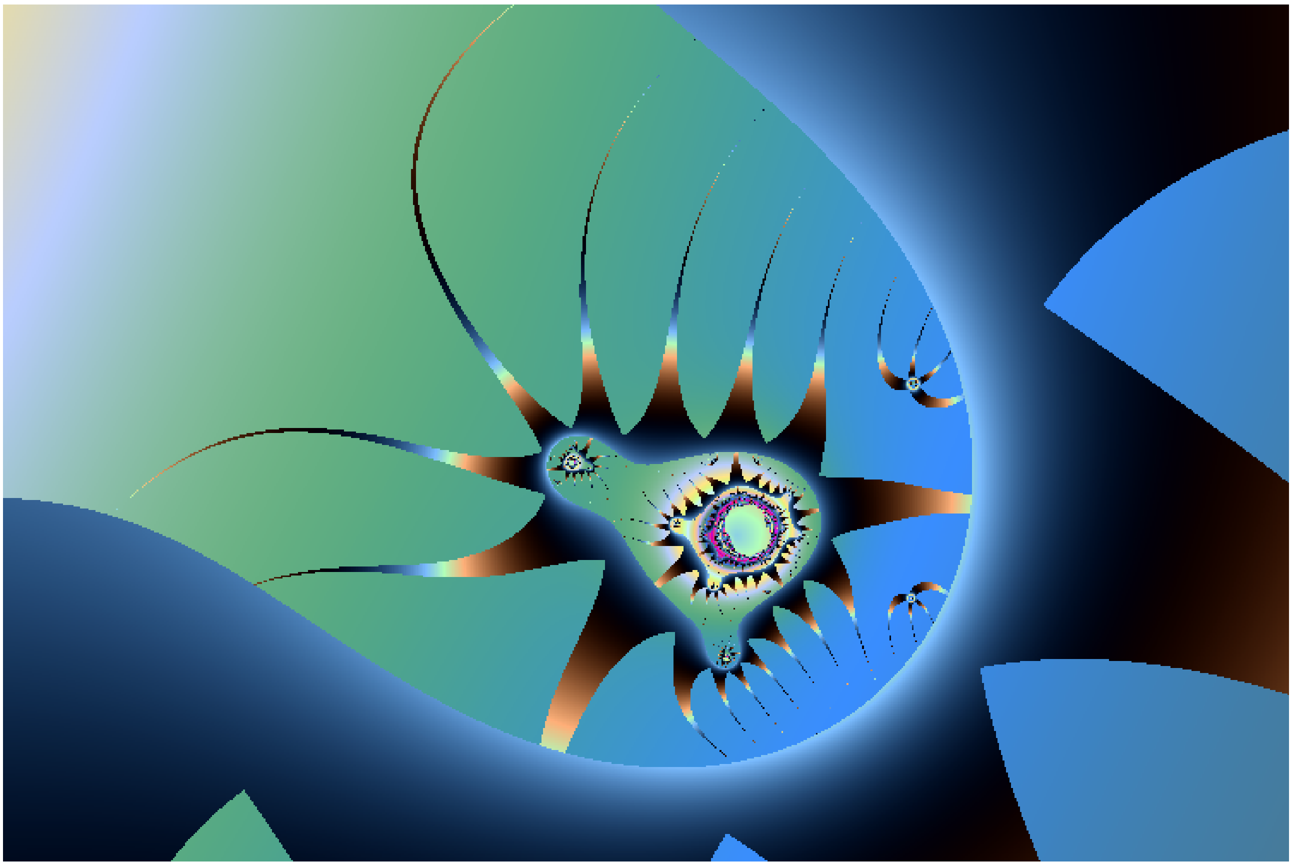
3.1.4 Biomorphs

There are 5 fractal types explicitly called Biomorph which were first implemented in an earlier program called Mars, there were originally 6 but the first one had the same formula as an other fractal type and was removed. They are essentially Julia fractals with low iteration counts and a specific method of determining colour.

Originally the Biomorphs produced by Clifford A. Pickover were coloured either black or white, I used his method for determining black or white to select between inner and outer colouring. For more information on colour selection see the Saturn User Manual.

The technique for colouring Biomorphs can be applied to all escape time fractals.

An example biomorph:



Biomorph.spf

3.1.5 Combinations & Combination Julias

These fractals feature a combination of two terms plus either c or a fixed constant. The Combination formula is:

$z \leftarrow \text{transform}(z)$
$z \leftarrow \alpha z^{\beta} + \gamma z^{\delta} + c$

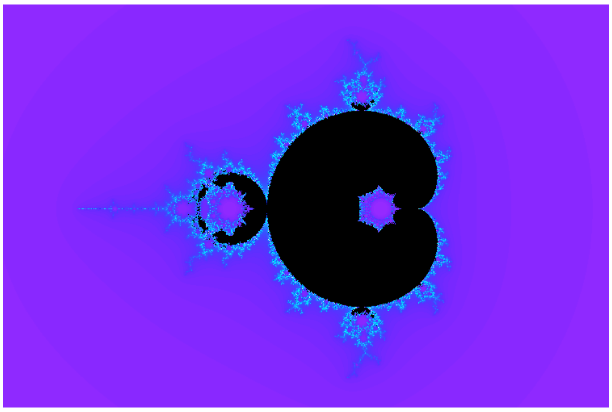
the Julia substitutes ε for c. Care must be taken with the value of z_0 so that invalid values are not used, for example the default values $\alpha = 1$, $\beta = 2$, $\gamma = 1$ and $\delta = -1$ should not use $z_0 = 0$ as it results in a rectangle of a single colour. If you want to disable the second term by setting γ to 0 you will not get the expected Mandelbrot unless you also change δ to 1 because if you leave δ at -1 the result of the first iteration will be “not a number” the escape condition will never be met and computation will be very slow.

The Combination 2 formula is:

$z \leftarrow \text{transform}(z)$
$z \leftarrow c\alpha z^{\beta} + \gamma z^{\delta}$

there is no Combination 2 Julia as it is a subset of the Combination Julia (ε set to 0).The value of z_0 should be set to 0.

The best results when using these formulae is when one term uses a positive power, the other uses a negative power and one term is scaled to be considerably smaller than the other.



3.2
3.3
3.4
3.5
3.6
3.7
3.8
3.9
3.10

$\alpha = 1$, $\beta = 2$, $\gamma = -1.25\text{e-}07$, $\delta = -6$
z_0 = transformed c

3.10.1 Compasses & Compasses Julia

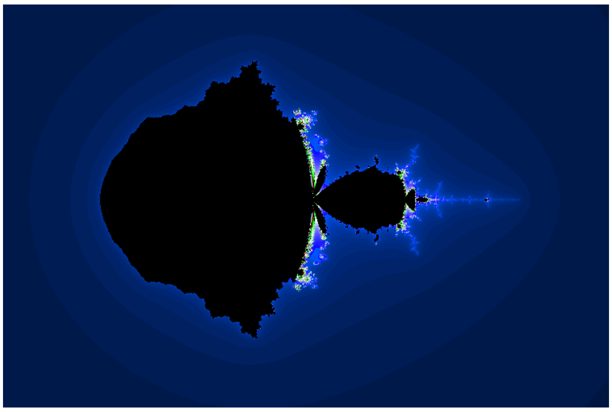
As is typical of a lot of fractal formulae these are simple, though slightly more involved than the Mandelbrot formula:

$$z \leftarrow \text{transform}(z)$$
$$z \leftarrow z^{\alpha} - \alpha c^{\alpha-1} z + \beta$$

and the Julia form:

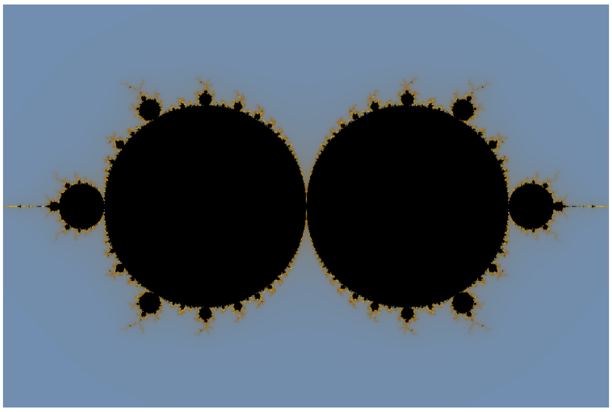
$$z \leftarrow \text{transform}(z)$$
$$z \leftarrow z^{\alpha} - \alpha \beta^{\alpha-1} z + \gamma$$

The Compasses formula produces images similar to the standard Mandelbrot, Mandelbrot Islands and distorted variants also appear.

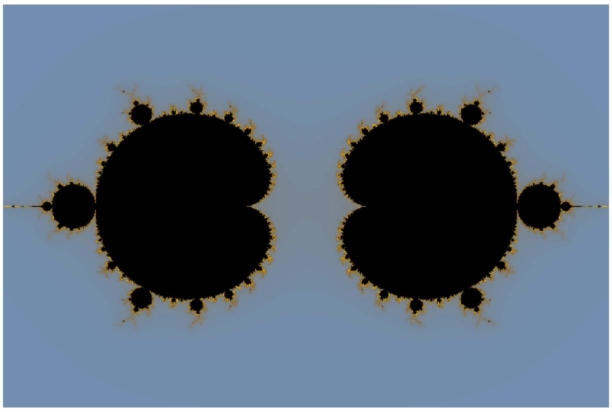


$\alpha = 2$, $\beta = 0$
$z_0 = 1$

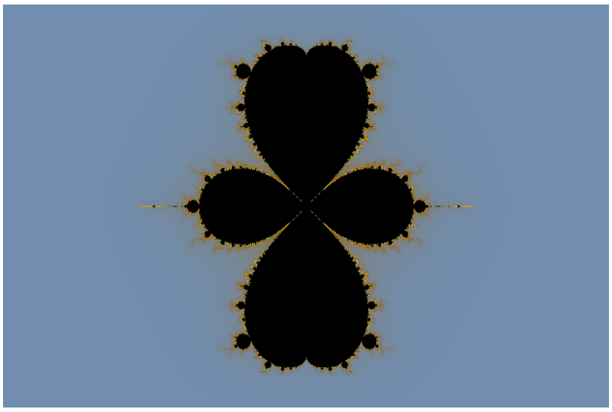
That image isn't that interesting, however if the location in the complex plane is used instead for the initial values of z we get the following images with varying values of β:



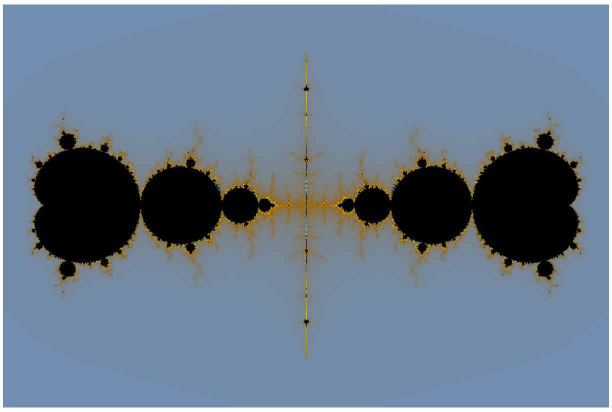
$\alpha = 2$, $\beta = 0$
$z_0 = c$



$\alpha = 2$, $\beta = 0.0875$
$z_0 = c$



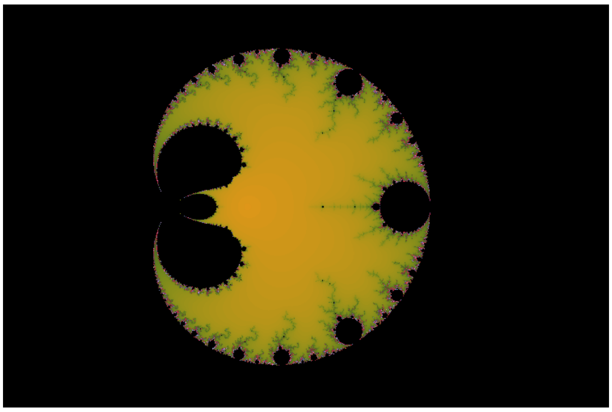
$\alpha = 2$, $\beta = -1$
$z_0 = c$



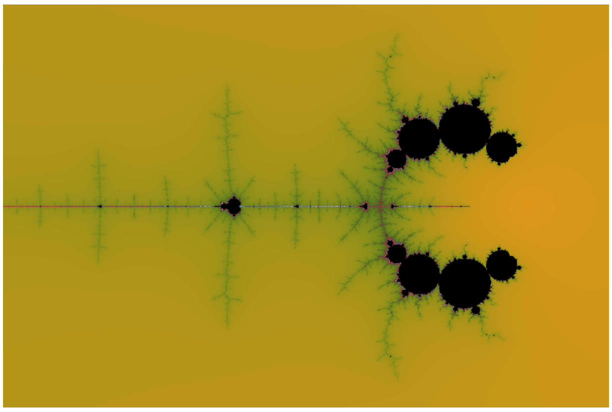
$\alpha = 2$, $\beta = -1.665$
$z_0 = c$

Note: this image has been rotated 90 degrees

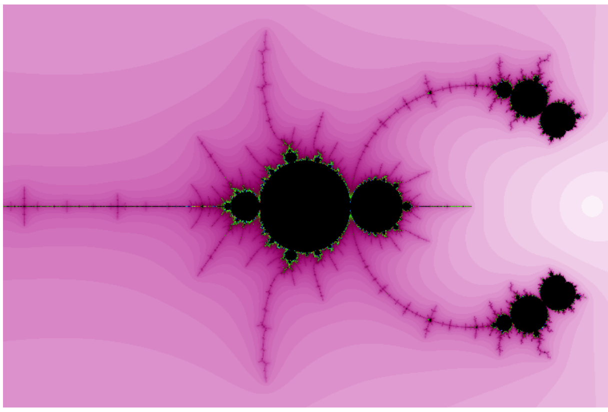
As can be seen shapes similar to the Mandelbrot are generated. As with the Mandelbrot inverse Compasses are also worth investiagating, there now follow three Inverse Compasses with varying values for β.



Complex Plane tranform, power, -1
$\alpha = 2$, $\beta = -1$
z_0 = transformed c

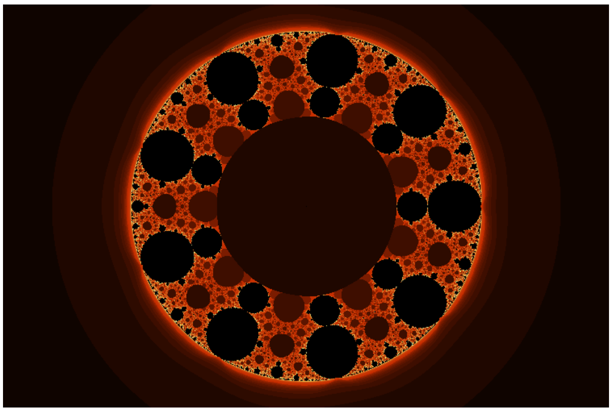


Complex Plane tranform, power, -1
$\alpha = 2$, $\beta = -1.665$
$z_0 = c$

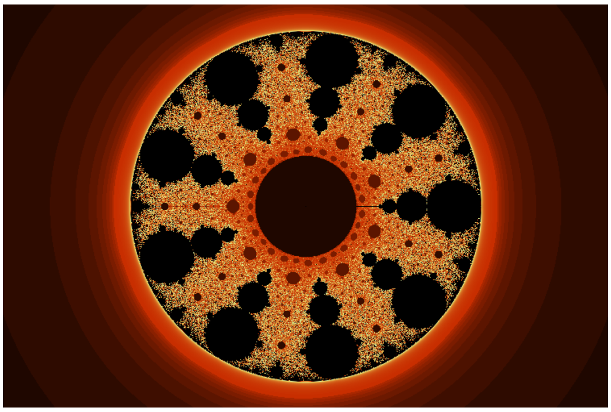


Complex Plane tranform, power, -1
$\alpha = 2$, $\beta = -2$
$z_0 = c$

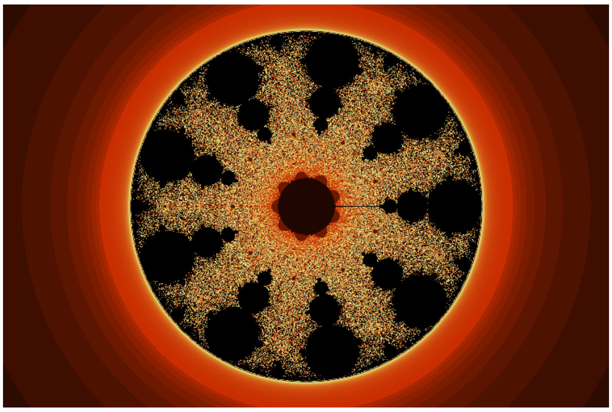
So, that's a positive power ($\alpha = 2$) and Inverse Compasses, now for a negative power ($\alpha = -2$). The negative power Compasses fractal is senstive to the escape limit.



$\alpha = -2$, $\beta = 0$
$z_0 = c$
Limit = 16

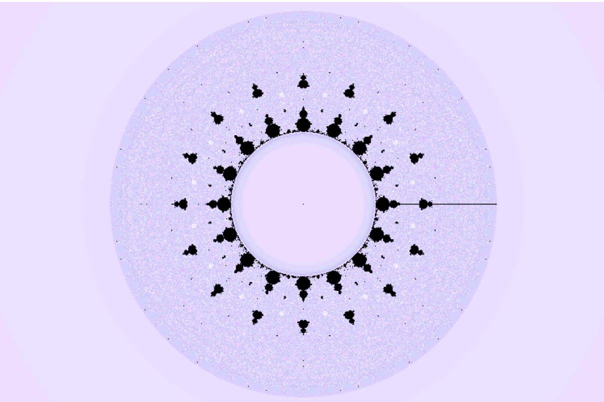


Limit = 1600

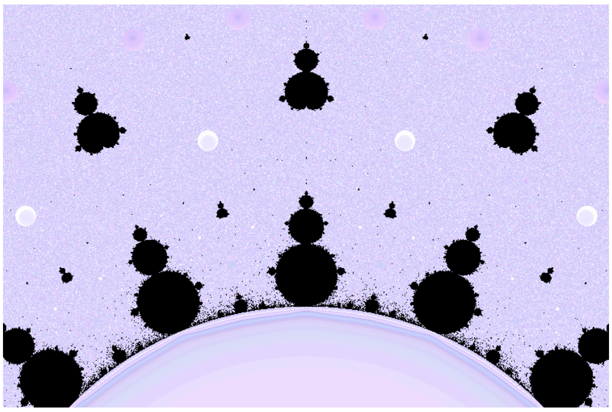


Limit = 160000

Now a negative power Inverse Compasses.

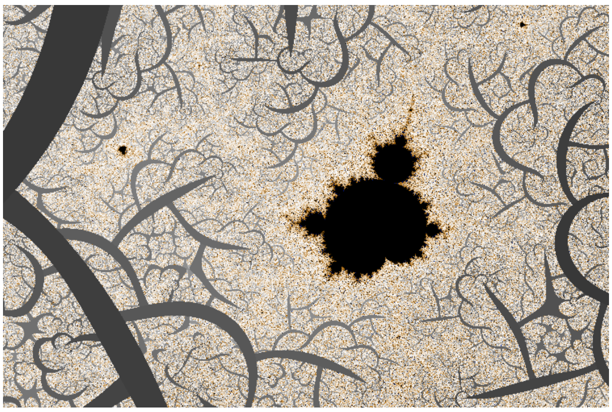


Complex Plane tranform, power, -1
$\alpha = -3$, $\beta = 0$
$z_0 = c$

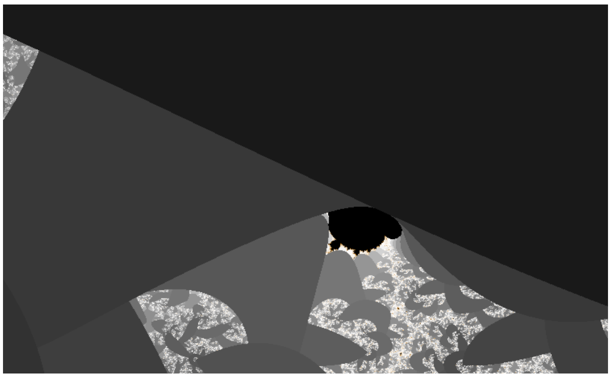


Zooming in elongated Mandelbrot Islands resolve themselves.

We still have some more interesting byways of the Compasses fractal and they are the imaginary powers. These are also sensitive to the escape limit.

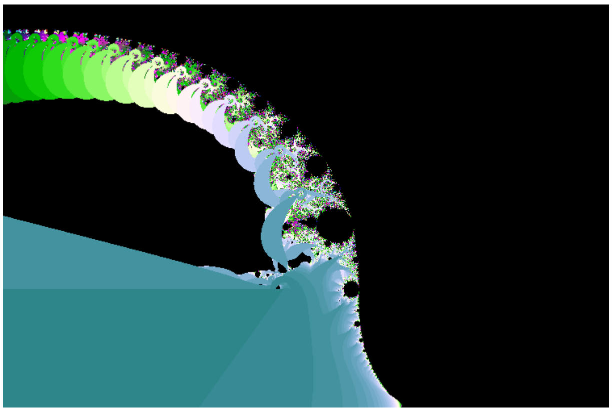


$\alpha = -1.75i$, $\beta = 0$
$z_0 = c$
Limit = 16000



This shows why the limit value is important, this is the same image as the previous one with a limit of 160.

The limit value used to get interesting pictures is dependant on the value of α.

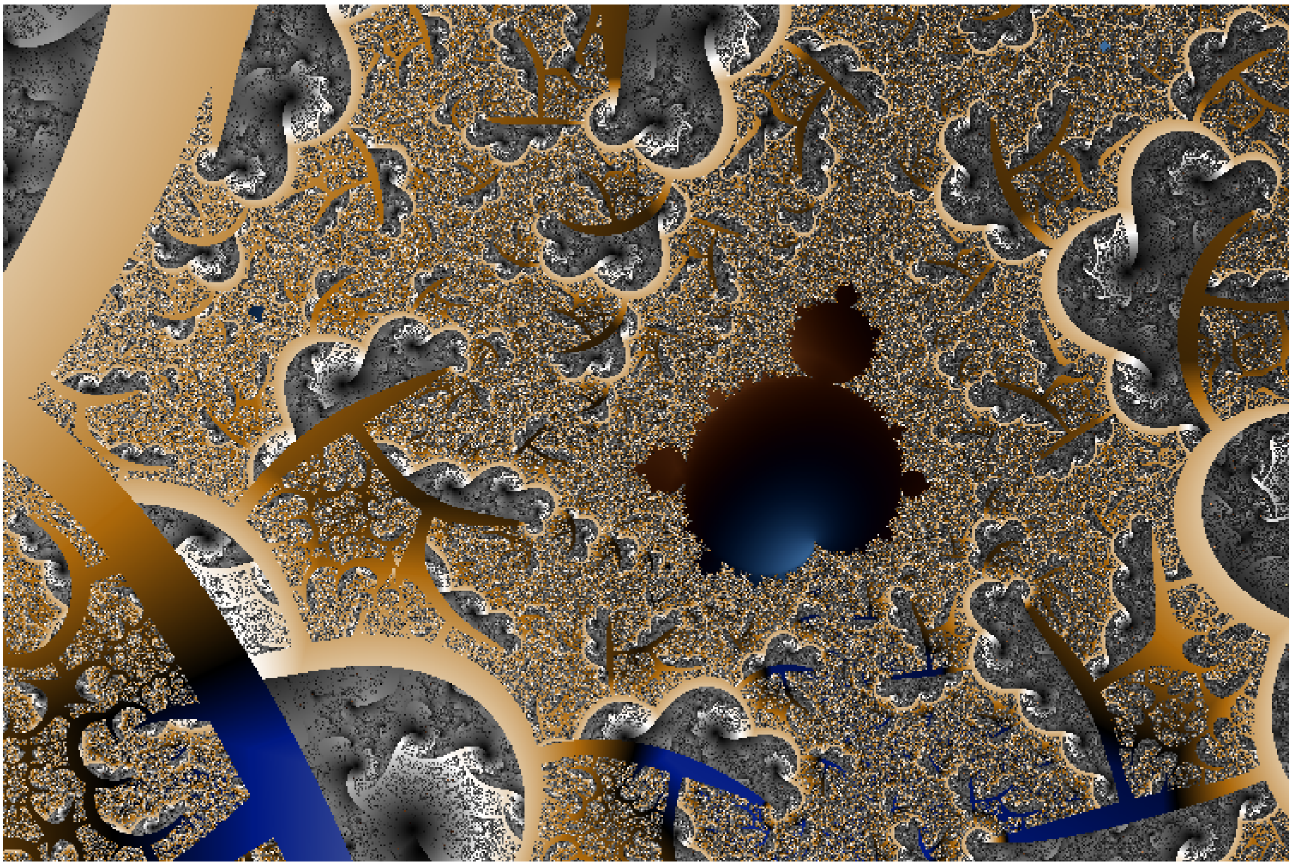


α = i
Limit = 160

Limit = 16000

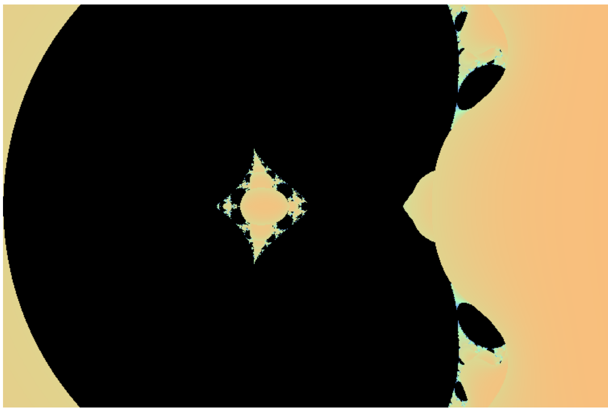
Good values of α for exploring imaginary power Compasses fractals start at around 1i and continue beyond 2.5i. The colouring methods used in these examples has been restricted to iteration colouring for outer and fixed colour (black) for inner.

