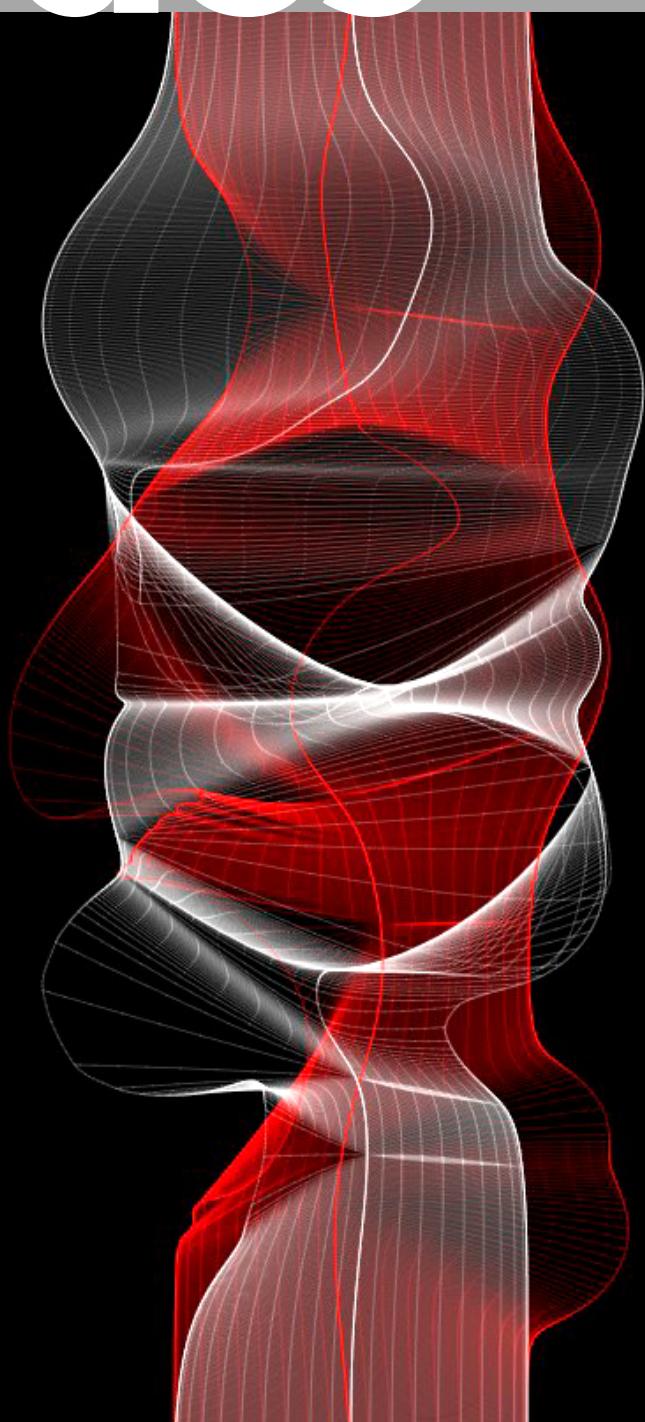


# Nodes



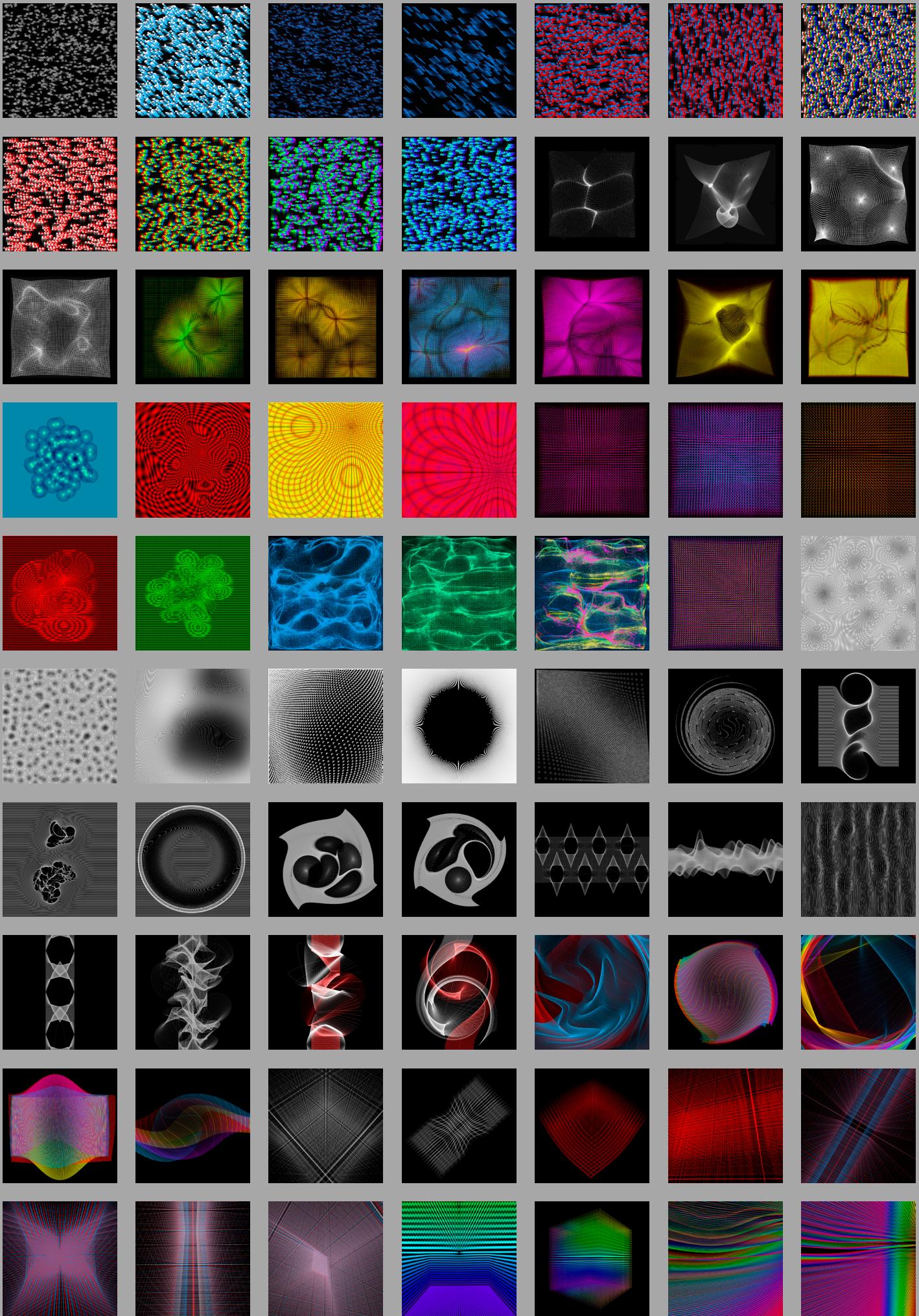
The node is the basic starting point of all graphics in this chapter. A node is a point, knot or intersection. The term is most common in mathematics, computer science and engineering. Nodes are virtual magnetic points in a two- or three-dimensional space. Nodes have attraction or repulsion. In that sense, they can also move within a two- or three-dimensional space. The space itself is thereby influenced visually. This is particularly noticeable when nodes are moving frame by frame. Sometimes nodes are separate points in space. But nodes can also be connected with each other. In a three-dimensional space nodes within a grid can deform the space itself.

---

The amusement Lewis Carroll derived from logical problems is illustrated in an anecdote related by his younger cousin, Alice Raikes. 'The room they had entered had a tall mirror standing in one corner. Lewis gave his cousin an orange and asked her which hand she held it in. When she replied, 'The right,' he asked her to stand before the glass and tell him in which hand the little girl in the mirror was holding it. 'The left hand,' came the puzzled reply. 'Exactly,' said Lewis, 'and how

do you explain that?' Alice did her best: 'If I was on the other side of the glass,' she said, 'wouldn't the orange still be in my right hand?' Years later she remembered his laugh. 'Well done little Alice,' he said. 'The best answer I've had yet.'

Lewis Carroll, 1832–1898, British author, don, and mathematician.



MyCodeHistory: 22 May 2015