Use of colouring methods for inner and outer colouring will enhance the images, you'll need to experiment with the methods and the colour maps to find pleasing combinations.

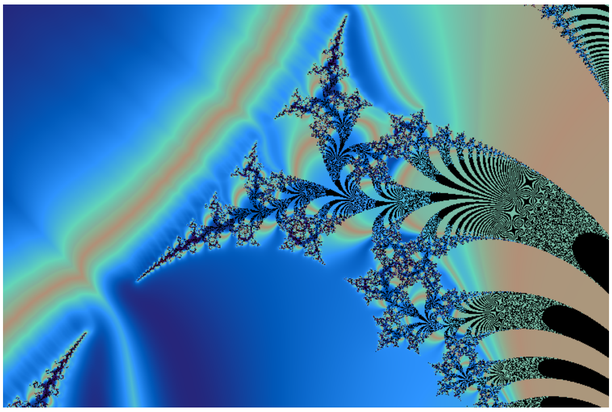

Compasses.spf

3.10.2 CpowerZ & ZpowerC

These fractals are relatively boring and as such haven't been explored or transformed much at all so there isn't that much to say about them. Here are the basic fractals with iteration for outer colouring and black for inner.



CpowerZ on the left and ZpowerC on the right.



Zoom in on the island in CpowerZ, no bailout, 45 iterations, coloured using coefficient of variation of change in magnitude.

3.10.3 Hybrid and Hybrid Julia

This pair of fractals are generalised versions of a couple of formulae created by Pablo Roman Andrioli (aka Kali on Fractal Forums and FractKali on DeviantArt). These formulae were originally implemented in Mars (Saturn's forerunner) as four distinct formulae.

The formulae implemented in Saturn are:

$$n \leftarrow \text{norm}(z)$$
$$z \leftarrow \text{transform}(z)$$
$$z \leftarrow \alpha(z_r/n)^\beta + \gamma(z_i/n)^\delta + c$$

for the Julia version c is replaced with ε.

Kali's formulae are:

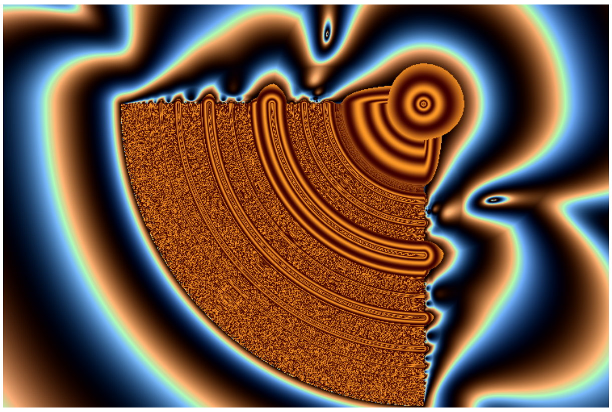
$$z \leftarrow \text{abs}(z_r)/\text{norm}(z) + \text{abs}(z_i)/\text{norm}(z) + c$$
$$z \leftarrow z_r/\text{norm}(z) + \text{abs}(z_i)/\text{norm}(z) + c$$

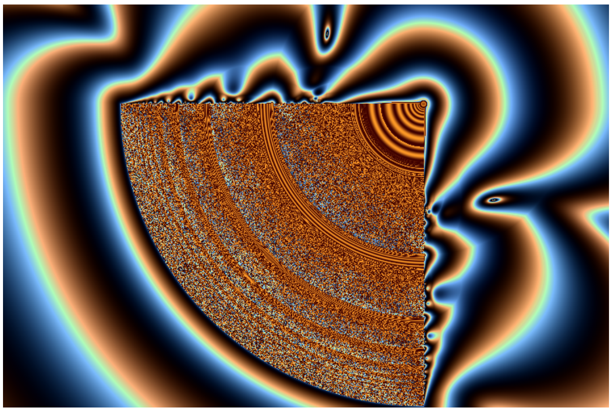
for the Julia versions c would be replaced with α.

To use Kali's formulae α, β, γ, and δ are all set 1, the first formula would have the transform “Top Right” applied and the second would have “Unsign Imaginary” applied.

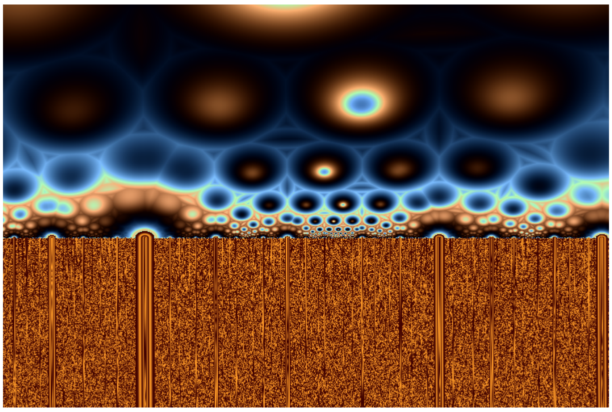
These fractals are not much use unless transforms are applied, the Mandelbrot algorithm forms aren't that interesting either. The Julia algorithm forms on the other hand offer a rich variety of images.

Here is a Hybrid with α, β, γ and δ all set to 1, top right transform, outer colouring absolute log of minimum magnitude and inner colouring absolute log of coefficient of variation of magnitude. It is sensitive to the escape limit so there are two examples each with a different limit.



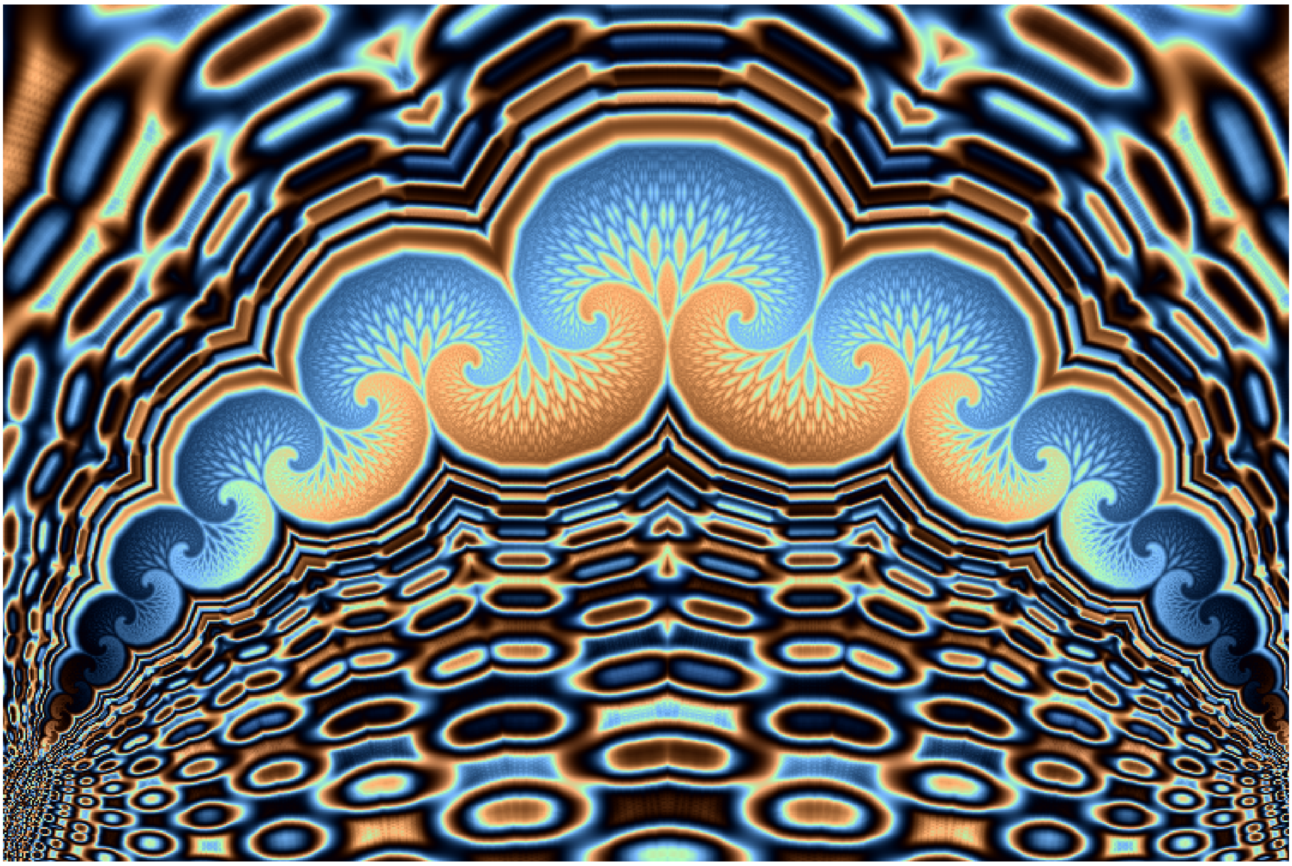


The limit used for the picture on the left is 16 and on the right it is 1600.



Zooming in on the top edge of the quarter quarter circle.

The Julia forms of these fractals are tricky, so far the best result have been with α, β, γ and δ all set to 1 however it can take a long time adjusting the ε values to find anything of interest. Start with ε set to zero and try small increments of the real and imaginary components, the bailout condition should be set to "no bailout" and inner colouring should be set to one of the magnitude statistics, average and options after average should give a good indication. The way that Saturn displays its images will give you an idication of whether you are on the right track or not. The images you get are highly dependent on the number of iterations, again Saturn will help you here because if the image degrades into a mush as the iterations mount just reduce the maximum number of iterations (entry box is above the image).



Hybrid.Julia.spf

The parameter file for this will help you get started with the Hybrid Julia.

3.10.4 Novas

The Nova fractals are variants of Newton fractals and are convergence escape time fractals i.e. the bailout condition checks for values less than a limit instead of greater than a limit as used for divergence escape time fractals.

The basic formula of the Nova fractal is:

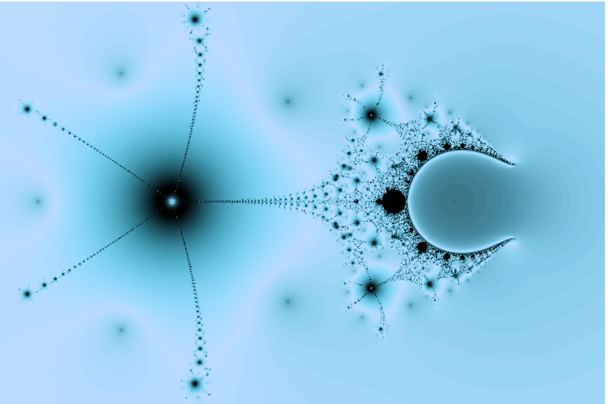
$$z \leftarrow z - \alpha(f(z)/f'(z)) + c$$

f(z) is a function and f'(z) is its derivative.

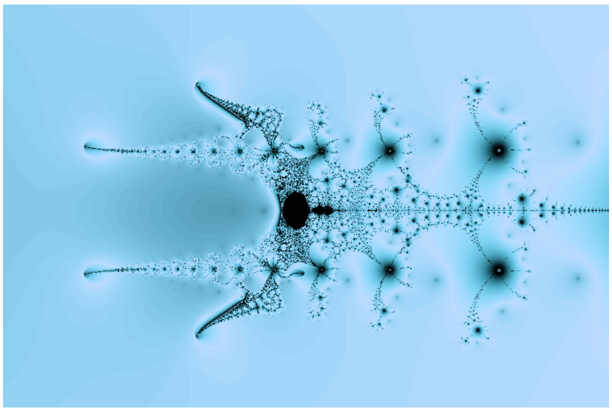
For the Generalised Nova the function and derivative are:

$$f(z) = z^{\beta} - 1$$
$$f'(z) = \beta z^{\beta-1}$$

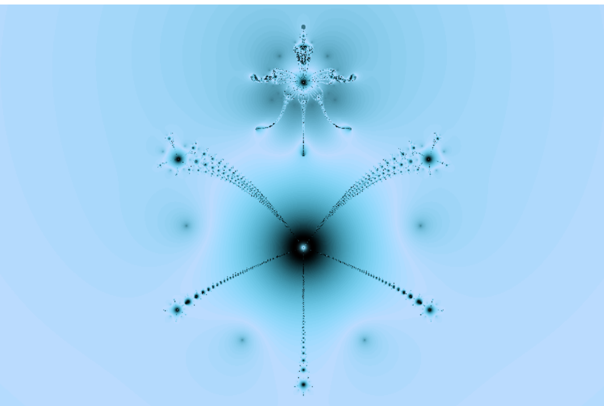
The initial value of z can not be 0 as the result of the second term will be infinity, the escape condition can't be met and calculation will continue resulting in “not-a-number” which when used in calculations results in VERY SLOW computation. For Mandelbrot Islands in your fractal use the Generalised Nova and an initial value of z of 1. The initial value doesn't have to be 1 it can be anything you like other than zero. Negative initial values produce mirror images about a vertical axis of the images produced with positive initial values.



$f(z) = z^6 - 1$
$f'(z) = 6z^5$
$z_0 = 1$

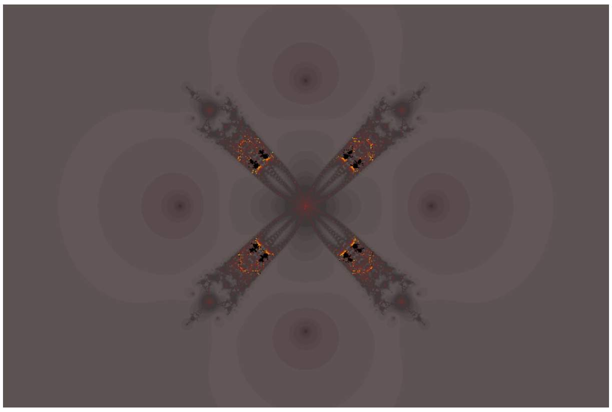


$f(z) = z^6 - 1$
$f'(z) = 6z^5$
$z_0 = -4$



$f(z) = z^6 - 1$
$f'(z) = 6z^5$
$z_0 = 1i$

Instead of using a fixed initial value of z, the location in the complex plane can be used so that a different starting value is used for every point calculated.

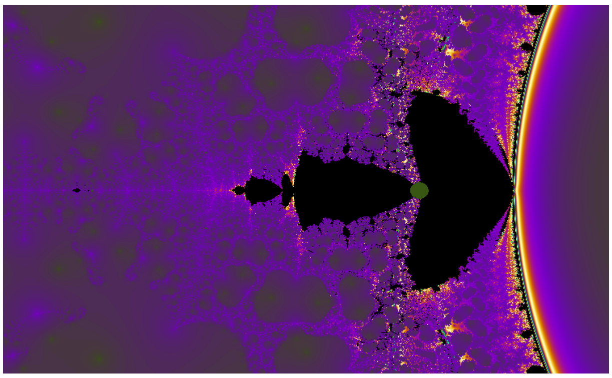


$f(z) = z^4 - 1$
$f'(z) = 4z^3$
$z_0 = c$

All the Novas so far have an α of 1 and have been coloured using iteration for outer colouring and fixed colour (black) for inner colouring.

There are 3 types of Nova formula implemented in Saturn, “Generalised Nova”, “Nova 1” and “Nova 2” with respective functions:

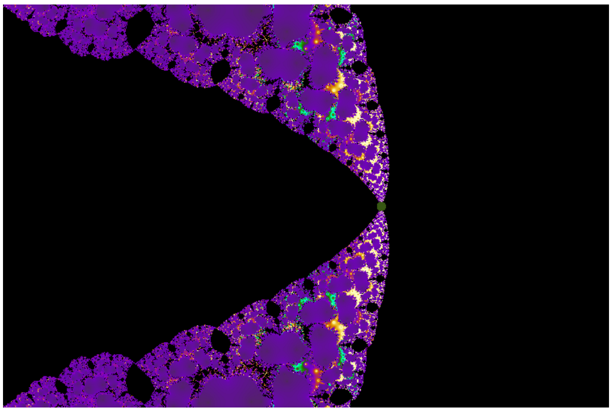
$f(z) = z^\beta - 1$
$f(z) = \beta z^\gamma - \delta z^\varepsilon - 1$
$f(z) = z\beta - (c - 1)z^\gamma - c$



Nova 1
$f(z) = z^4 - z^3 - 1$
$f'(z) = 4z^3 - 3z^2$
$z_0 = 1$

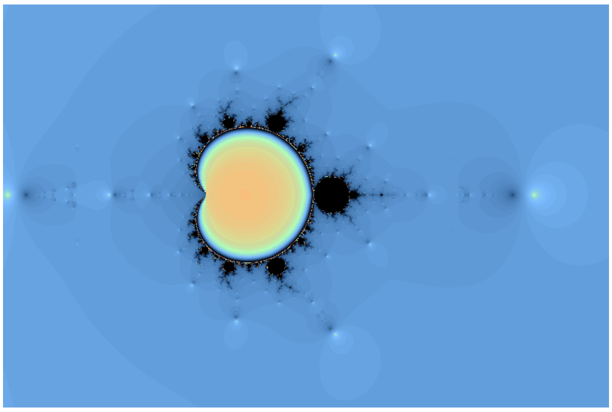
Note: no Mandelbrot Islands and a good example of the Nova Spot.

The Nova Spot is an artifact of the escape condition, the smaller the escape limit the smaller the Nova Spot. Nova Spots occur in many locations in Nova fractals and vary in size. The Nova Spot in the above example is with an escape limit of 0.01, changing the limit to 0.001 and zooming in we get a much smaller spot:



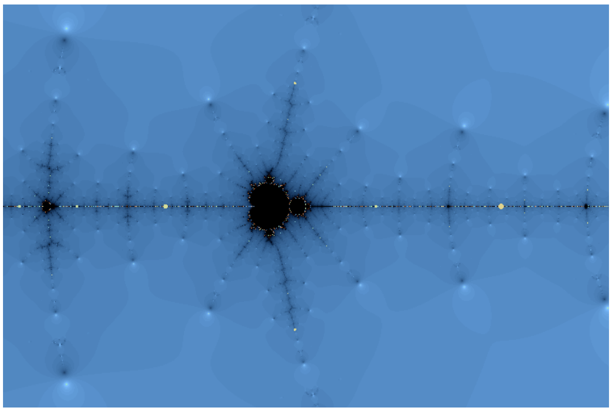
There is a green spot where the purple regions meet and it is very much smaller than before.

The previous example of a Nova 1 fractal had no Mandelbrot Islands, here's an other example which does have Mandelbrot Islands and in which there there is a large hollowed out Mandelbrot shape:

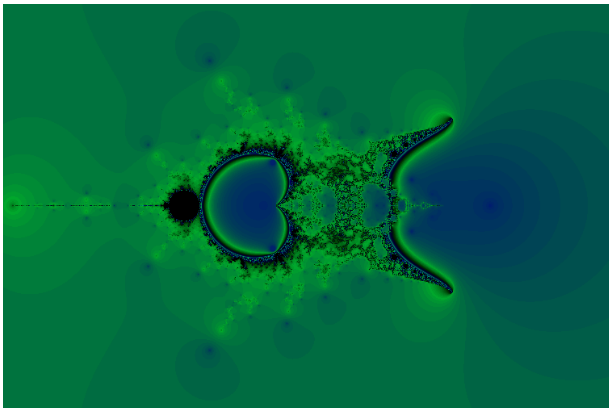


f(z) = 0.0333333z6 – 0.5z2 – 1
f'(z) = 0.1999998z5 – z
z_0 = 1

A hollowed out Mandelbrot shape.

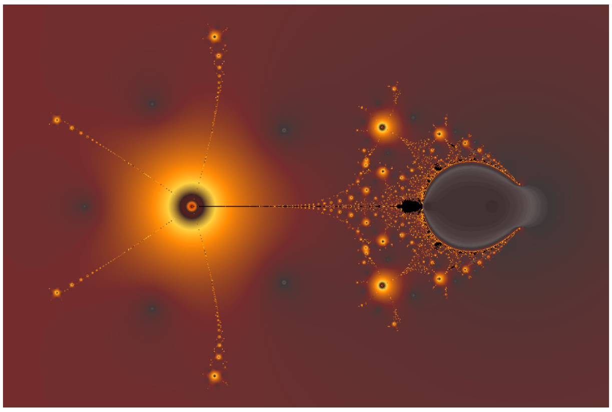


Mandelbrot Islands from the needle of the previous hollowed out Mandelbrot.

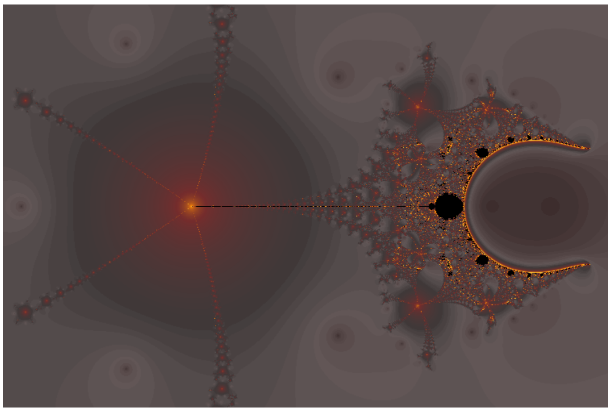


Nova 2
f(z) = z3 - (c - 1)z2 - c
f'(z) = 3z2 - 2(c – 1)z
z_0 = 1

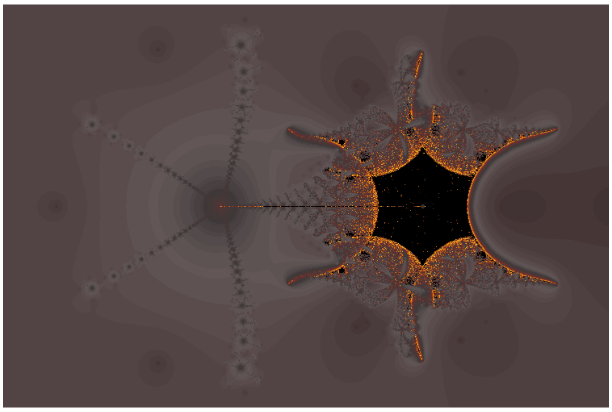
So far the value of α has been fixed at 1. Changing α alters the appearance of Nova fractals here are some examples of the Generalised Nova with various values for α.



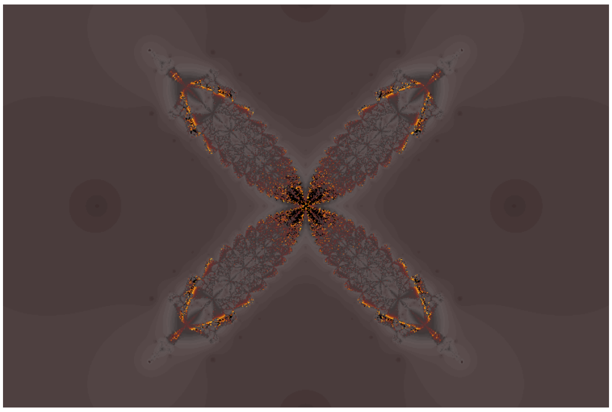
$f(z) = z^6 - 1$
$f'(z) = 6z^5$
$z_0 = 1$
$\alpha = 0.5$



$\alpha = 1.5$



$\alpha = 2.5$



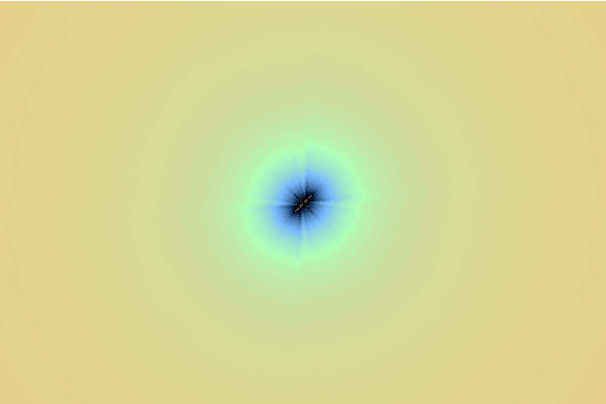
$z_0 = c$
$\alpha = 2$

3.10.5 PP Mandelbrots and Julias

These fractals are the Mandelbrot and Julia versions of the formulae used for the Pickover Popcorn fractals and instead of plotting the orbits the formulae are used with the Mandelbrot and Julia algorithms.

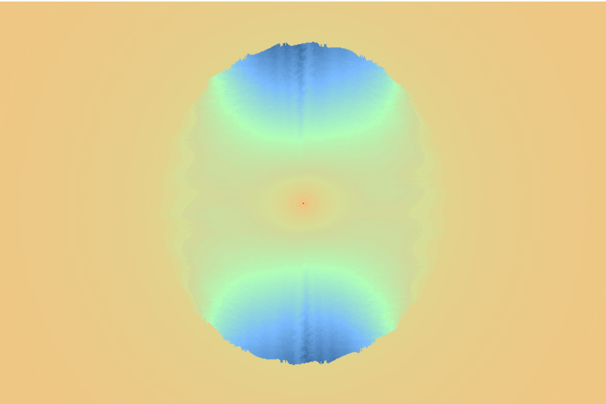
These formula produce the sort of spirals that are commonly called gnarls, however some guidance is required to produce them.