I've started with increasing the node count from 20 to 2000. Changed the display size to 800 x 800 pixels. Made the background black. Reduced the size of the nodes to 1 x 1 pixels. At that moment I would like to see very much nodes and at a very small scale. I also slowed down the damping to 0.005. And I changed the random velocity (or speed) to -1, 1. This results in a very lively field with moving points which come to a standstill after a while. As a next step I decreased the amount of node count to 1000. Increased the size of the nodes to ellipses of 10 pixels. I also removed the fill of the ellipses. I would like to replace the ellipses by another shape. I created a function for that called newShape. To begin with I used three rectangles which are diagonally connected. Why rectangles? I have no idea. Sometimes you have to follow your intuition. The color scale ranges from white to blue. When running the program it has sometimes a three-dimensional quality. Which I noticed later. Let's see what happens if I make these rectangles smaller. That gave me an interesting effect. What if I double the amount of rectangles in the newShape function. I got an Exception in thread "Animation Thread" java.lang.RuntimeException: Nothing left on the event queue. at processing.core.PApplet.remove(PApplet.java:2569) at processing.core.PApplet.dequeueEvents(PApplet.java:2639) at processing.core.PApplet.handleDraw(PApplet.java:2317) at processing.core.PGraphicsJava2D.requestDraw(PGraphicsJava2D.java:237) at processing.core.PApplet.run(PApplet.java:2177) at java.lang.Thread.run(Thread.java:744) in my console. But the program keeps running. And that is the most important thing for me.