The bailout condition in all the examples is norm(z) > limit.



Default settings and starting position.
Limit = 16
z_0 = untransformed c
No gnarls.

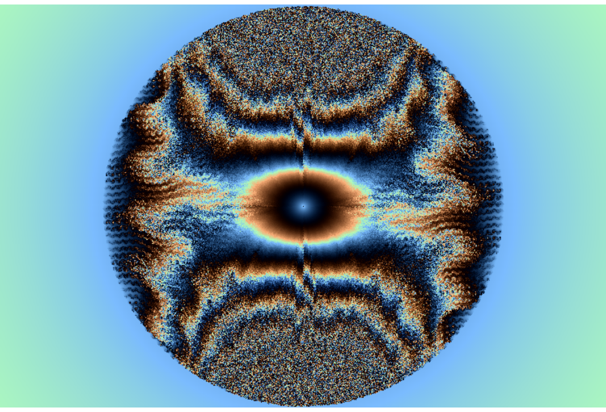
Outer colouring: iteration.



Width increased to 16.
Complex plane transform: power, -1

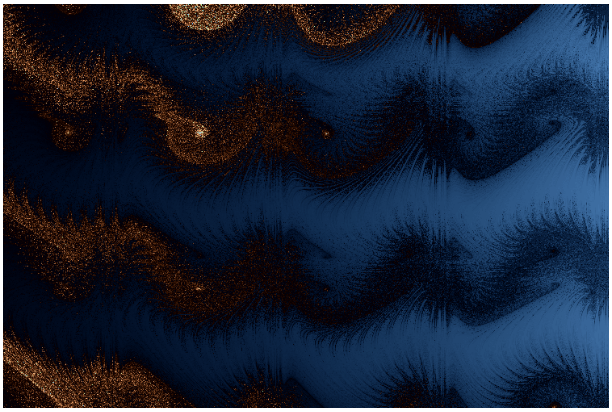
Unusually the complex plane is transformed but the transformed values ARE NOT used for the z_0.

Still no gnarls.



Limit increased to 700
Width increased to 80

Gnarls are beginning to form.

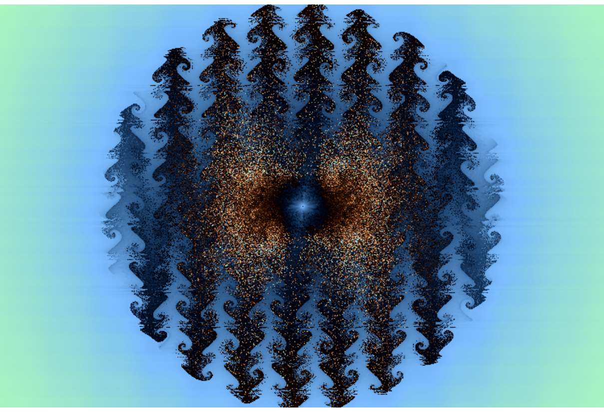


Zooming in on the right hand edge of the disc.

Gnarls are visible as is an odd criss-cross structure that disrupts the image which is characteristic of the PP Mandelbrot.

It can be difficult getting a good image with these fractals, you have to try many colour methods and adjust their options such as number of divisions and whether absolute log is enabled. Greatly increasing the resultion using Titan can clean up an image immensely.

The default values used for the fractals came from the Pickover Popcorn orbit fractals and turn out not be that good for these fractals.



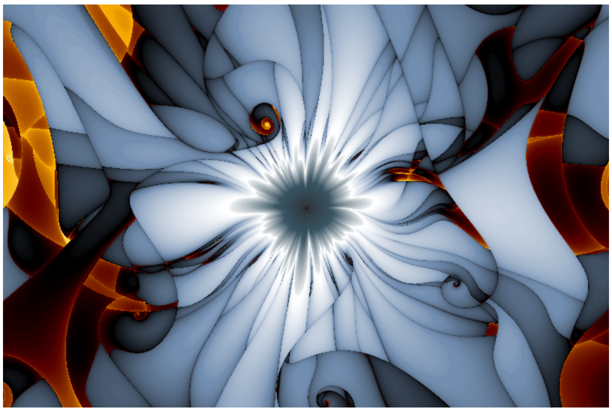
This is the same PP Mandelbrot fractal disc as before with the parameter values changed.
$\alpha = 0.5 + 0.7i$
$\beta = 0.7 + 0i$

Much more gnarly.

PP Mandelbrot 4 same location and parameters as the picture above.

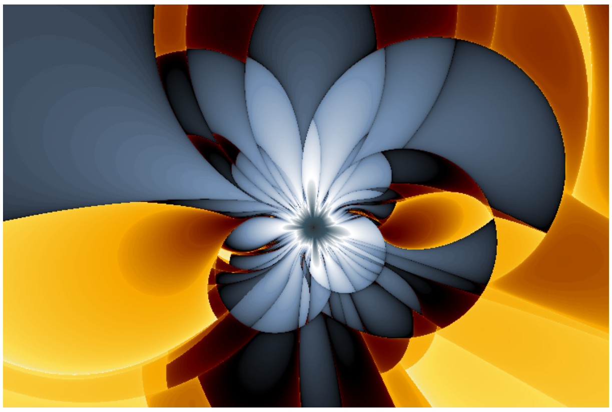
Colour method is still iteration but uses a different colour map.

Note: the odd criss-cross pattern is absent in this variation of the PP Mandelbrot.

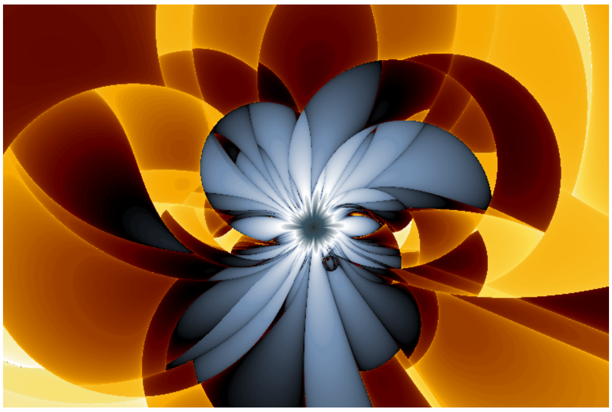


Zoom in on the central part of the previous picture.

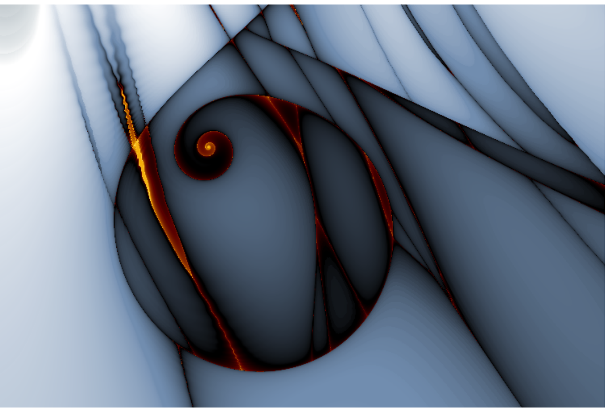
Using a fixed value for z_0 produces some interesting pictures where gnarls are mostly absent.



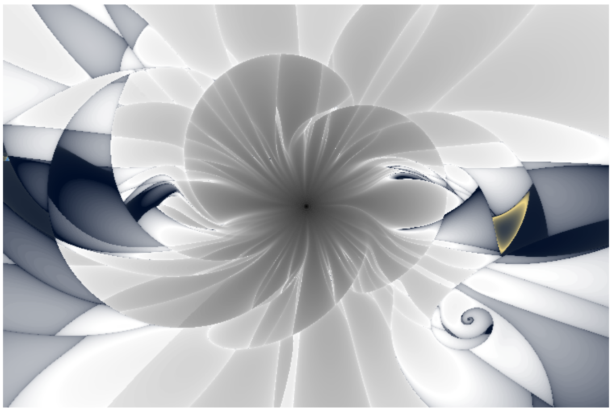
The same settings are the previous picture except that z_0 is set to 3.



$z_0 = 1$



Zooming in on the circular structure just to the the right and below the centre we find a gnarl.

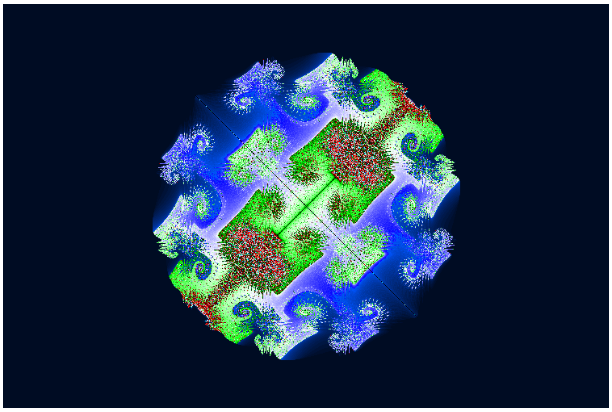


Finally for PP Mandelbrots z_0 set to the transformed complex plane.

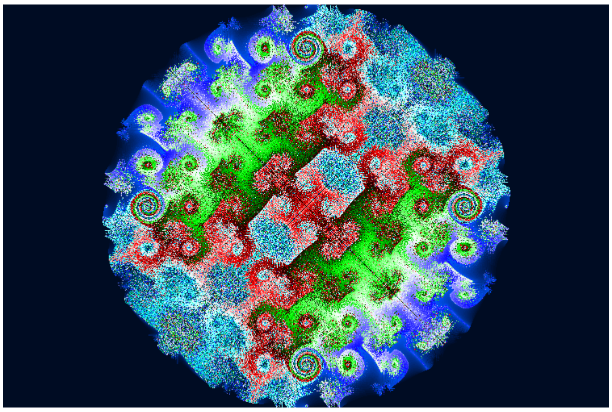
$\alpha = 0.5 + 0.7i$
$\beta = 0.7 + 0i$

It may be possible to find setting for α & β that give good result with no complex plane transform, further exploration is required.

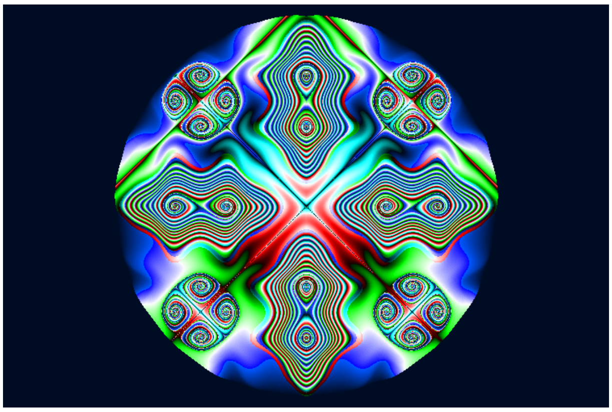
Now for the Julias. PP Julias are sensitive to the the value of the bailout limit, the larger the limit the larger and usually more intricate the fractal disc.



PP Julia
$\alpha = 0.05 + 0.05i$
$\beta = 3 + 3i$
$\gamma = 0 + 0i$
Bailout:norm(z) > 4

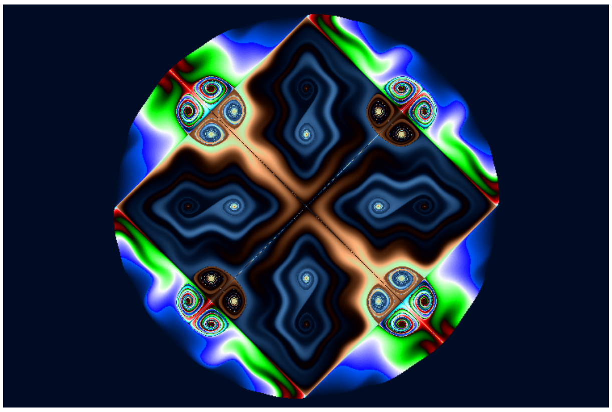


PP Julia with default settings, no transforms.
PP Julia
$\alpha = 0.05 + 0.05i$
$\beta = 3 + 3i$
$\gamma = 0 + 0i$
Bailout:norm(z) > 16



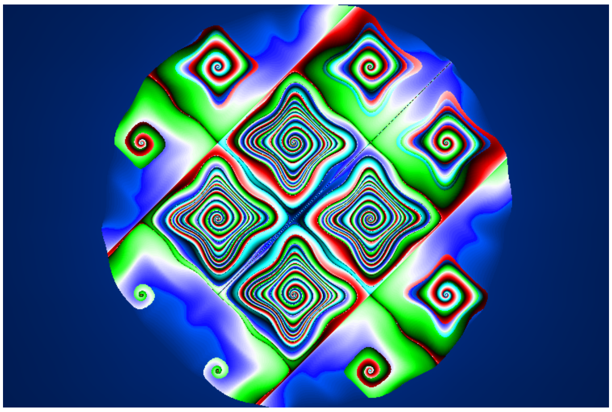
PP Julia
$\alpha = 0.05 + 0.05i$
$\beta = 3 + 3i$
$\gamma = 0 + 0i$
Bailout:norm(z) > 32
Iterations: 25000 there is a clear indication that the central area will be to filled at 100 iterations. All iterations are calculated.

As the value of the limit goes up it is increasingly likely that portions of the fractal take a very long time to escape or never escape at all so inner colouring is required.



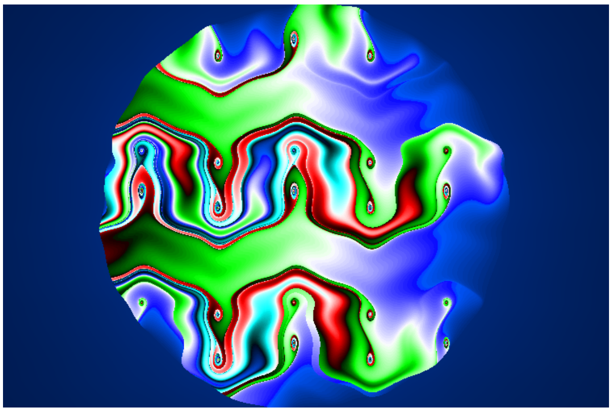
The same as the previous picture this time with a limit of 40.

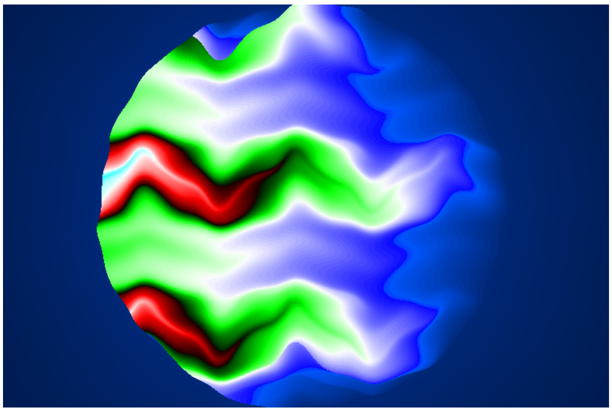
Inner: absolute log of average magnitude. Iterations: 500 as there is no indication the that the central area will be filled with outer colour.



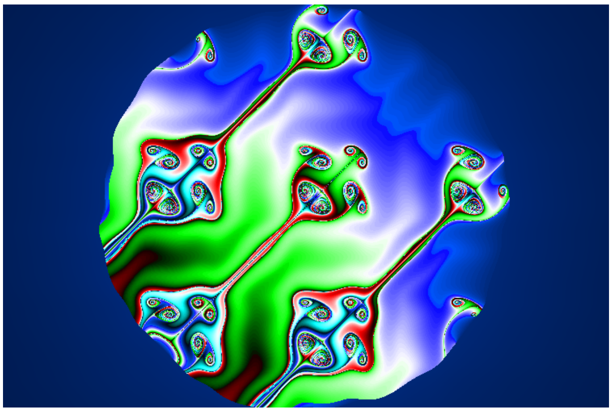
PP Julia 3
Bailout:norm(z) > 70
$\alpha = 0.125 + 0.125i$
$\beta = 2 + 2i$
$\gamma = 0 + 0i$

So far the Julias have also used γ set to zero. Care must be taken when setting γ (δ for PP Julia 5) as the addition of the constant has a tendency to push anything of interest out of the circle.





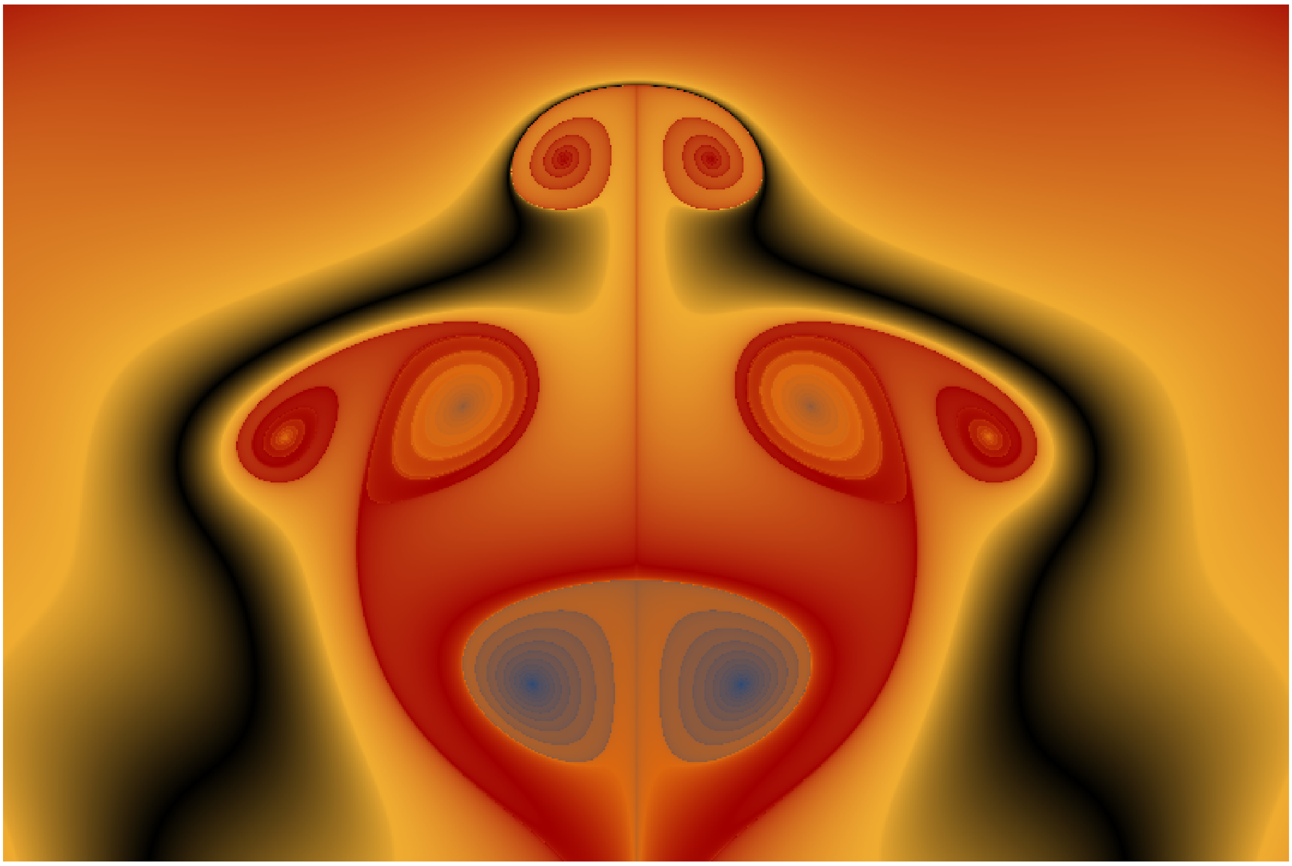
These use the same settings as the PP Julia 3 with $\gamma = 0.1 + 0i$ on the left and $\gamma = 0.2 + 0i$ on the right.



$\gamma = 0.1 + 0.1i$

Below: a zoom of this rotated 45° and the colour map and method changed. Outer colouring method: absolute log of exponential sum of magnitude.

Available as a parameter file.



PP.Julia.spf

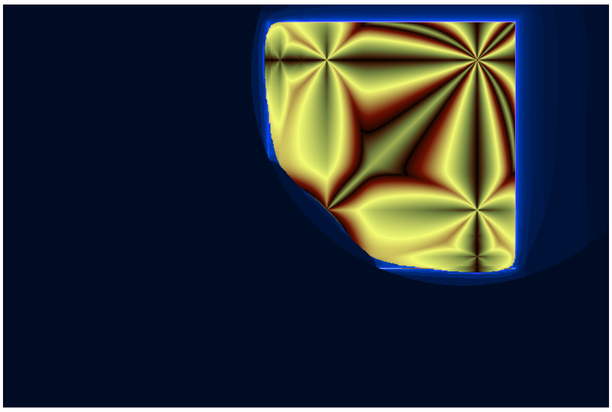
3.10.6 Real Number Mandelbrot and Julia

These fractals are based on formulae that do not use complex numbers. The formula for RN Mandelbrot:

$z = \text{transform}(z)$
$z_r = z_r^2 + c_r$
$z_i + z_i^2 + c_i$

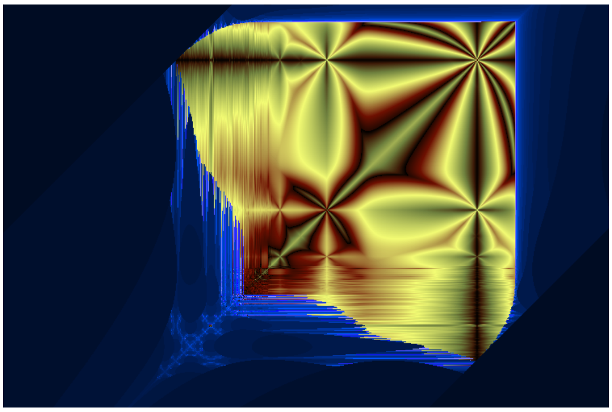
The Julia forumula differs in that α replaces c.

The generation of pictures for these fractals is sensitive to the type of bailout condition and limit used. The colouring used for the examples is interation for outer colouring and average ratio, smaller of real(z), imag(z) divided by larger of real(z), imag(z). The number of iterations is 40.



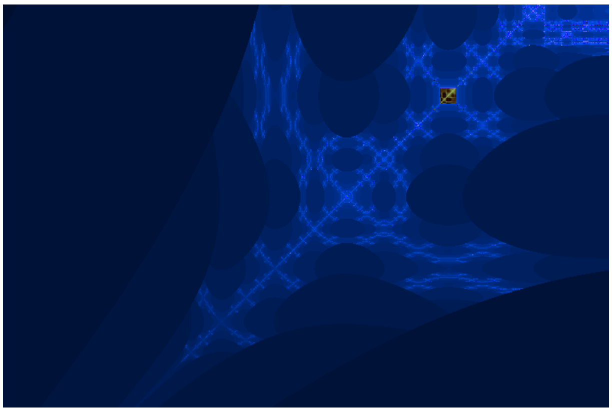
Bailout condition of abs(z) > 2.

Not that interesting. Most bailout conditions result in similarly boring images.



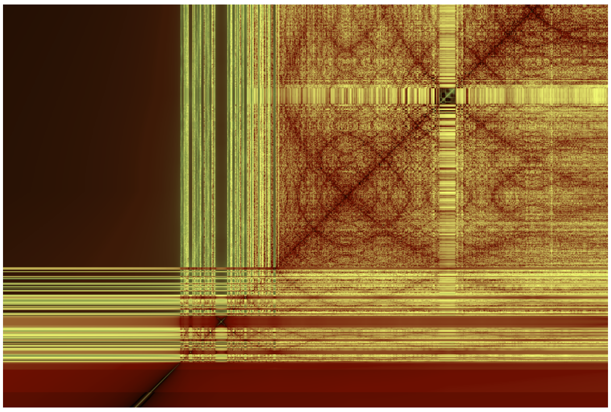
Bailout condition abs(real(z) – imag(z)) > 2.

Fractal features appear at the bottom left hand corner.



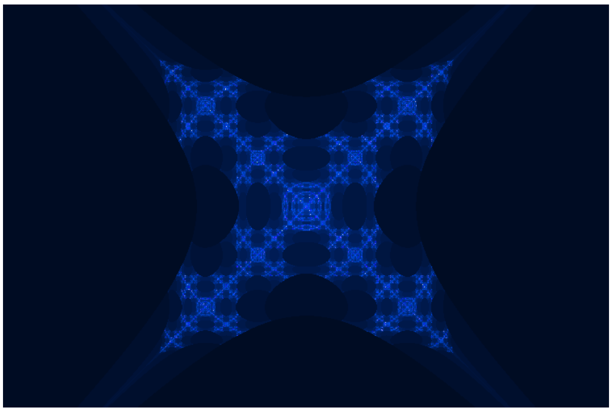
Zoom in of the bottom left hand corner.

The value for the bailout limit can be varied, the lower the value the fractal features are less dense, the higher the value results in more of the bottom left hand corner being coloured as inner. The inner colouring method reveals fractal features as well.

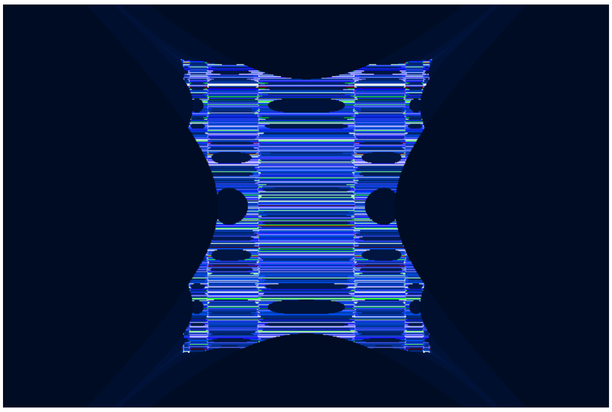


No bailout, all points are inner, this is the same area as the previous image.

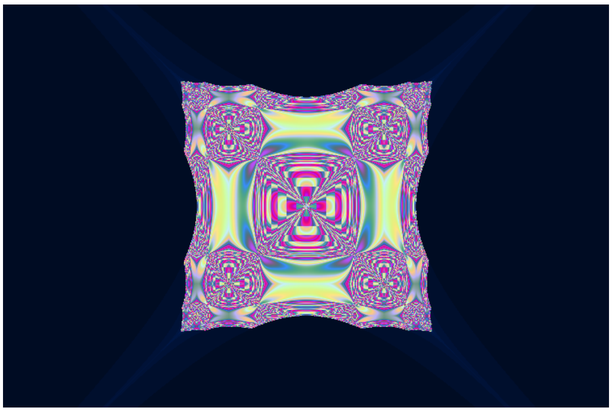
The Julias are sensitive to the value of α, values where $\alpha_r = \alpha_i$ work best when they are negative. The examples use a bailout condition of abs(real(z) – imag(z)) > 2.



α = -1.7 - 1.7i.



α = -1 - 1.7i.



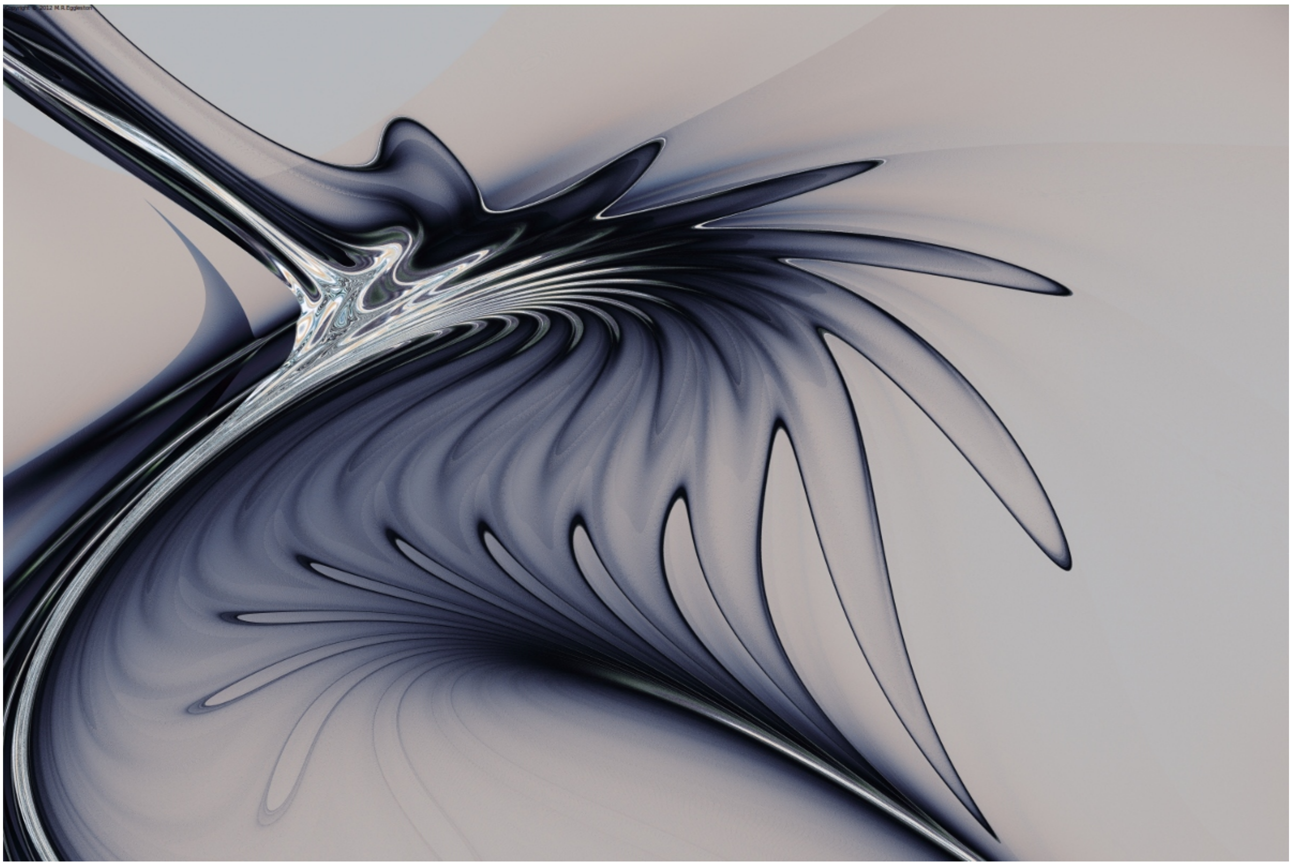
α = -1 - 1i.

Outer : iteration
Inner: exponential inverse change sum of ratio, smaller of real(z), imag(z) divided by larger of real(z), imag(z).

4 Orbit Plotted Fractals

In versions 1.0.x and 1.1.x of Saturn the only “orbit plotted” fractals were the Pickover Popcorn fractals. For version 2.0.x an option has been added to all escape time fractals as a result the Pickover Popcorn fractals have been removed as they are the same as PP Julia fractals with the orbit plotting option enabled all be it with addition of a extra parameter (set to zero to emulate the old Pickover Popcorn fractals).

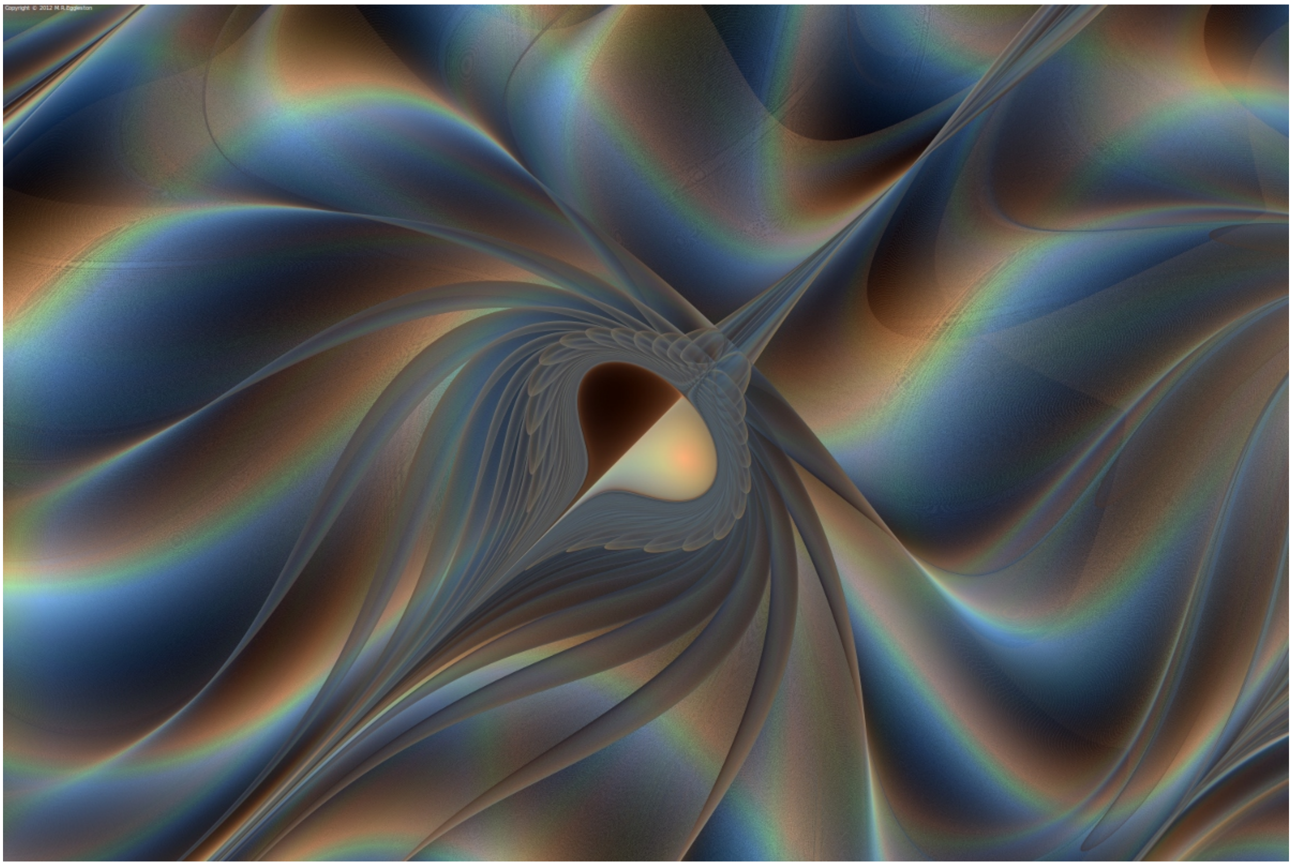
Now for some examples:



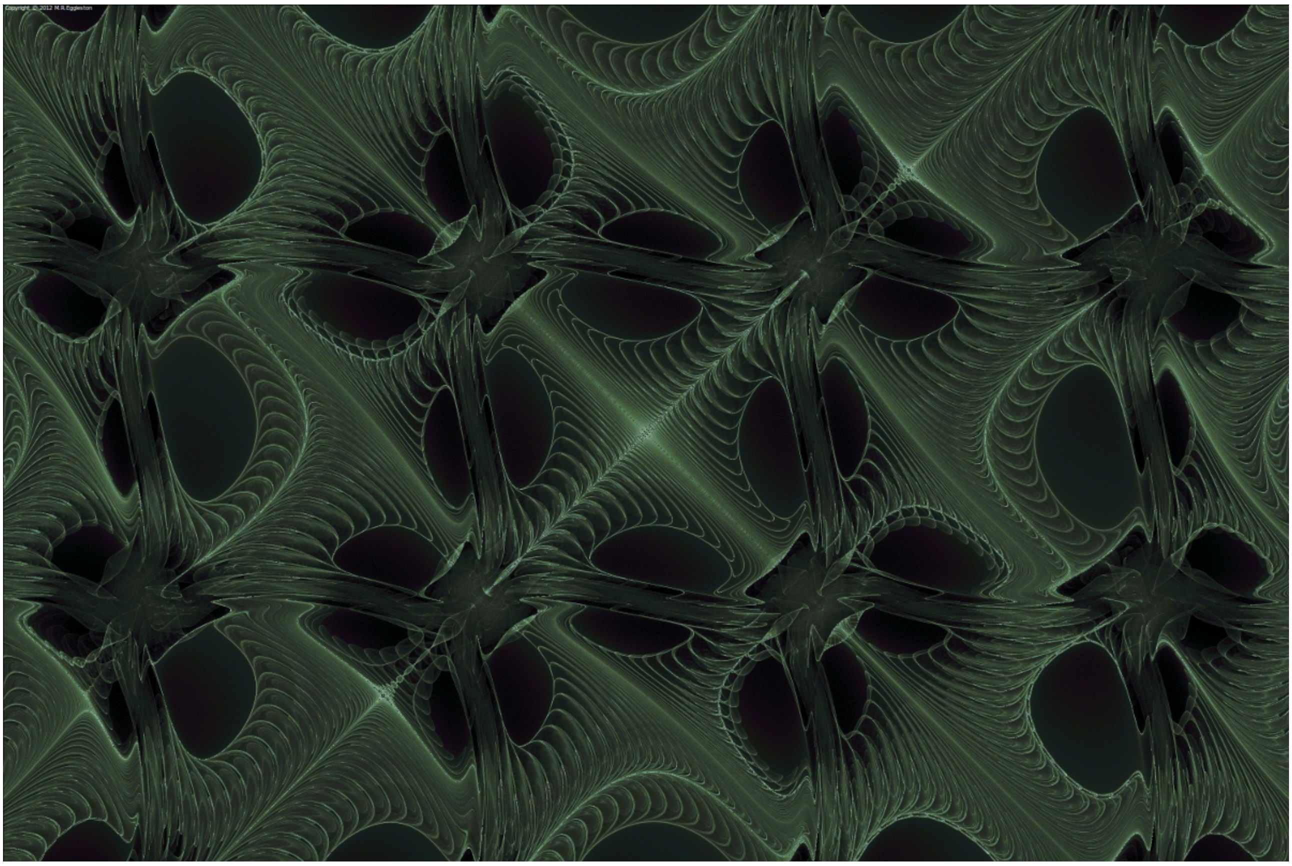
Flow.of.Ink.spf

The picture above is a PP 6F Julia fractal.

PP Julias in their 4 function and 6 function forms produce a wide variety of interesting orbit plotted images, apparently PP Mandelbrots do not, I least I've not yet found a good combination of parameters.

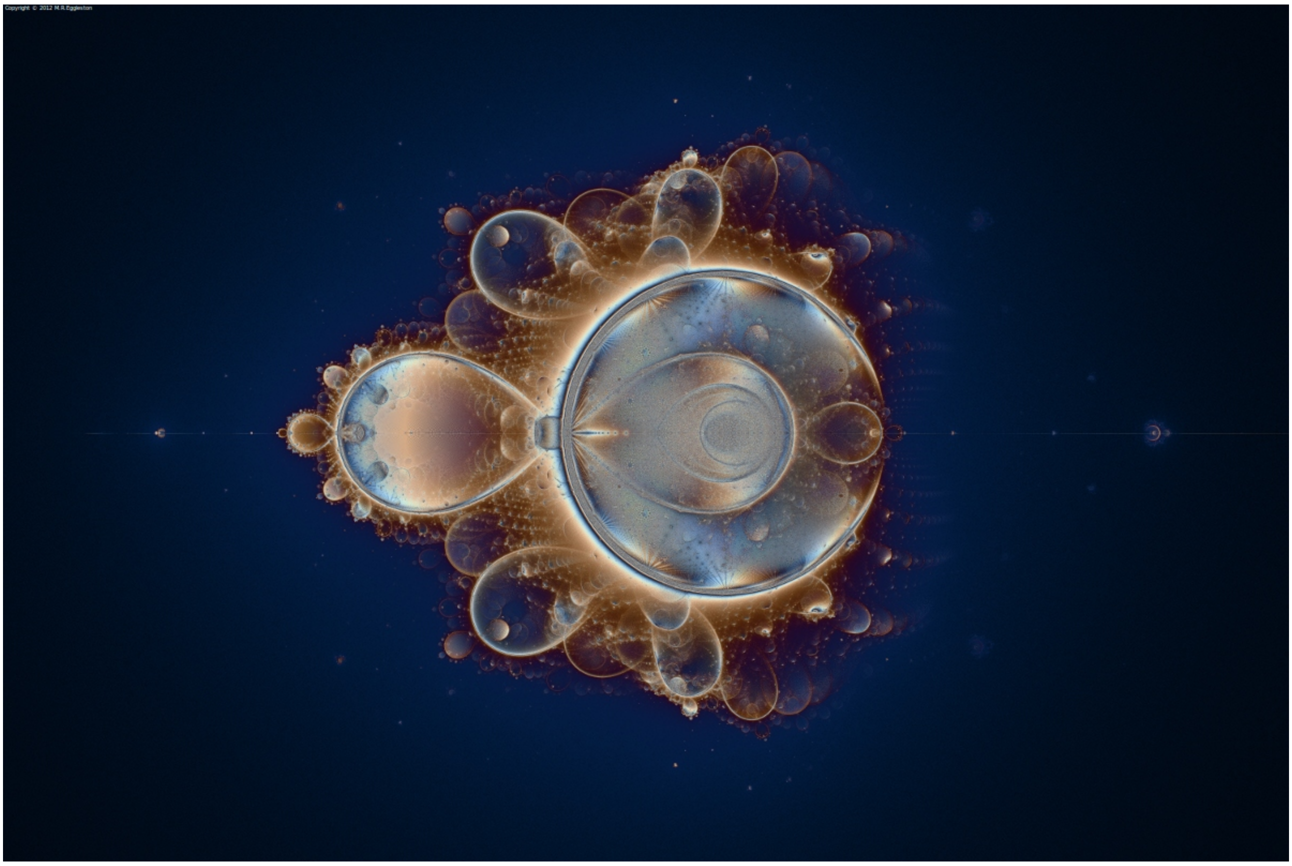


The above picture is a PP Julia 4F (Ruff.spf).



The above picture is also a PP Julia 4F (Popcorn.In.Green.spf).

There are three type of orbit plot, all orbits, escaped orbits or captive orbits. Pickover Popcorn type orbit plots usually have no bailout so are plotted using all orbits (orbits) or captive orbits. The PP Julia formula is an adaptation of the original Pickover Popcorn formulae which specifically designed so that the distance between each member of an orbit is small, for other escape time fractals this is not the case especially with divergent fractals such as the well known Mandelbrot set.

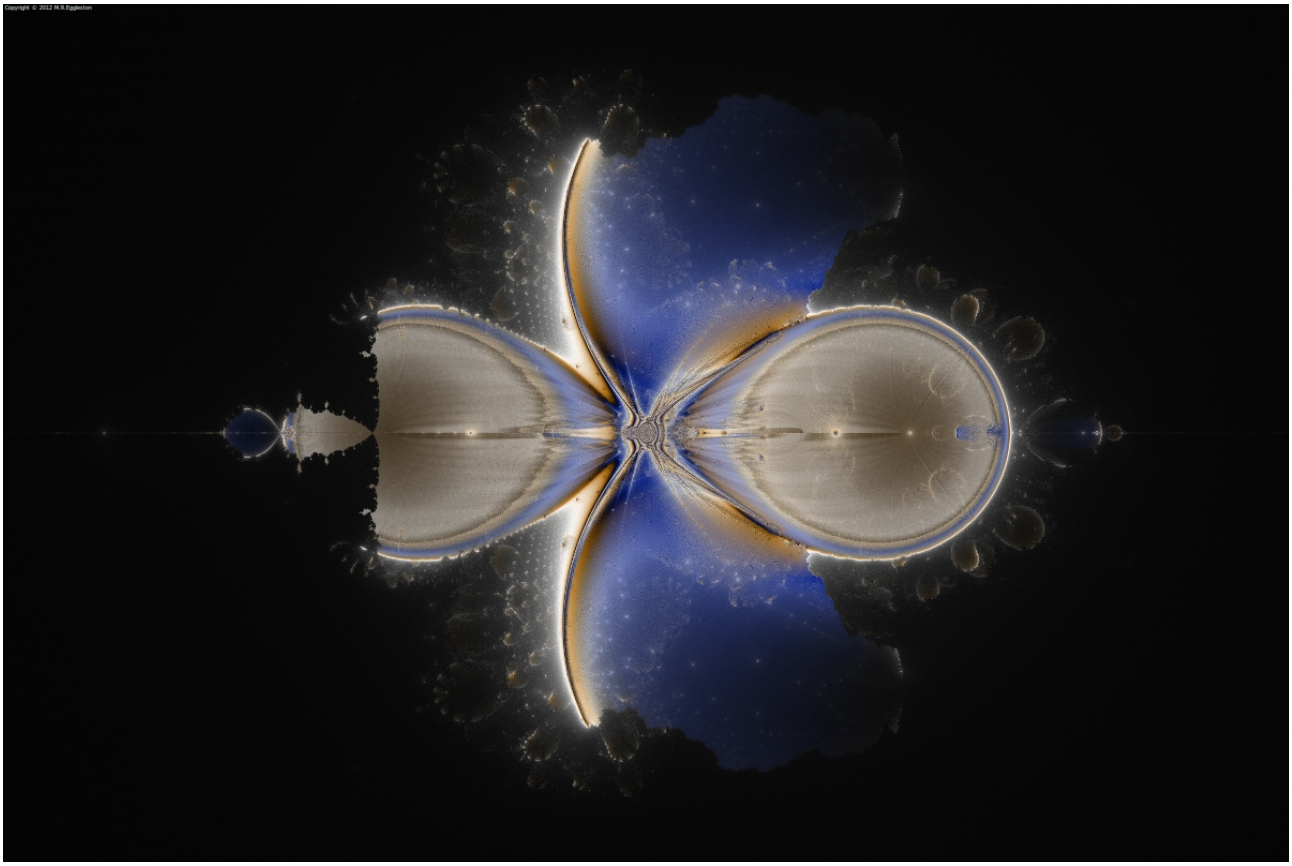


Orbit.Mandelbrot.spf

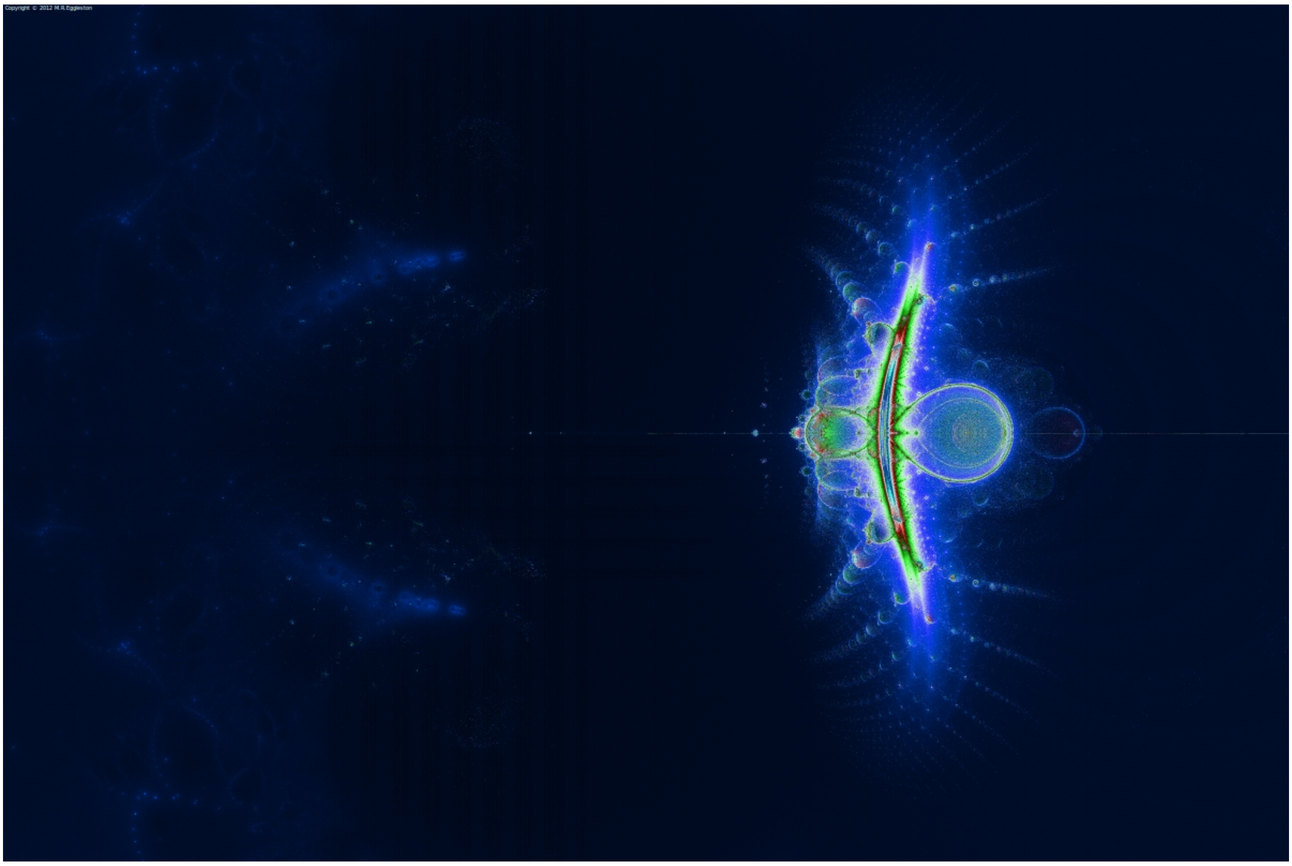
An orbit plotted Mandelbrot, all orbits (Orbit Mandelbrot). Note: this picture was produced using an early development version of version 2.0 so the parameter file will not produce exactly the same image). This is very similar to the picture produced using only captive orbits which is also known as the Anti-Buddhabrot although thos are produced using an algorithm that employs random numbers.

The pictures produced using only the escaped orbits (Buddhabrot) so far have been poor, they can be improved by greatly increasing the orbit length which in turn greatly increases the time required to generate the images.

One aspect of orbit plotted fractals is that zooming into an image is limited as detail is lost as orbits that would have made up an image are never calculated as the originate outside the image, often the result is a black screen. In order to compensate for loss of detail the area used to calculate the orbit plotted fractal is larger than the are displayed, when only Pickover Popcorn was orbit plotted an area 9 times the display allowed for a reasonable depth of zoom and the size of the calculating area was fixed. For fractals other than Pickover Popcorn a calculating area of 9 times the display area was inadequate. For version 2.0 the size of the calculating area can be anything up to 1000 times the display area production of good quality images becomes impractical as the number of calculations becomes enormous as unlike Pickover Popcorn which typically has orbit lengths of around 50 the orbit length required for other types can run into thousands and even larger if only escape orbits are used.



Orbit.Compasses.spf



Orbit.Nova.spf

5

The appearance of orbit plotted fractals when they have been expanded to much higher resolutions may not feature some of the structures that appear to be present at lower resolutions, Pickover Popcorn orbit fractals do not appear to be as prone to this phenomena as the fractal calculated using the Mandelbrot algorithm.

I've only tried orbit plotting a small subset of the escape time fractals, so far with the exception of PP Julia fractals the images produced using the Julia algorithm are of little interest, conversely the images produced using the Mandelbrot algorithm usually produce something of note with the exception of the PP Mandelbrot fractals which are poor.

The exploration of orbit plotted fractals appears to be in its infancy as the number of fractal types it has been attempted with is limited due to the difficulty in getting decent deep of zoom in a reasonable amount of time. As Saturn has made available orbit plotting to all its escape time fractals exploration of this unpopular fractal by way is encouraged.

6 Formula Transforms

Saturn can apply transforms to all of the escape time fractals. The transforms can be applied to the complex plane or to the formula. The transforms applied to the formula can be applied in two sets, only one set is applied to the formula for a given iteration, the sequence in which the sets are applied is controlled by a sequence string.

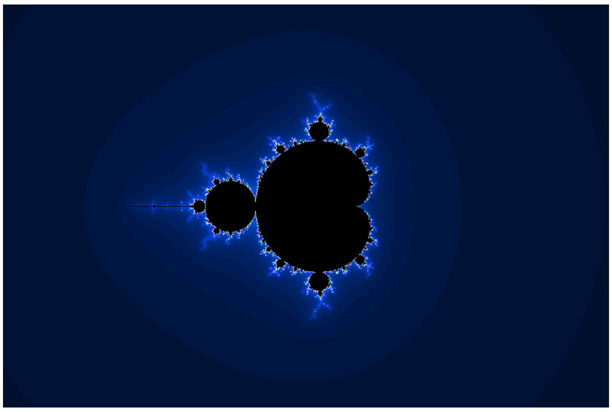
This section illustrates the use of various transforms on their own, in combination and applied periodically.

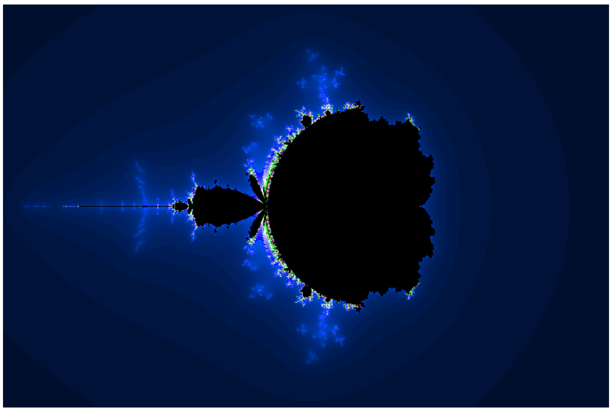
The Mandelbrot set is probably the best known fractal of them all. The Quadratic fractal has an extra term in its formula it is a more general form and it parameters can be set so that the Quadratic formula is produced. The formula are:

$$z \leftarrow z^2 + c$$

$$z \leftarrow \alpha z^2 + \beta z + c$$

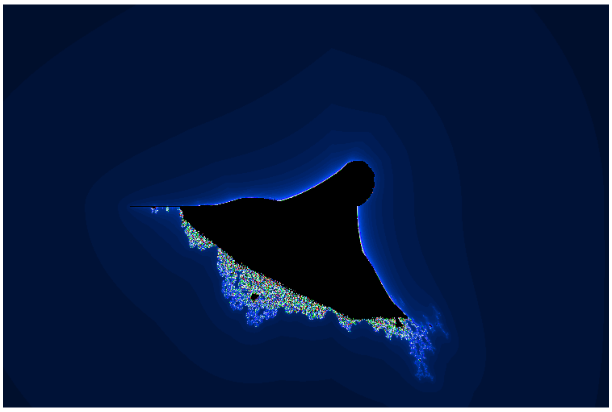
The first formula is for Mandelbrot and the second for Quadratic, so if α is 1 and β is 0 the Quadratic formula becomes the Mandelbrot formula.



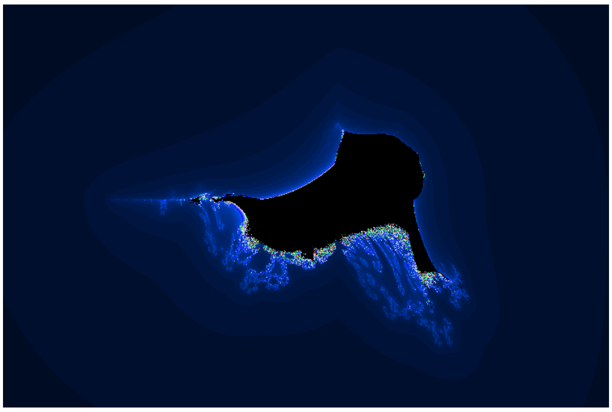


Mandelbot and Quadratic with $\alpha = 1$ and $\beta = 1$.

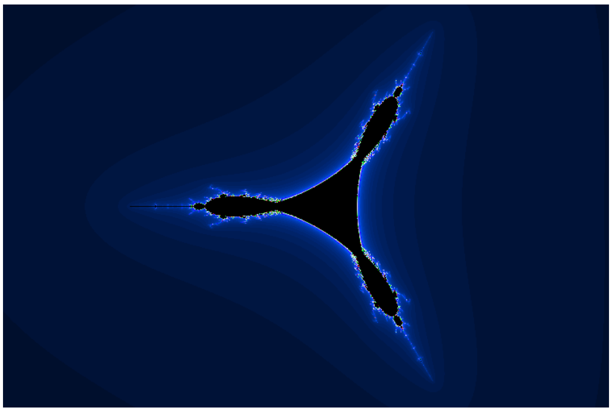
These fractals can have transforms applied to them to produce a variety of other well known named fractal types:



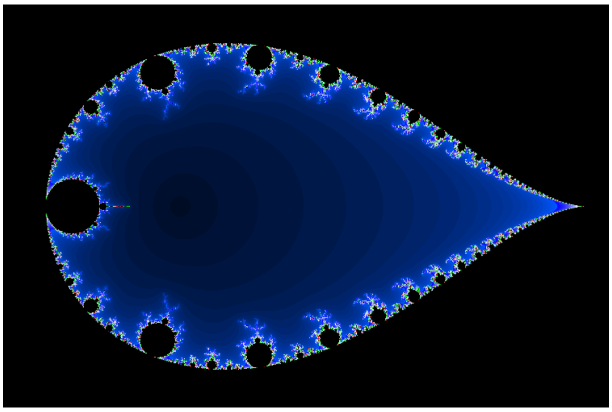
Burning Ship, Mandelbrot with Top Right transform.



Buffalo, Quadratic with α i= 1 and β = -1 and Top Right transform.



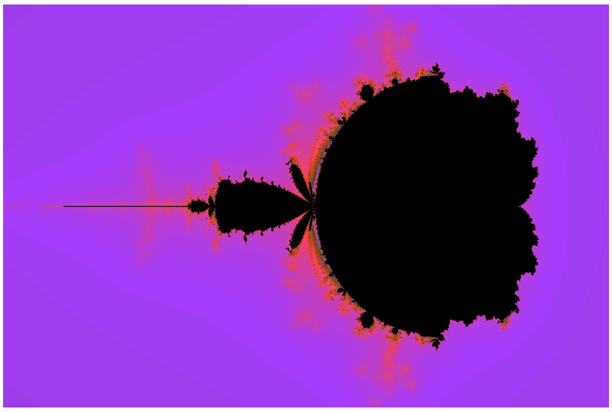
Tricorn (or Mandelbar), Mandelbrot with either Reverse Sign Imaginary or Reverse Sign Real transform.



Inverse Mandelbrot, Mandelbrot with a complex plane transform of Power, -1.

6.1 Translation

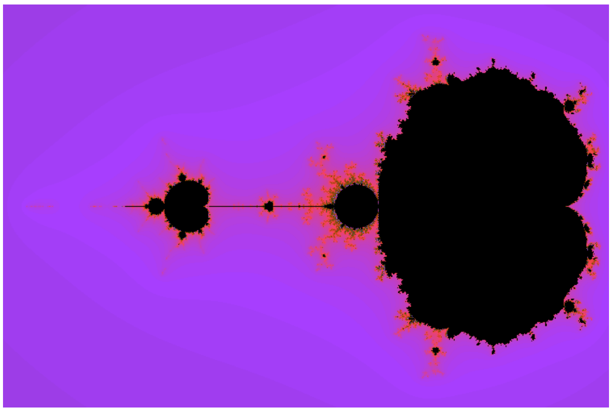
Here are a couple of simple translation transforms applied to the Mandelbrot set:



Translation transform adding 0.5 for the real component of z resulting in a image similar to the quadratic fractal, the actual formula is transformed into $z \leftarrow z^2 + z + 0.25 + c$.

$z_0 = 0$

Note: changing z_0 to the location in the complex plane produces the Mandelbrot set.

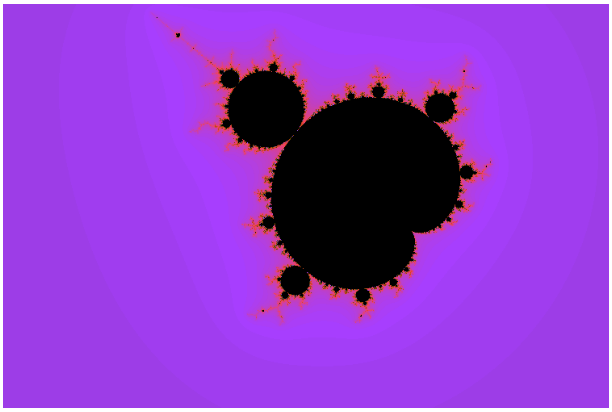


Translation transform adding 0.5 for the real component of z for Transforms A and no transforms for Transform B applied using the sequence BA.

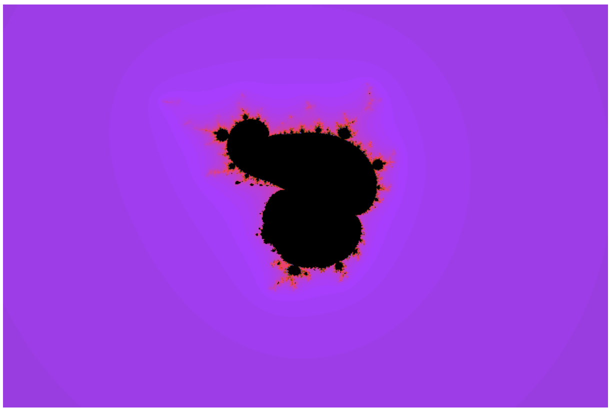
$z_0 = 0$

6.2 Rotation

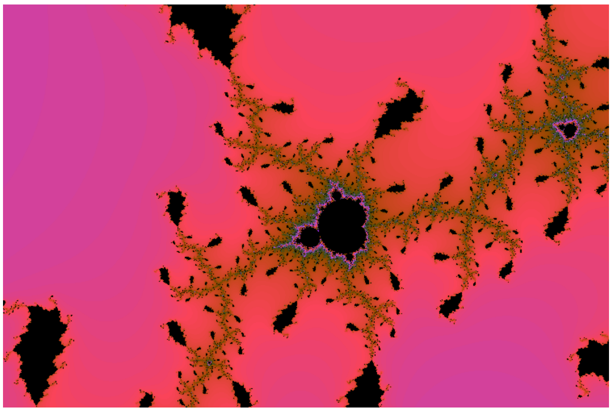
The rotation transform on its own just rotates the image, this is because the magnitude of z is unaffected. Using the transform periodically however does have an affect.



Mandelbrot with 10 degree rotation transform.



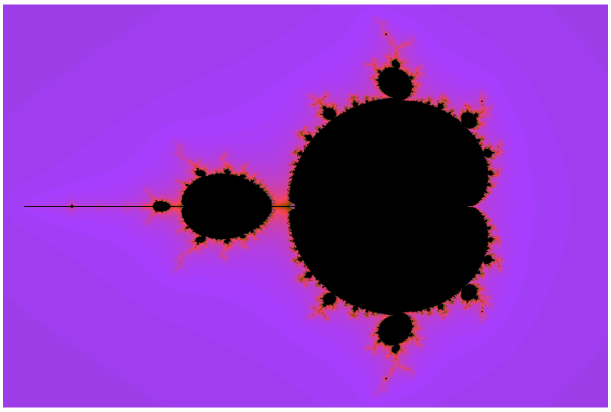
Mandelbrot with 10 degree rotation transform for Transforms A and no Transform B applied using the sequence BA.



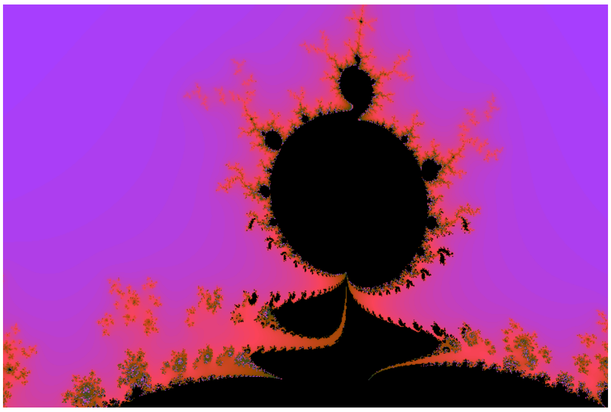
Zooming in we find distorted Mandelbrot Islands.

6.3 Scale

The scale transform doesn't do much on its own, using equal values for real and imaginary just changes the size of the image, un-equal values will distort the image, for example:



Mandelbrot, with scale transform, real = 0.875, imag = 1.0

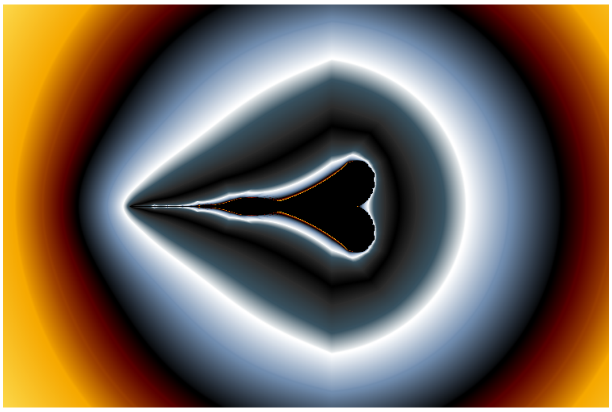


Mandelbrot, with scale transform, real = 0.875, imag = 1.0 for Transforms A, no transforms for Transforms B and a transform sequence of BBA

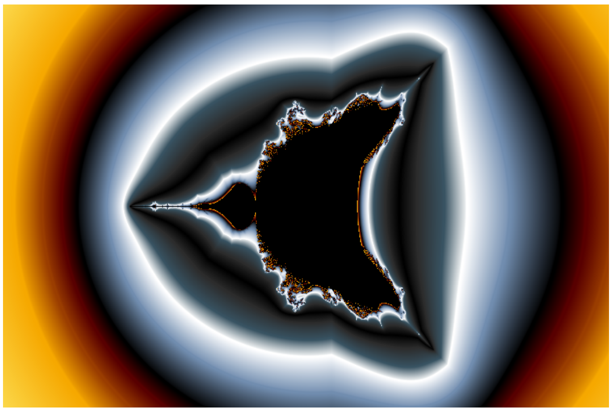
There is nothing to stop negative values being used for the real and imaginary scales, in the case of the Mandelbrot set the result is the same for both negative and both positive, for both, scaling of one positive and one negative you will get the Tricorn (or Mandelbar) fractal.

6.4 Unsign/Sign/Reverse Sign Real

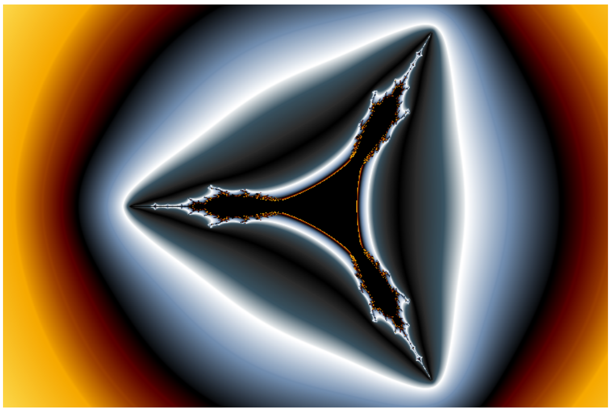
These transforms only manipulate the real component of a complex number. Applied to the Mandelbrot set we get the following:



Unsign real.



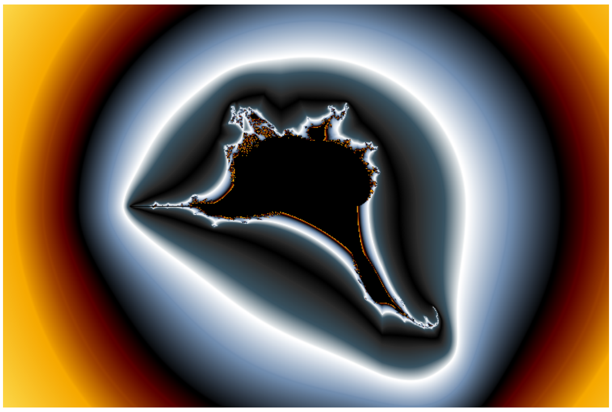
Sign real.



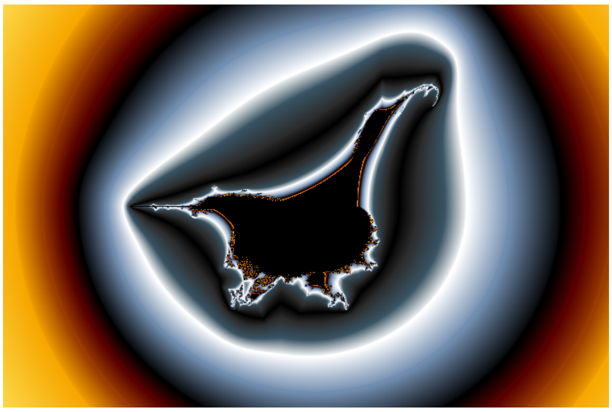
Reverse sign real.

6.5 Unsign /Sign/Reverse Sign Imaginary

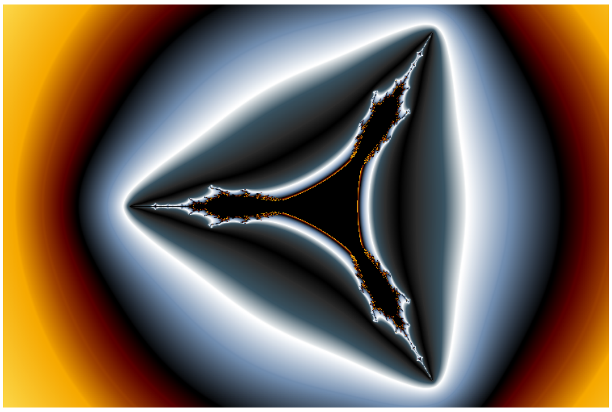
These transforms only manipulate the imaginary component of a complex number. Applied to the Mandelbrot set we get the following:



Unsign imaginary.



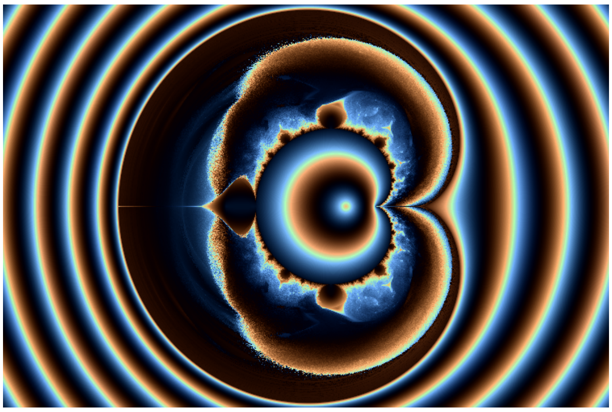
Sign imaginary which is the mirror image of unsign imaginary.



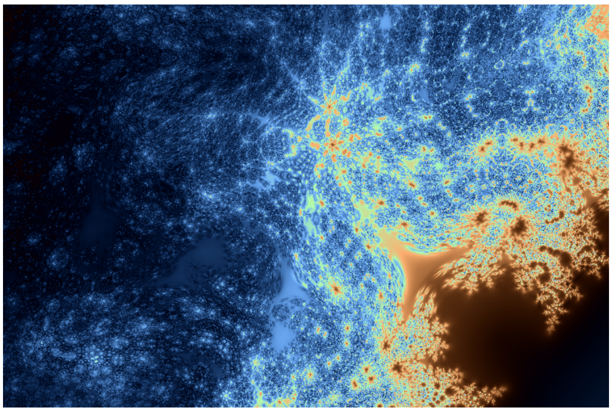
Reverse sign imaginary which is the same as reverse sign real.

6.6 Circle Fold In

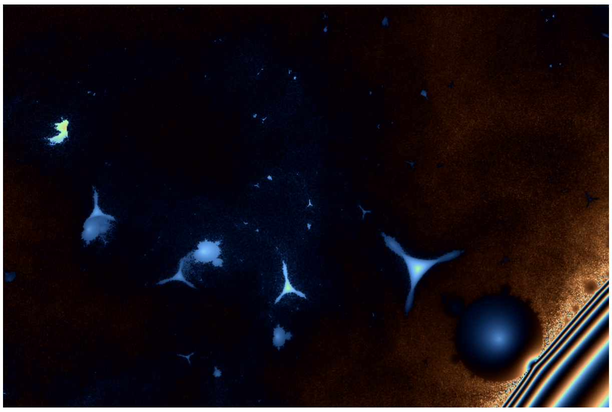
This transform folds all points outside a circle inside the circle so if it is applied to the Mandelbrot set in it default position of the origin with a diameter of 2 and black is used for inner colouring then you will get a black circle (only if you zoom out far enough). So when using this transform inner colouring such as average magnitude is required.



Average magnitude colouring with default transform settings. 500 iterations.

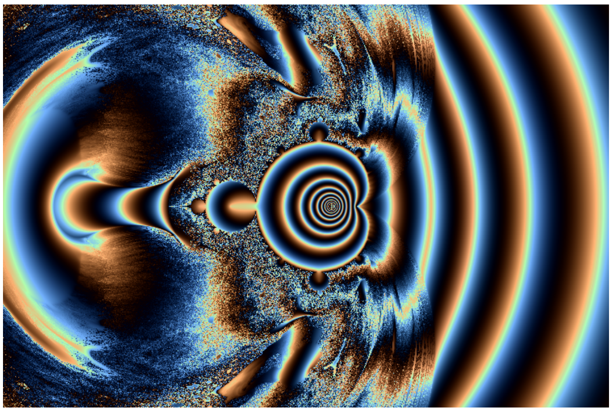


Zooming in. 50 iterations.

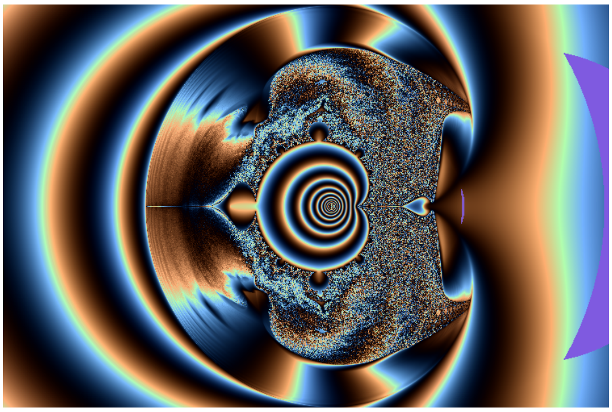


Same location, this time with coefficient of variation of magnitude colouring and 500 iterations.

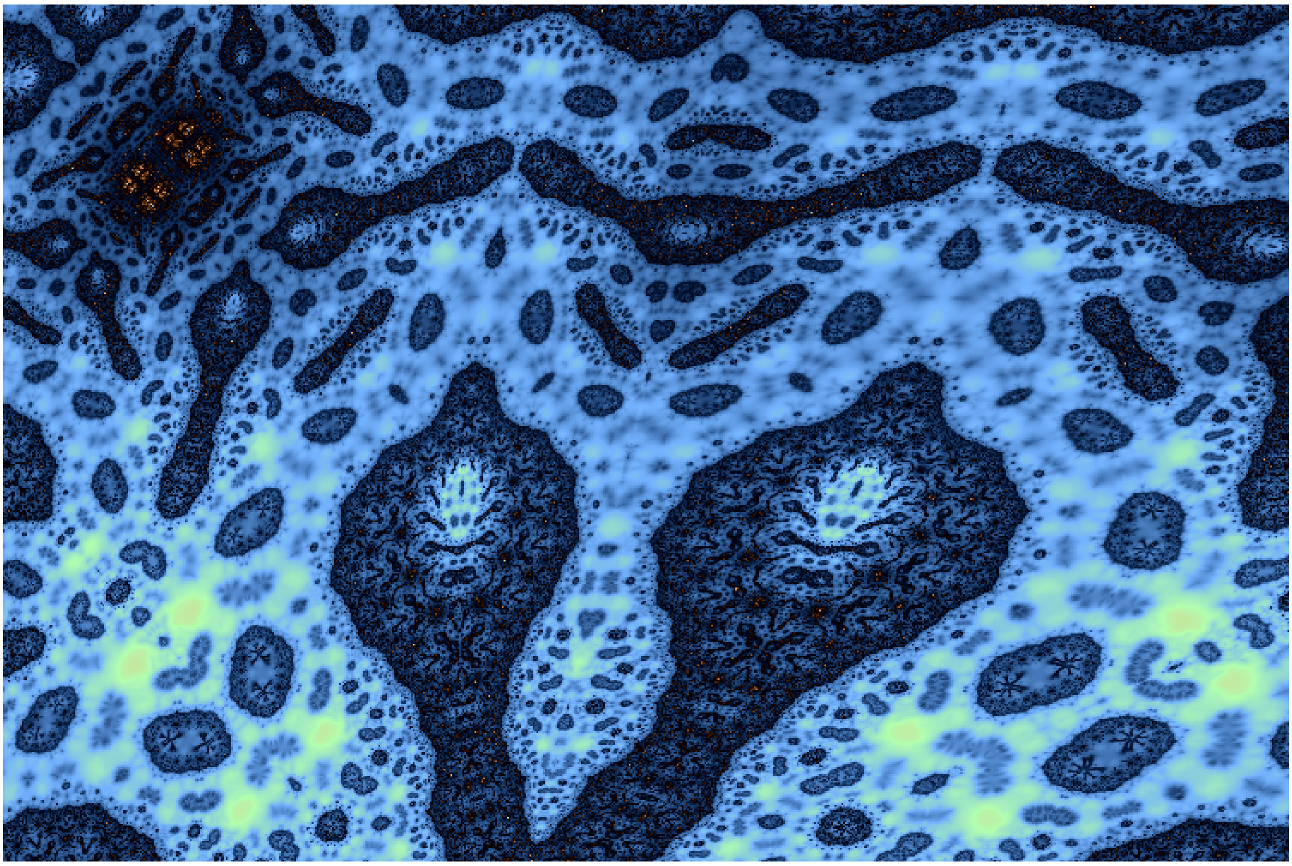
To get good images you'll need to experiment with various colour maps, colouring methods and the number of iterations..



Position of centre of circle moved to -1 + 0i and diameter changed to 3.



Position of centre of circle changed to 1 + 0i and diameter changed to 4. Note the purple crescent to the right of the picture, this is an area where the escape condition has been met, however the values calculated are not diverging so the bailout condition can be changed to “no bailout” and these “outer” areas will disappear.



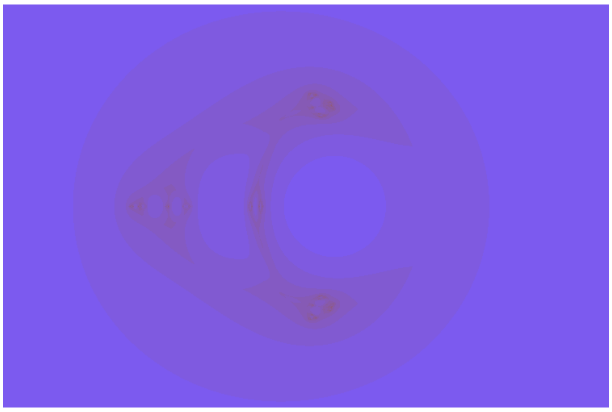
Circle.Fold.In.spf

An example parameter file for circle fold in with a circle diameter of 2.25 centred at the origin.

6.7 Circle Fold Out

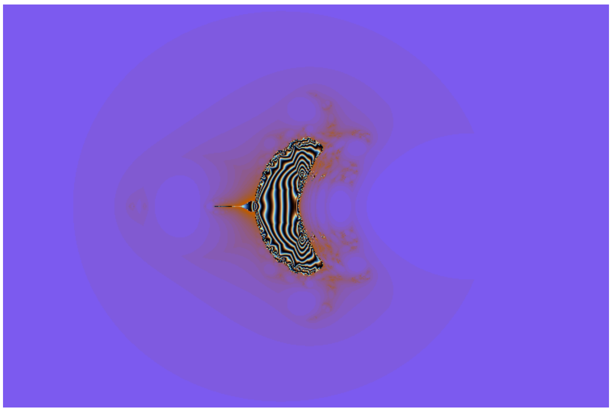
This transform folds all points inside a circle outside the circle. The initial value of z must not be zero, or if the position of the circle and the initial value results in zero when the transform is applied, e.g: if the start value is 1 and the position of the circle is also 1 the value is moved to the origin by subtracting 1 giving a position at the origin which then gets mapped to infinity before being move back by adding 1. To avoid this problem the transformed complex plane can be used to set the initial vaue of z.

So applying it to the Mandelbrot fractal we get:

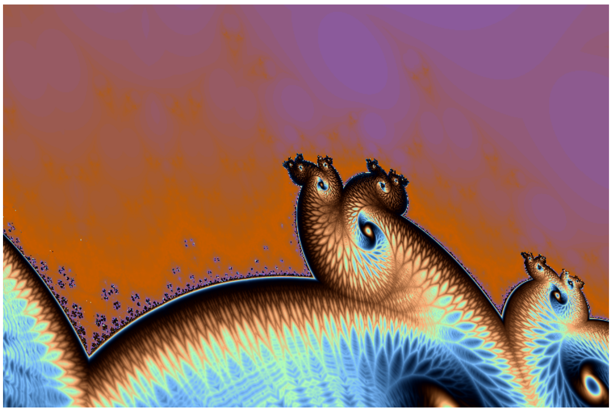


Circle of diameter 2 positioned at the origin and z_0 = transformed c.

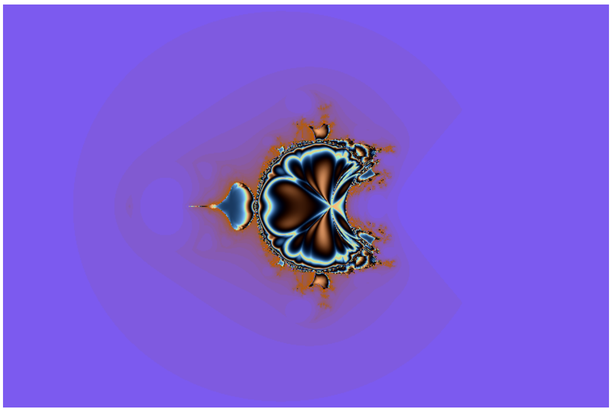
Not that interesting.



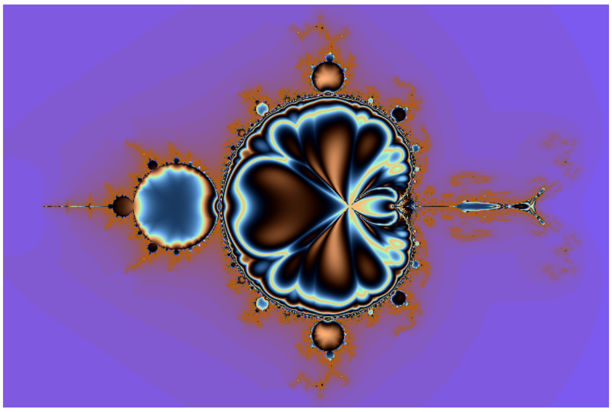
As before but with the circle positioned at 0.75 + 0i. Inner colouring is coefficient of variation of magnitude.



Zooming in at the top of the inner coloured area.

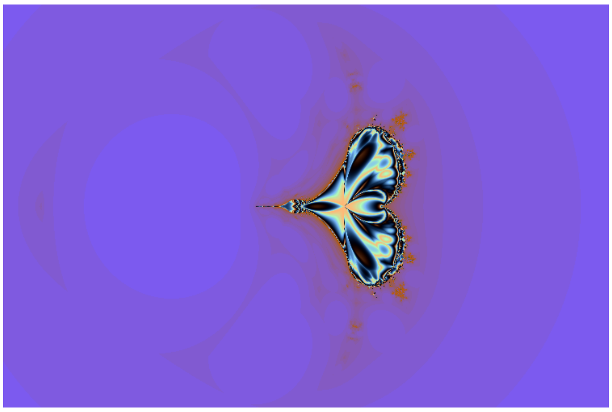


Circle positioned at 1 + 0i with fractal dimension of magnitude for inner colouring.

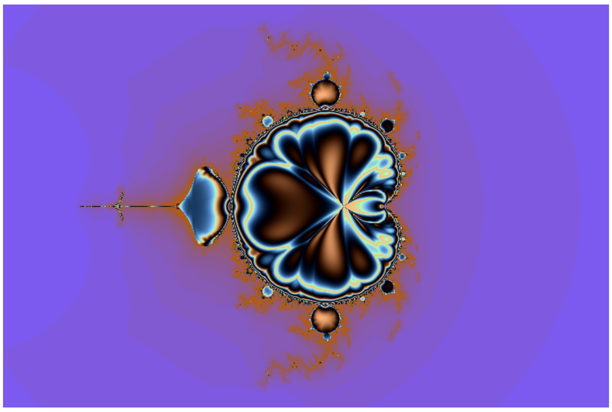


Circle positioned at 1.65 + 0i with fractal dimension of magnitude for inner colouring.

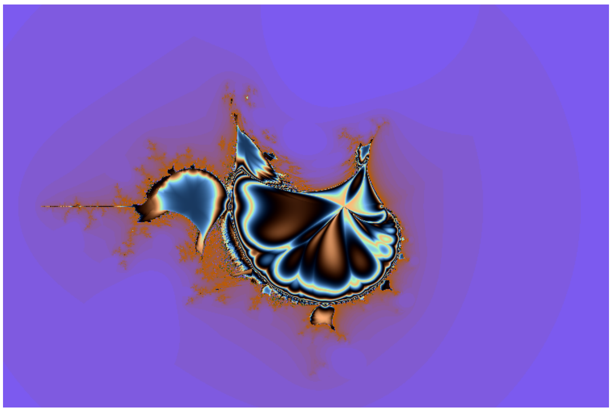
This looks pretty much like a Mandelbrot with additions, the Manelbrot needle is interesting in that where Mandelbrot islands would be Tricorns have appeared, in fact the form to the right of the Mandelbrot is an elongated Tricorn.



Circle positioned at -1 + 0i with fractal dimension of magnitude for inner colouring.

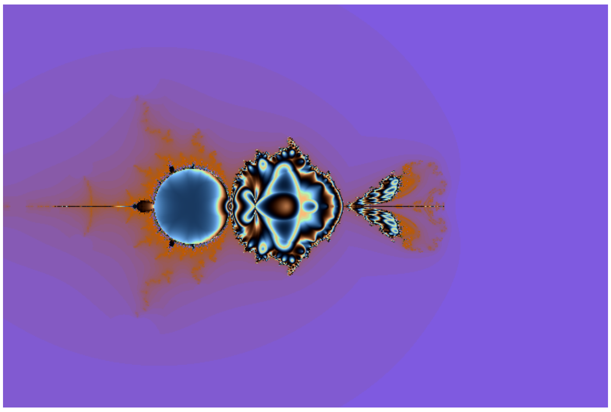


Circle positioned at -2 + 0i with fractal dimension of magnitude for inner colouring.

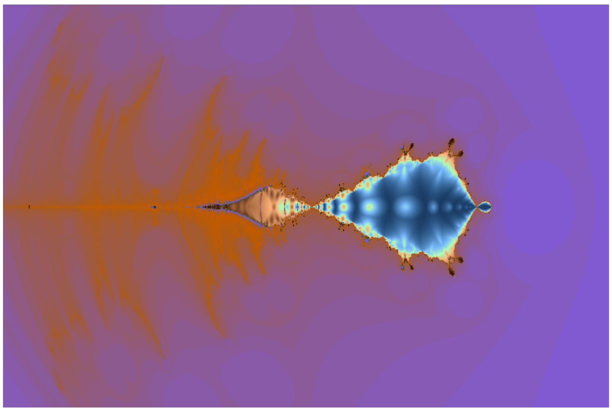


Circle positioned at 0 + 1i with fractal dimension of magnitude for inner colouring.

So far all the examples have been with z_0 = transformed c, now for some examples with



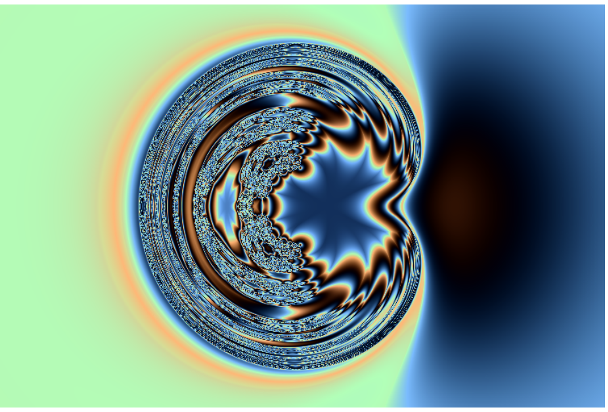
Circle position positioned at 2 + 0i with z_0 = 1 + 0i.



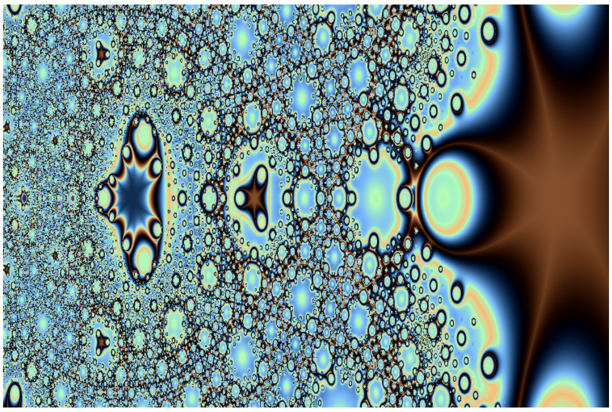
Circle position positioned at 1 + 0i with z_0 = -1.25 + 0i.

6.8 Circle Reflect

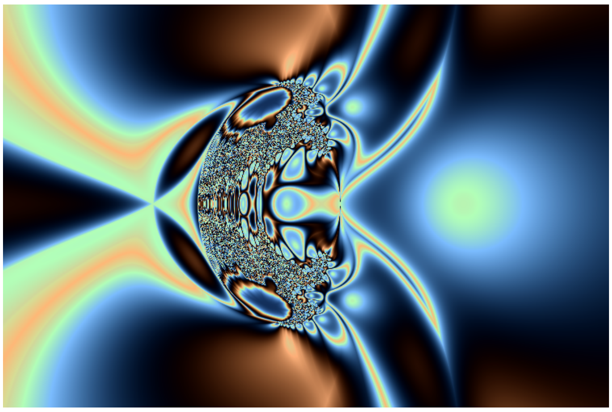
This transform maps points inside a circle outside the circle and vice versa. The effect on an orbit is that it will neither diverge or converge so the bailout condition should be set to no bailout.



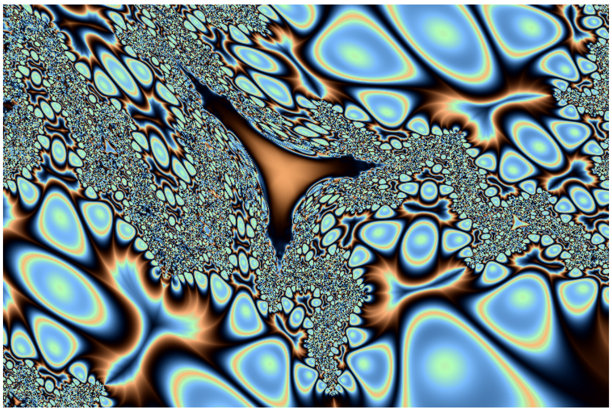
Diameter 2 centred at 0 + 0i, 20 iterations, colouring method: absolute log of fractal dimension of magnitude.



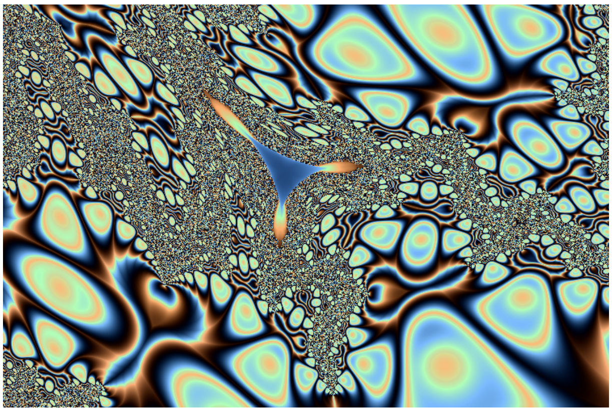
Zoom in of the picture above.



Settings as for the first picture this time with the circle centred at 1 + 0i



Zoom in of the previous picture. A Tricorn is in the process of forming only 20 iterations have been calculated at this stage.



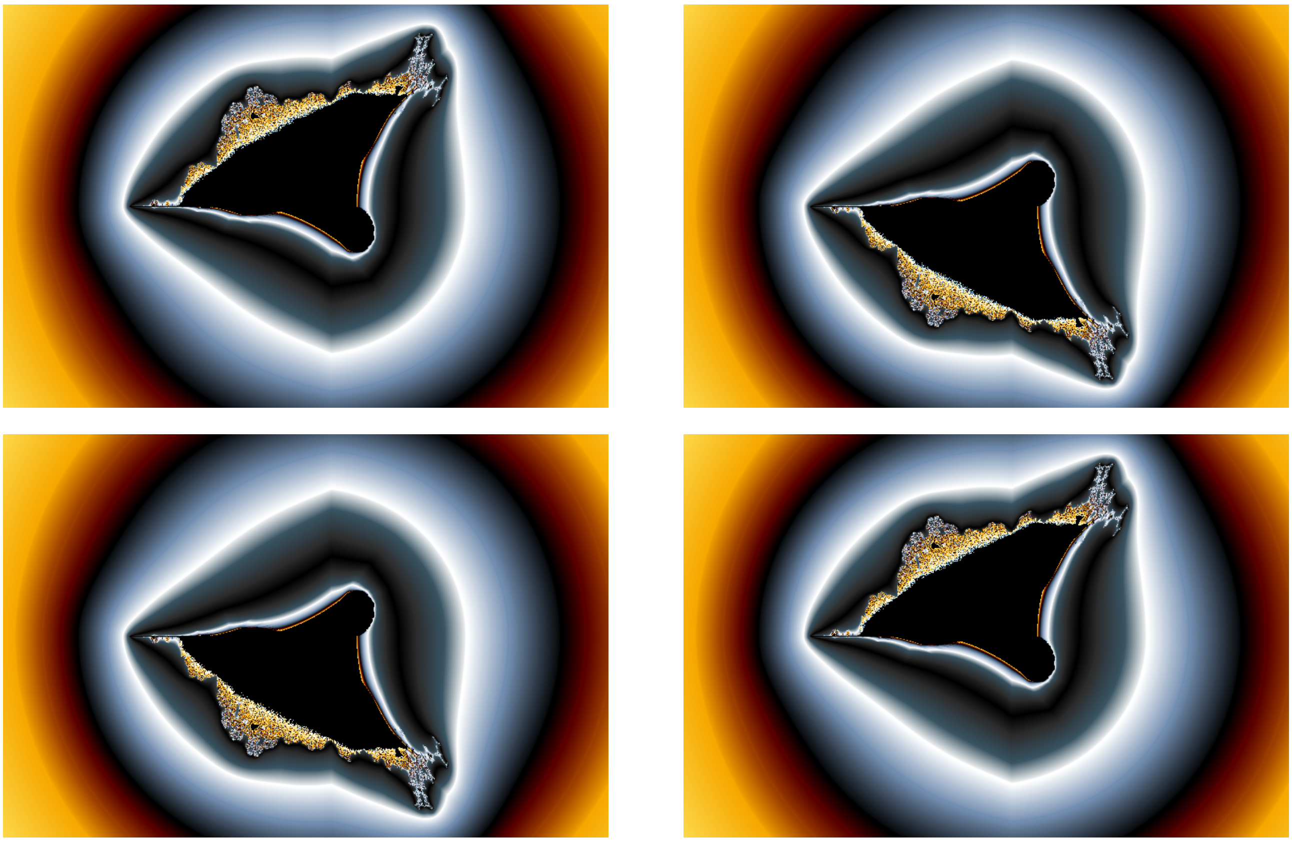
The same image at 200 iterations.

6.9 Quadrant Transforms

The complex plane has a real axis and and an imaginary axis (re, im) which divide the plane into four quadrants, top right (re, im), top left (-re, im), bottom right (re, -im) and bottom left (-re, -im). The quadrant transforms fold values outside the transform's quadrant into the transform's quadrant e.g. for the top right transform:

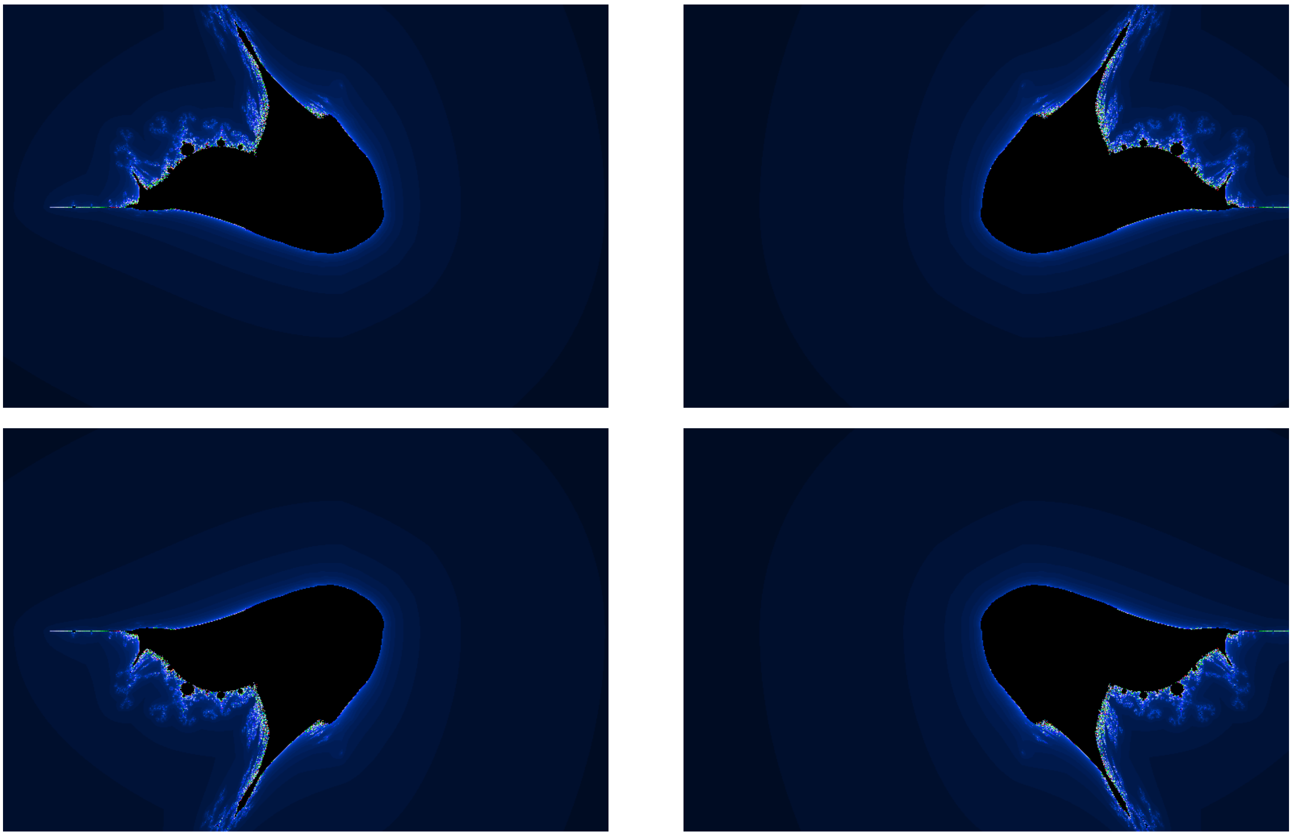
z = 1 + i becomes 1 + i
z = 1 - i becomes 1+ i
z = -1+ i becomes 1+ i
z = -1 - i becomes 1+ i

The transforms applied to the Mandelbrot set result in the following:

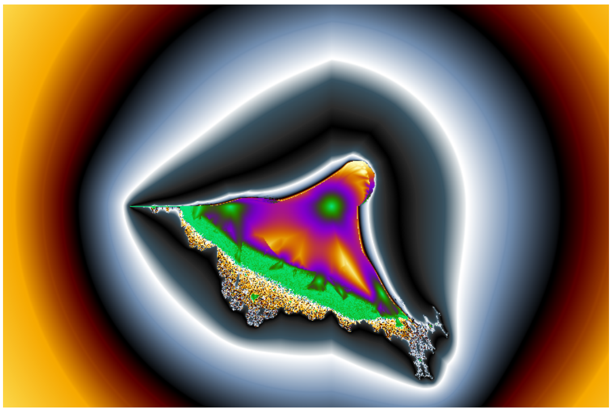


The images are in pairs, top left, bottom right and top right, bottom left and the pairs are mirror images of each other..

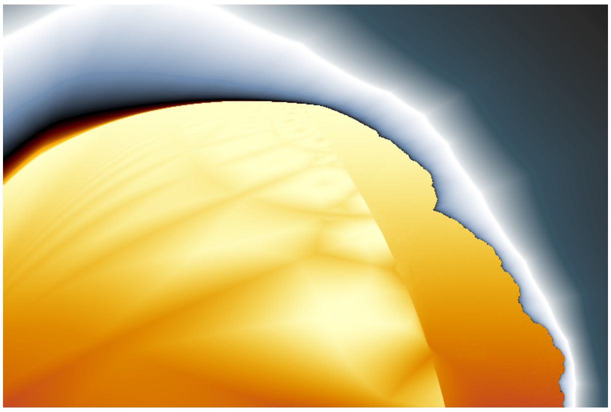
For Compasses (α = 2, β = 0) it's different, we get the same image but in for different orientations:



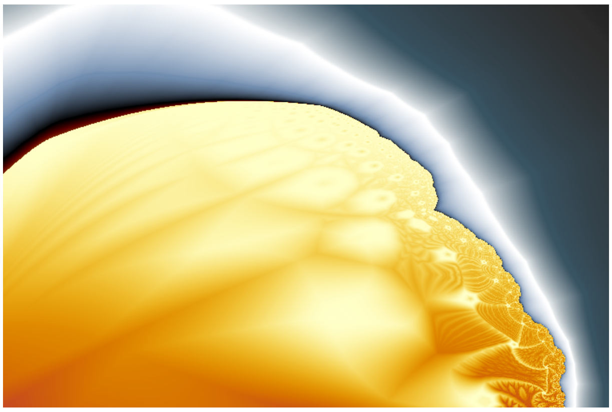
The pictures produced using these transforms becomes much more interesting when inner colouring is other than black. This is the Mandelbrot set with the top right transform:



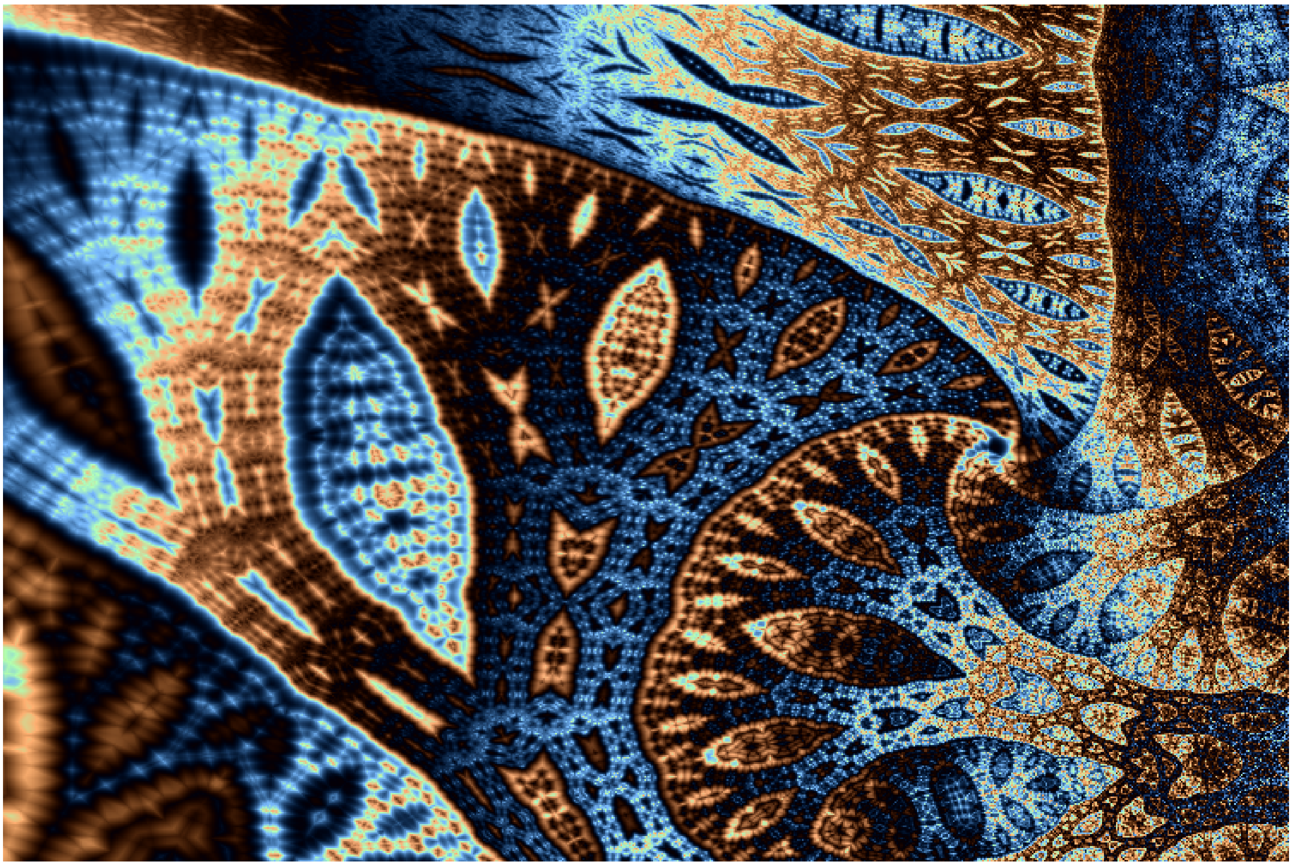
The inner colouring method used here is minimum magnitude. The inner area shows distinct structure.



Zooming into the top part of the inner area, this is the image we get at 2000 iterations. This doesn't look that interesting, this is because the fractral structures have become very small and just looks like shading. Fortunately because Saturn builds the image so many iterations at a time it can be observed that interesting patterns occur at much lower iterations.



This is what we get with 30 iterations.

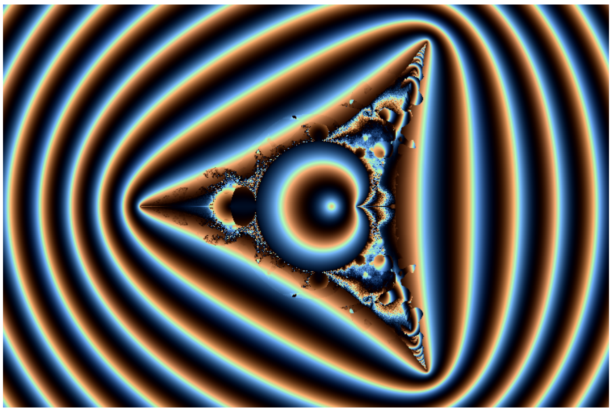


Top.Right.spf

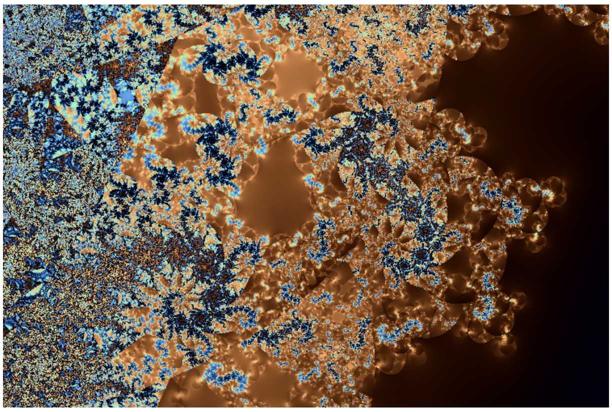
This is a zoom into the bottom right of the previous picture using a different colour map and average magnitude inner colouring and 56 iterations.

6.10 Inverse Fold In

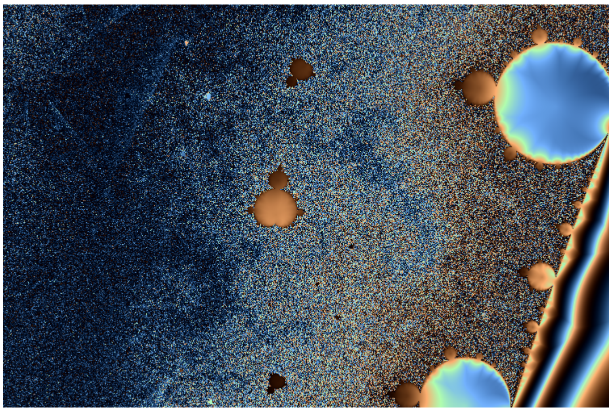
This transform inverts all points outside a circle so that they lie inside the circle so if it is applied to the Mandelbrot set in it default position of the origin with a radius of 2 and black is used for inner colouring then you will get a black circle (only if you zoom out far enough). So when using this transform inner colouring such as magnitude is required. This tranform is similar to circle fold in, however with circle fold in the transformed value is on the same line that originates from the centre of the circle as the untransformed value, but with this transform is does not.



Average magnitude colouring with default transform settings, 500 iterations.



Zoom in, 60 iterations.



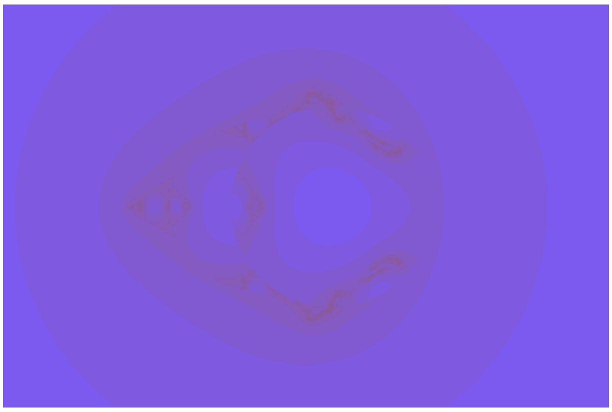
Same zoom, this time using fractal dimension of magnitude and 500 iterations.

As with circle fold in, the circle can be changed in size and the its position moved from the origin.

6.11 Inverse Fold Out

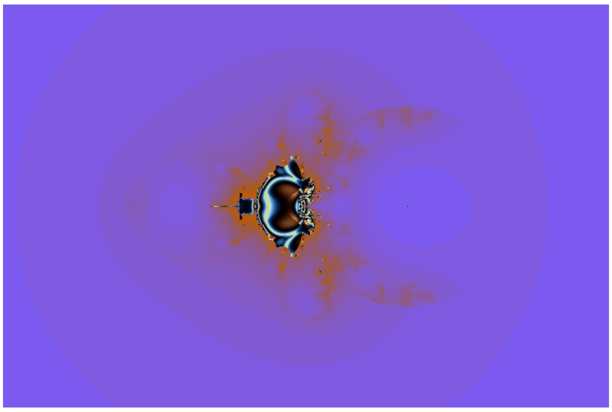
This transform inverts all points inside a circle to points outside the circle. The initial value of z must not be zero, or if the position of the circle and the initial value results in zero when the transform is applied, e.g: if the start value is 1 and the position of the circle is also 1 the value is move to the origin by subtracting 1 giving a position at the origin which then gets mapped to infinity before being move back by adding 1. To avoid this problem the transformed complex plane can be used to set the initial value of z. The difference between circle fold out and inverse fold out is the same as the difference between circle fold in and inverse fold in.

So applying it to the Mandelbrot fractal we get:



Circle of diameter 2 positioned at the origin and z_0 = transformed c.

Not that interesting.

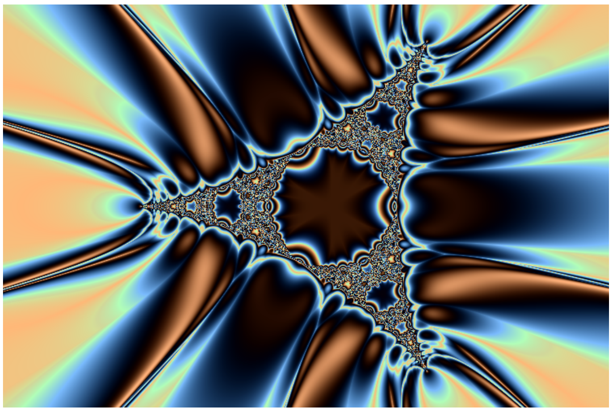


As before but with the circle positioned at 0.75 + 0i. Inner colouring is fractal dimension of magnitude.

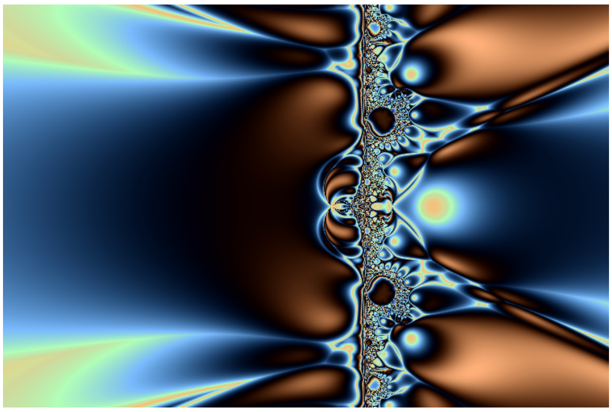
As with circle fold old the image can be changed by altering the position and size of the circle, unlike circle fold out only Mandelbrot islands appear (sometimes distorted), there are no Tricorns.

6.12 Inverse Reflect

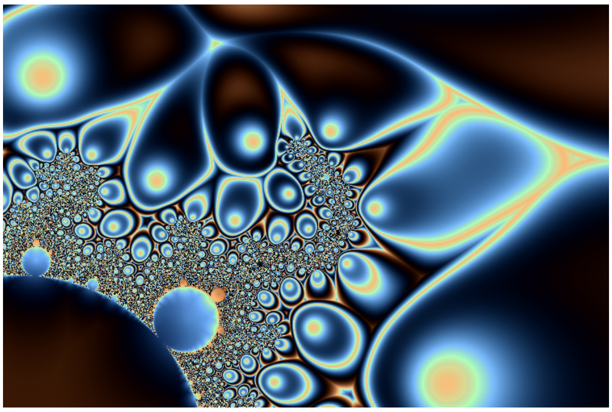
This transform maps points inside a circle outside the circle and vice versa. The effect on an orbit is that it will neither diverge or converge so the bailout condition should be set to no bailout.



Diameter 2 centred at 0 + 0i, 200 iterations, colouring method: absolute log of fractal dimension of magnitude.



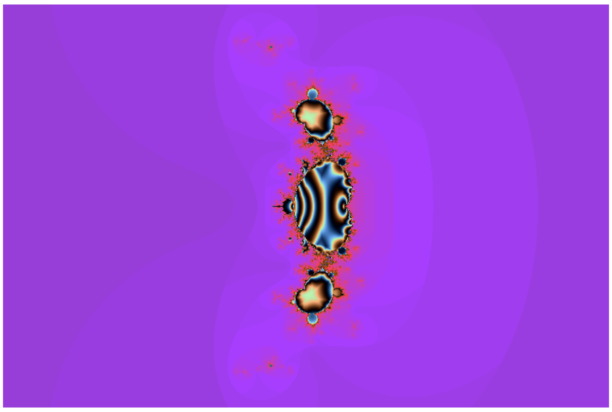
As above, this time centred at 1 + 0i.



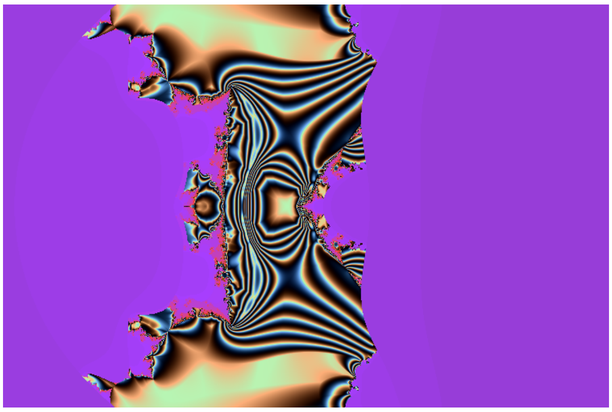
Zoom in of the picture above.

6.13 Logarithm and Exponential

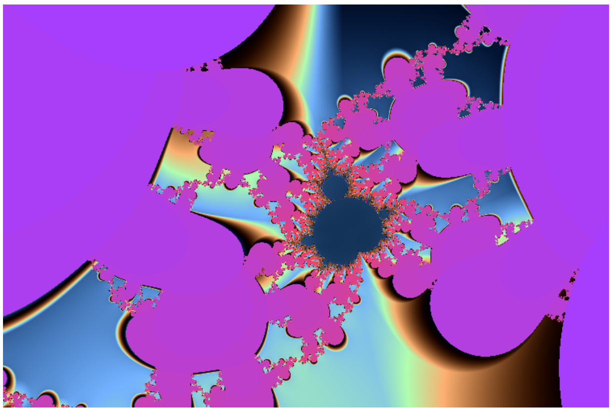
The value of z is transformed by replacing z with either either $\log_e(z)$ or e^z.



Logarithm (log).



Exponential (exp).



Zoom in on a portion of the image above.

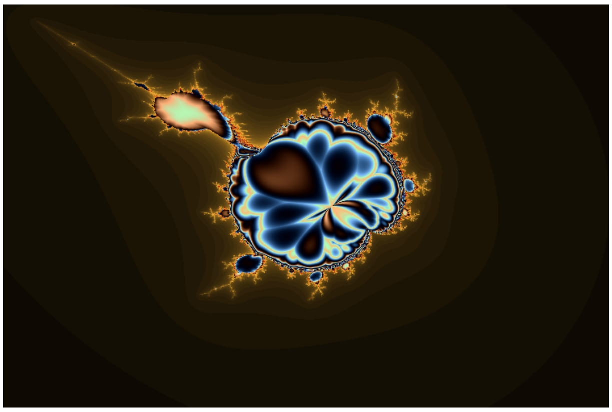
6.14 Formula Transform Combinations

Transforms provide a good deal of the flexibility in manipulating fractal formula in Saturn. The number of fractals types would be increased just by allowing a single transform to be applied to a formula, Saturn provides for two sets of transform combinations, one set can be applied for all iterations or the two sets can be applied alternately depending on a sequence.

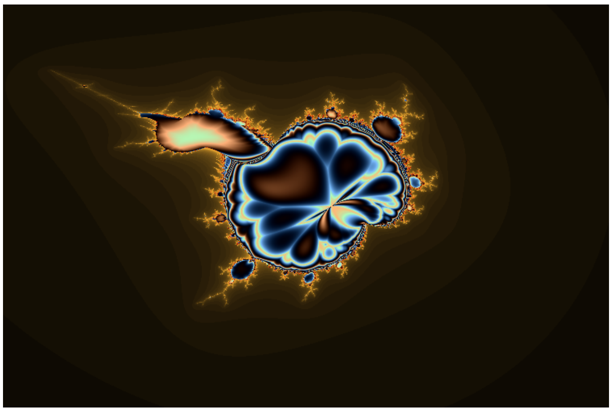
6.14.1 Transform Set

There are two transform sets A and B. Any number of transforms can be added to a set, care needs to be taken so that transforms don't cancel each other out. The transforms are applied in the order that they are in the set, changing the order of the transforms will usually change the image generated.

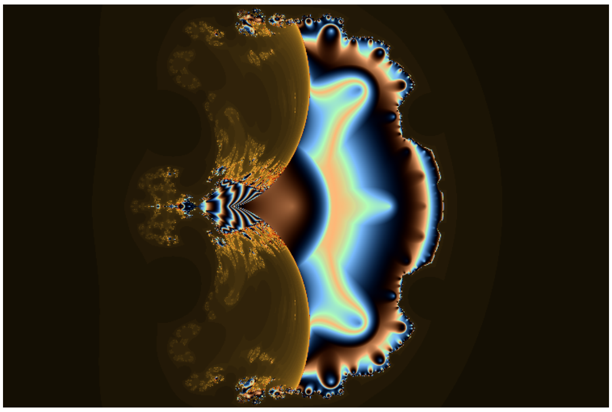
This section will show some basic transform combinations applied to the Mandelbrot set.



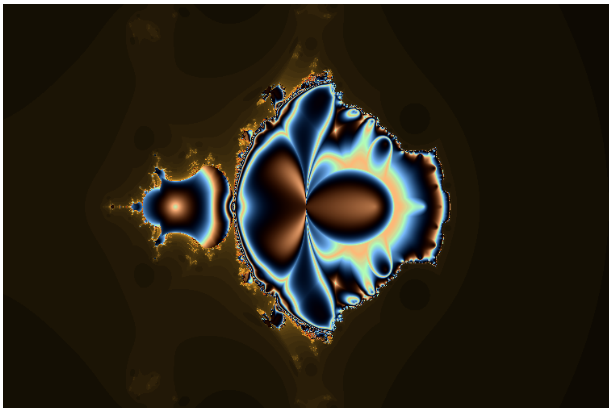
Rotation, 20°
Scale, real 0.75, imag = 1



Scale, real 0.75, imag = 1
Rotation, 20°



Circle Fold Out, real = -1 diameter = 2
Power, real = 2

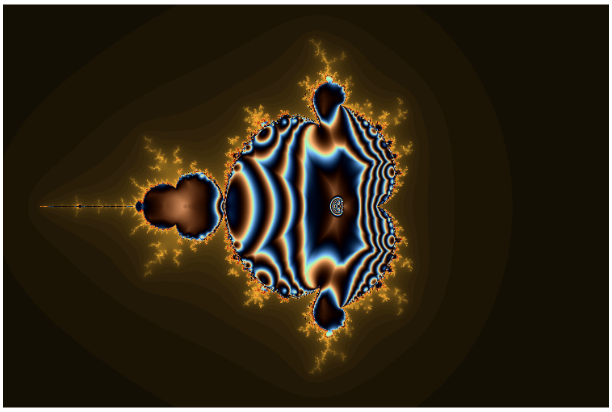


Power, real -2
Circle Fold Out, real = -1, diameter = 2

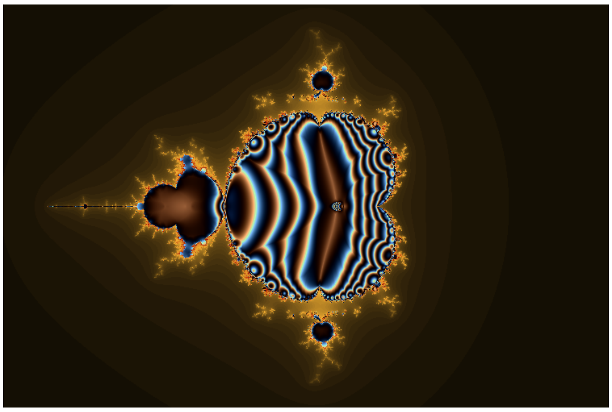
Reordering of transforms in Saturn Version 1.0 is not possible, transforms have to be deleted and re-entered. This will be fixed in the next version.

6.14.2 Transform Sequence

The transform sequence is a string of As and Bs and controls the sequence in which the transform sets are applied to and iteration. If transforms are to be applied periodically then one set of transforms should be empty.



Transforms A: Translate, real = 0.1
Transforms B: empty
Transform Sequence: ABB



Same transform sets.
Transform Sequence: BBA

That's just one example, the transform sequence can be as long as you like and you can add whatever you like to the transform sets, so there's a great number of combinations to try. Transforms can be applied to all the escape time fractals.

7 Complex Plane Transformation

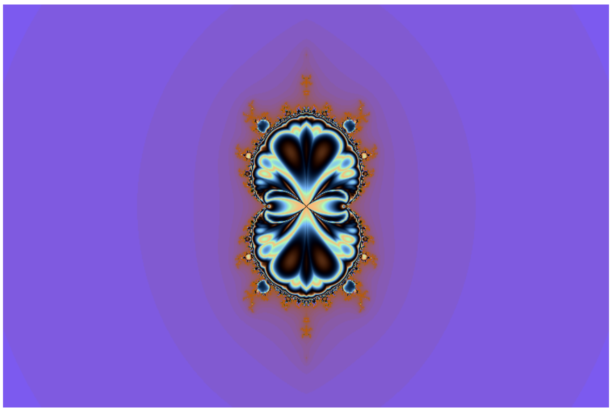
These transformations are applied to the complex plane, c. Locations in the complex plane are used to set z_0 and used in the formula of escape time fractals that use the Mandelbrot algorithm. Unlike formula transforms complex plane transforms are only applied once.

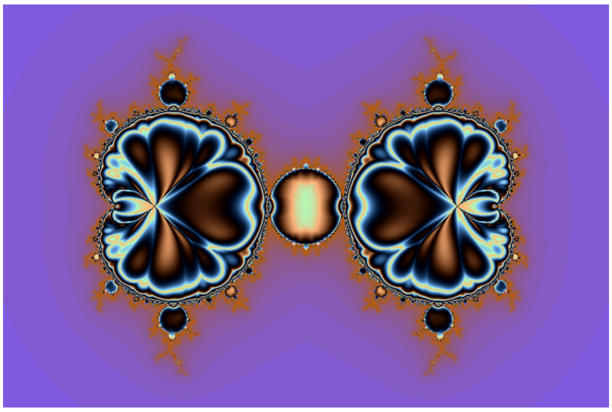
The first three transforms are simple, translation and rotation simply move the image, scale on the other hand will strech the image horizontally or vertically. In addition reverse sign real and reverse sign imaginary simply reflect the image either left to right or top to bottom.

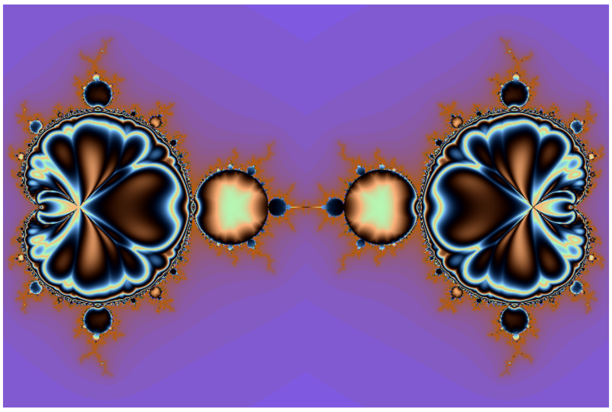
The remaining transforms will be dealt with in more detail using the Mandelbrot set to illustrate their effects. Unless otherwise stated outer colouring will be iteration and inner colouring fractal dimension of magnitude.

Circle Fold In, Circle Fold Out and Circle Reflect transforms appear to be the same as the corresponding Inverse Fold In, Inverse Fold Out and Inverse Reflect transforms.

7.1 Unsign Real



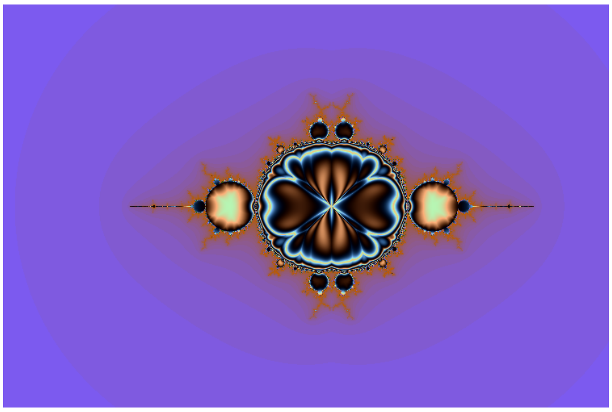


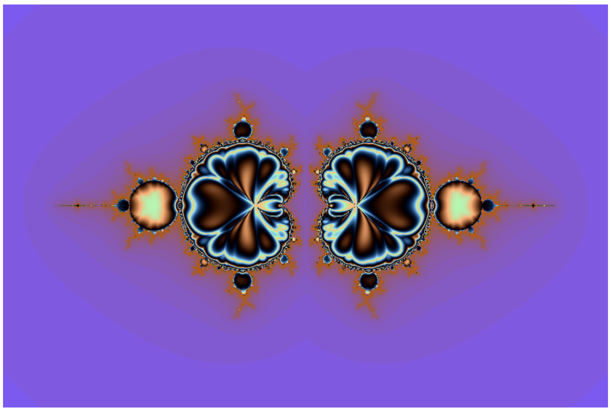


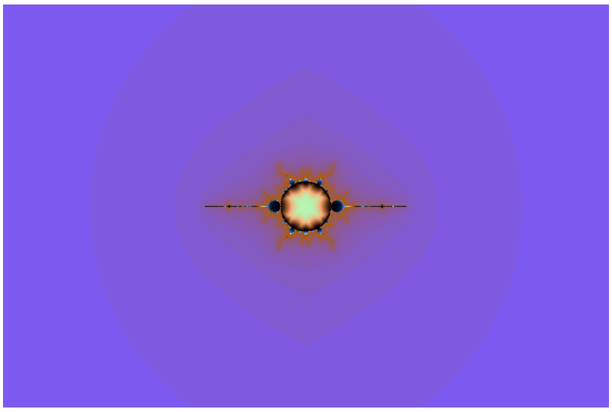
The real component of the transform was set as follows:

above left, real =0
above right, real = -1
left, real = -1.5

7.2 Sign Real



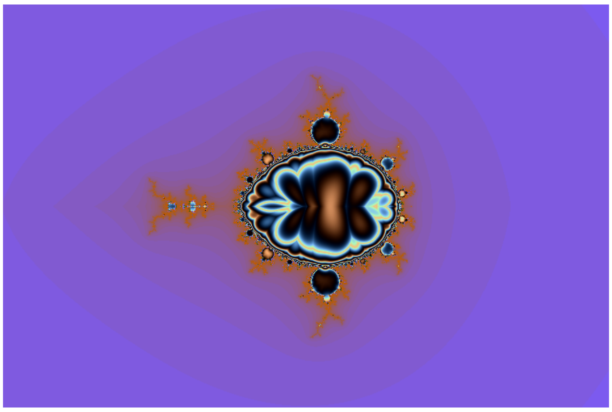


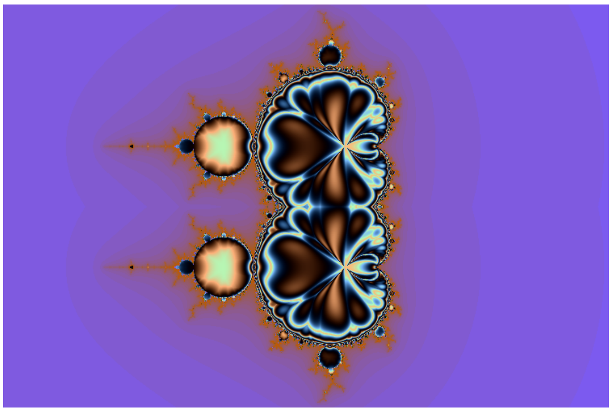


The real component of the transform was set as follows:

above left, real = 0
above right, real = 0.5
left, real = -1

7.3 Unsign Imaginary/Sign Imaginary





The imaginary component was set as follows:

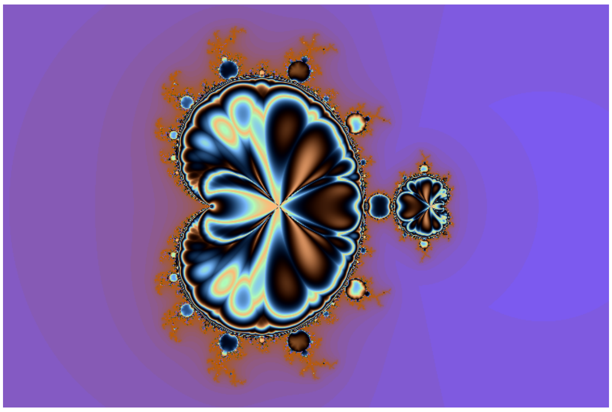
above left, imag = 0.25 for unsign, -0.25 for sign
above right, imag = -0.5 for unsign, 0.5 for sign

The use of 0 for imag has no effect on the image. The reason unsign imaginary and sign imaginary produce the same images is because the Mandelbrot set is symmetrical along the real axis.

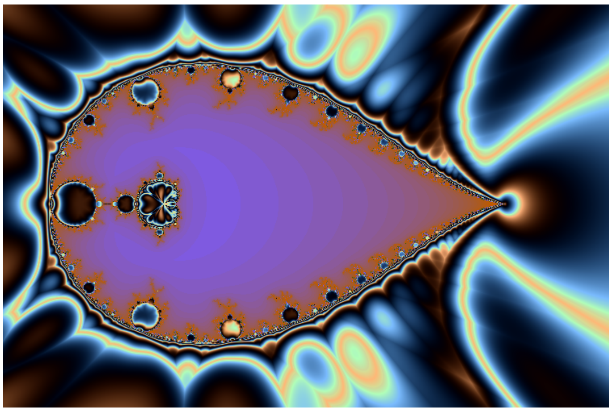
7.4

7.5 Circle Fold In/Inverse Fold In

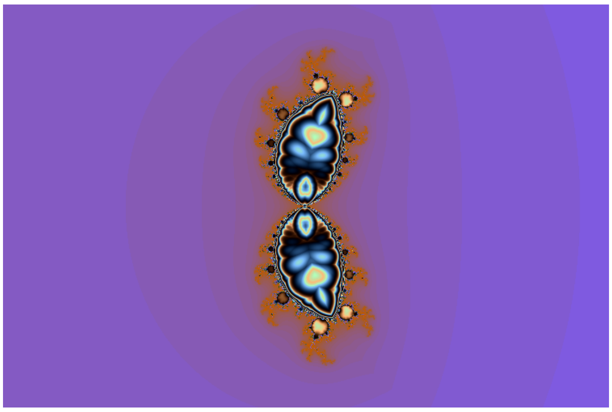
These transforms replace points outside a circle with points inside the circle. They have some interesting effects.



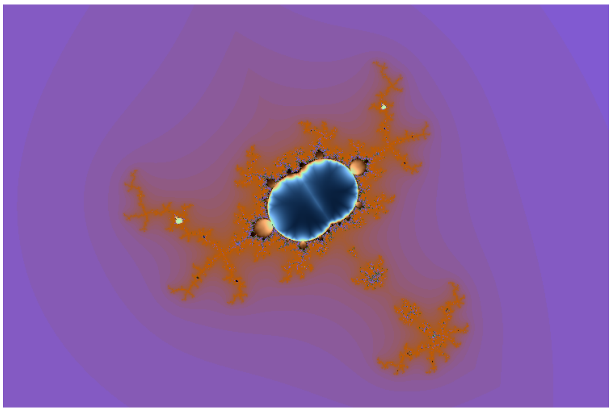
Circle of diameter 4 positioned at 1 + 0i.



Circle of diameter 3 positioned at 0 + 0i.



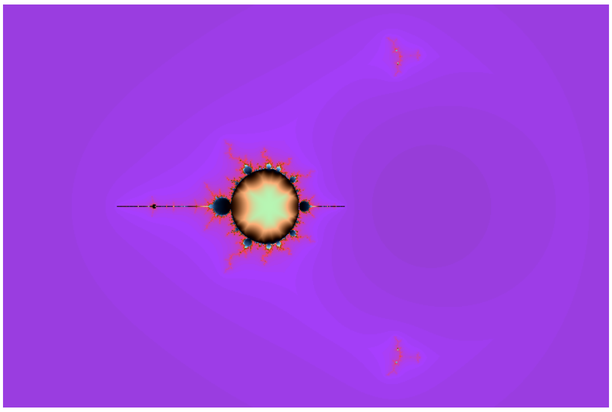
Circle of diameter 1.5 at 1 + 0i



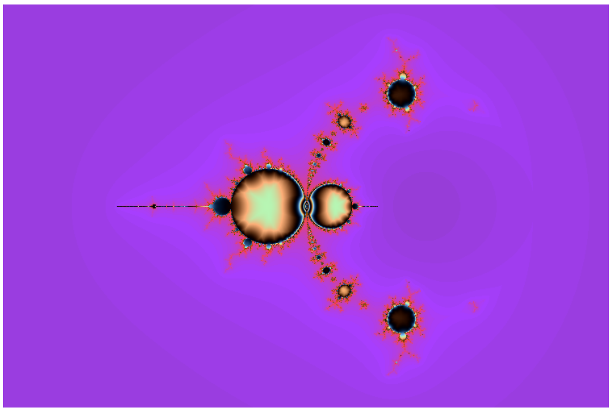
Circle of diameter 1.75 at 1 + 1i.

7.6 Circle Fold Out/Inverse Fold Out

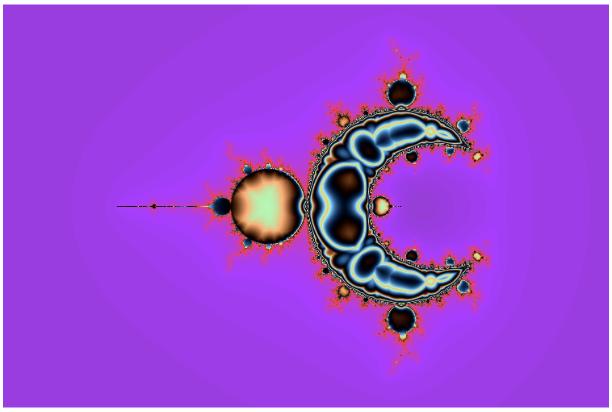
These transforms replace points inside a circle with points outside the circle. They have some interesting effects.



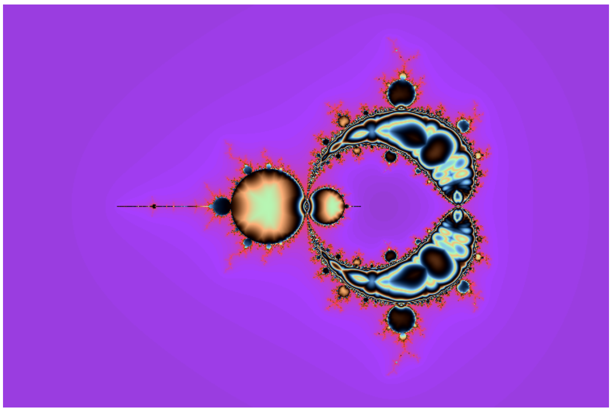
Diameter 2 centred at 0 + 0i.



Diameter 1.5 centred at 0 + 0i



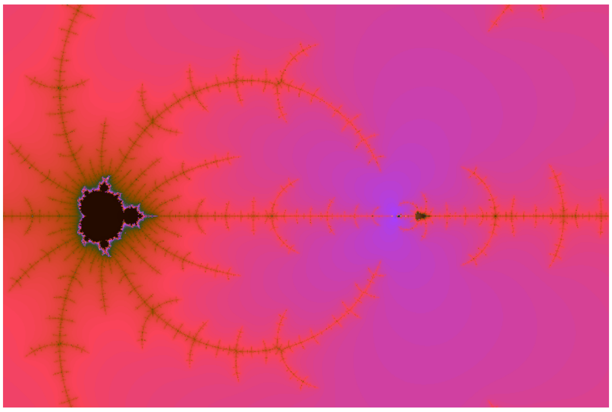
Diameter 1 centred at 0 + 0i.



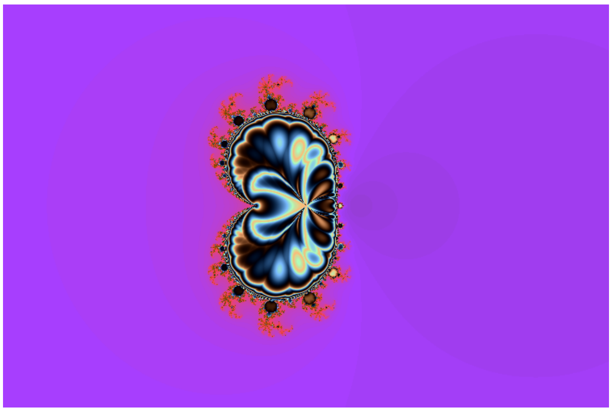
Diameter 1 centred at -0.25 + 0i

7.7 Circle Reflect/Inverse Reflect

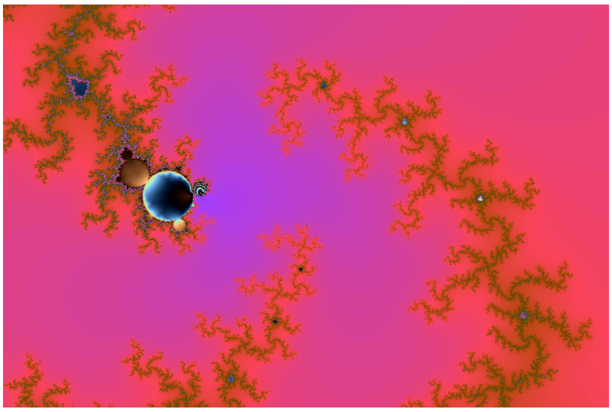
These transforms with default settings have the same effect as the power transform with a power of -1. The position of the inversion circle can be moved from the origin and the diameter of the circle can also be altered. Changing the circle diameter just scales the image.



Diameter 1 positioned at -1.5 + 0i



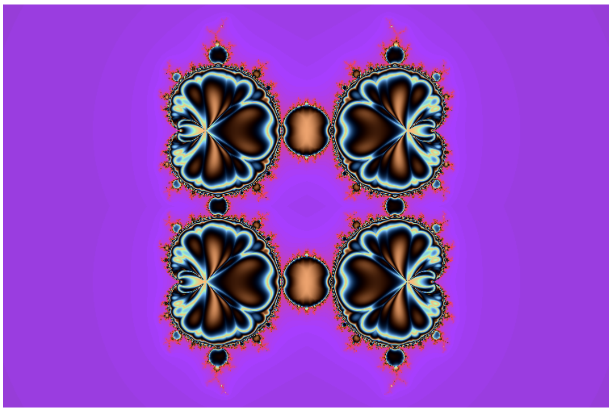
Diameter 2 positioned at 0.5 + 0i



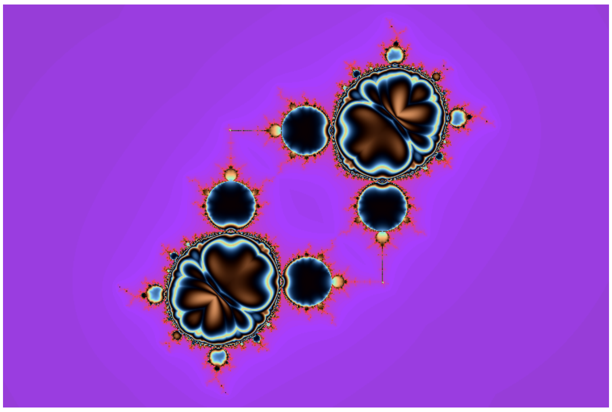
Diameter 2 positioned at 0 + 0.75i.

7.8 Quadrant Transforms

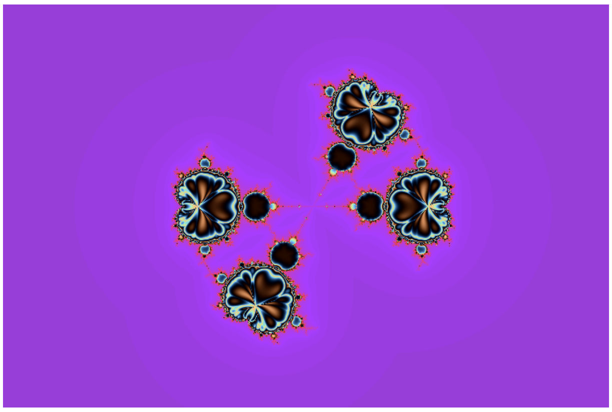
The quadrant transforms are similar to the sign/unsign transforms for real and imaginary but instead of reflecting about a single line they reflect about two lines. Depending on the values of real and imag the resulting image can be the same as for the sign/unsign transforms. When applied to the Mandelbrot set we can get four copies of the Mandelbrot either stand alone or overlapping, the sign/unsign transforms produce only two copies of the Mandelbrot.



Four overlapping Mandelbrots, Top Right transform, real = -1, imag = -0.75



Same as above with addition of 45° rotation.

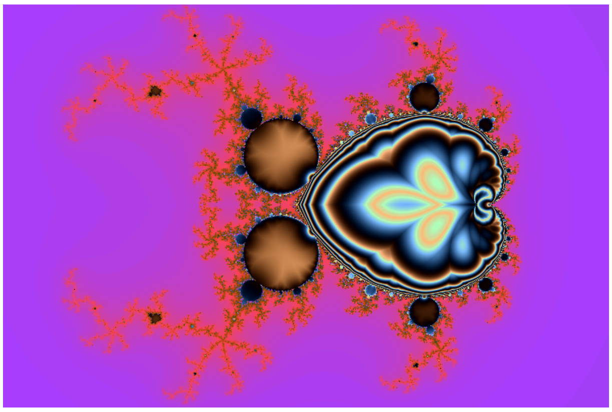


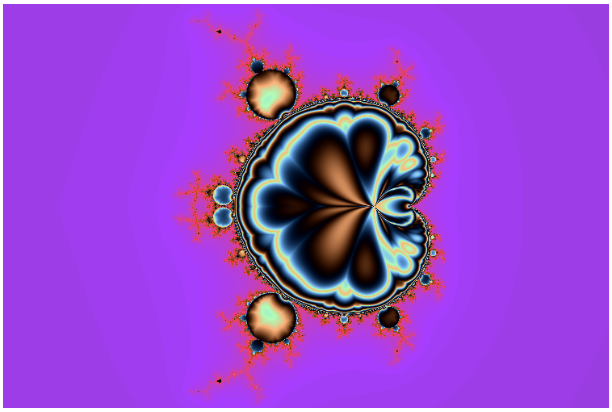
Bottom right transform, real = -2, imag = 0, rotation = 30°.

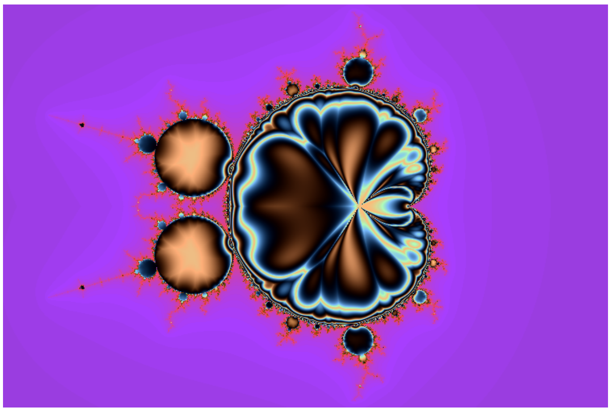
7.9

7.10 Power

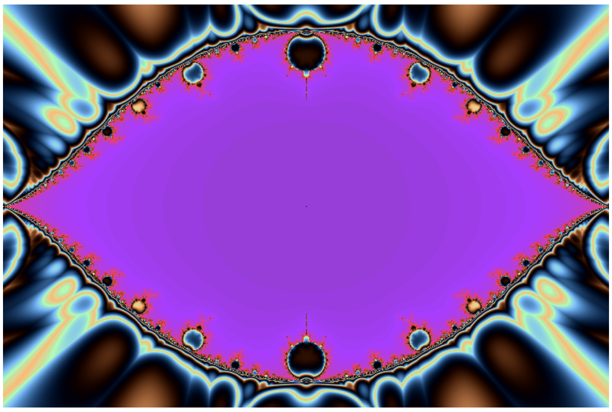
This transform replaces all points with the values raised to a power. This is the transform that produces the inverse Mandelbrot when the power is -1. The power can be non-integer or even complex. Non-integer powers can produce some interesting distortions of the Mandelbrot set.







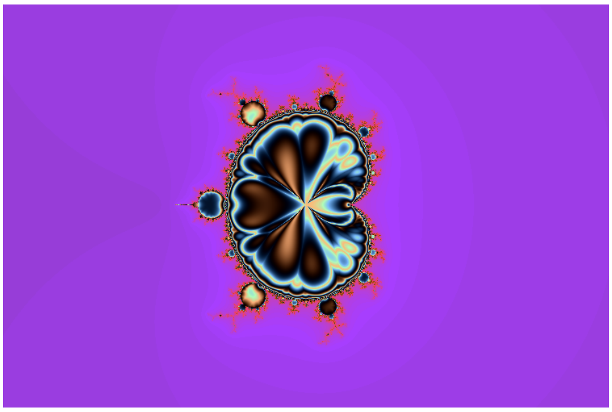
Above left, power 0.25, above right, 0.75, left 1.1.



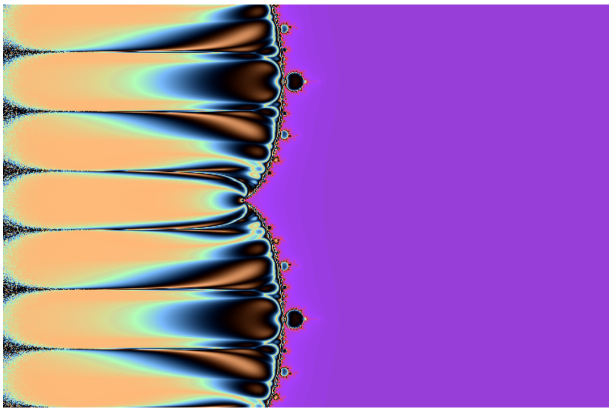
Power -2

7.11 Logarithm & Exponential

These transforms replace c with either $\log_e(c)$ or e^c, there are no options.



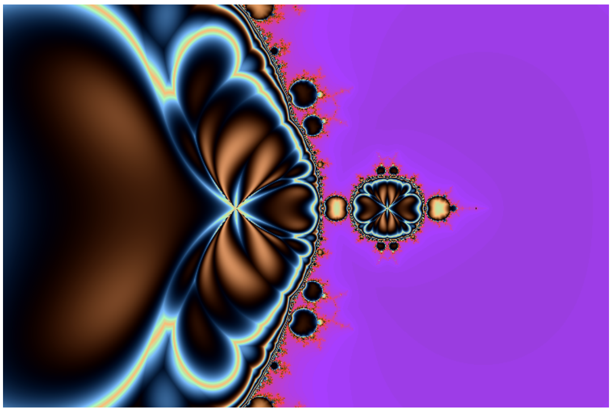
Logorithm (log) transform.



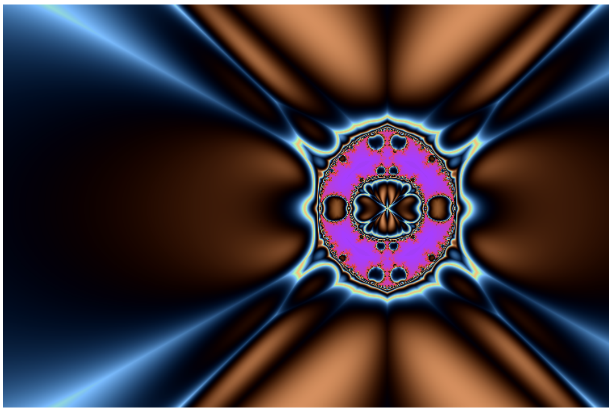
Exponential (exp) tranform.

7.12 Transform Combinations

Complex plane transforms can be be used in combination. The circle and inverse transforms are appear to be the same, however they are not and this becomes apparent when used in combination.



Inverse fold in, diameter 4 centred at 1 + 0i, followed by sign real.



Circle fold in, diameter 4 centred at 1 + 0i, followed by sign real.

8 History

Issue 1 – December 2011

Issue 2 – April 2012