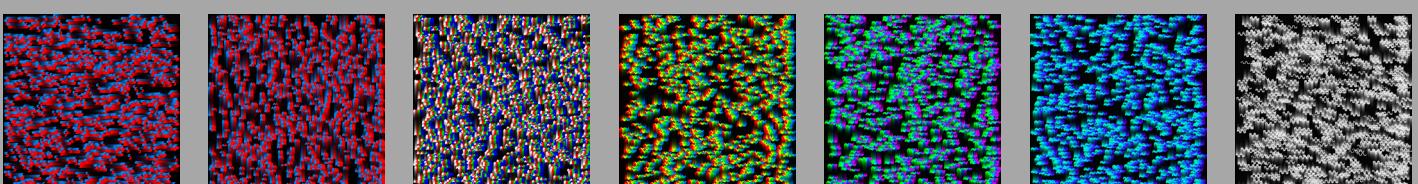
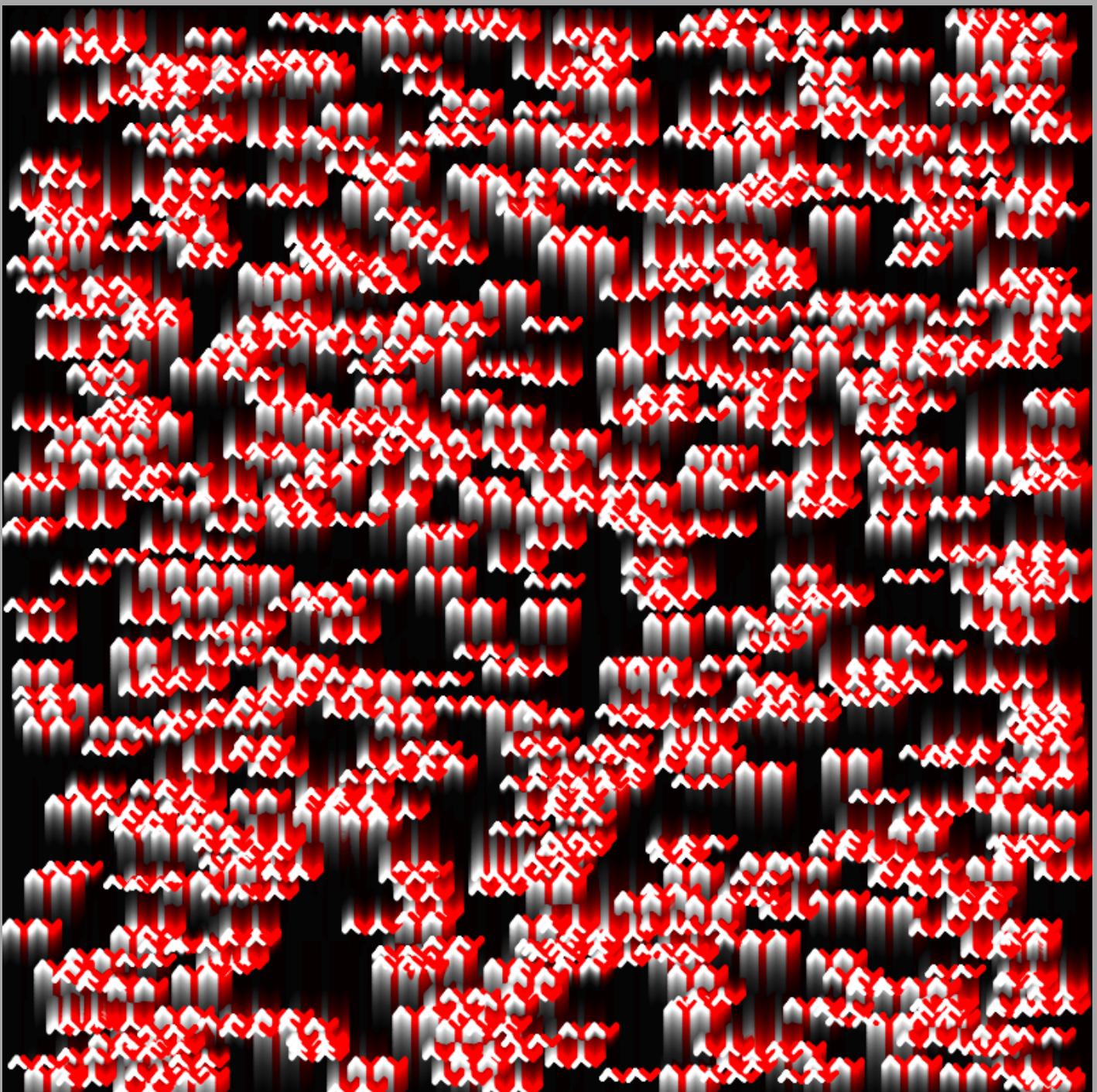
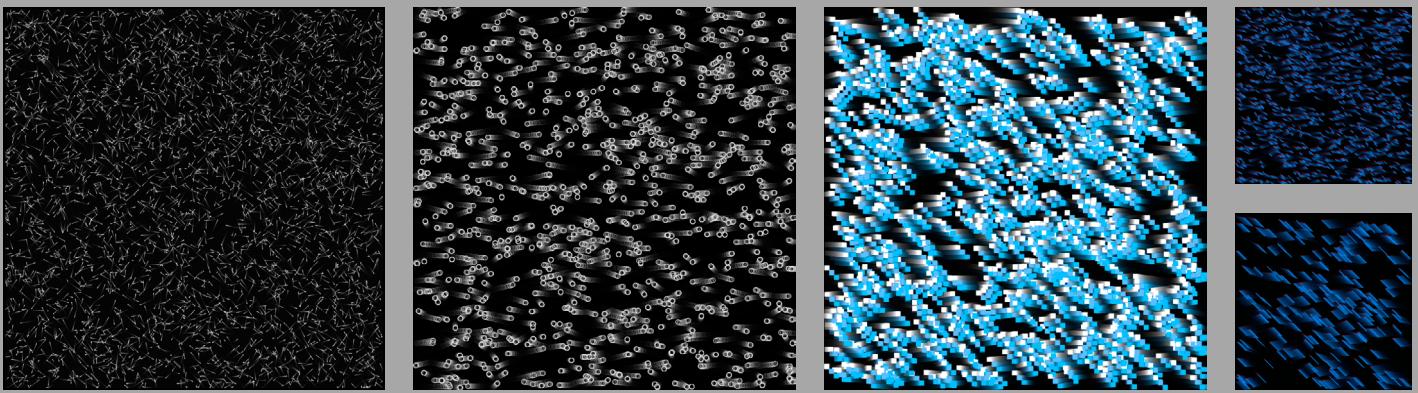
I reduced the vertical movement. That makes the objects even more flag-like. I could use a for loop for this newShape. Which gives me the possibility to make longer flag-like shapes without using repeating code. The red dot on top of the flags is outside the loop. Increased the node count to 1000. And I used an S-shaped form for newShape. I also reduced the randomness for velocity.y (the vertical speed) to almost nothing (-0.001, 0.001). That is giving me horizontal movement only. Swapping those numbers with velocity.x gives me a vertical movement. Used an m-based shape with the same vertical movement. Replaced the m-shape with a double sized m-shape with more color. Not sure if that is a good combination of colours. The newShape is now overlapping itself. Improved the shape by separating the original form from each other. Added a more colourful version too. And I added a Wimbledon color version. Which was not a great success. So I reduced the amount of color variations. And ended with red-white-red-white colours. Also reduced the stroke-weight to 1 pixel which makes the images smoother. Used a horizontal wave existing out of bezier curves. I started with a white version but changed that into a version which uses only red and white.

---

A lot of these programs behave like you are following a recipe in a kitchen. You add something to the minestrone... taste... make a conclusion and add some more color, movement or another shape. And of course you can make mistakes in the kitchen.

I'm not sure if this quote is on the right spot here but it has something in common with people who are trying to pull your leg. Chesterton was once asked what books he would most like to have with him if he were stranded on a desert island. 'Thomas's Guide to Practical Shipbuilding,' he replied.

Gilbert Keith Chesterton, 1874–1936, British essayist, novelist, critic and poet.



MyCodeHistory: 26 May 2015

An attractor simulates attracting forces. The same as with magnets. My first idea was to decrease the xCount and yCount from 200 to 50. Those variables are responsible for the amount of nodes. Adjusted the grid size to 600 x 600. Inverted the background to black. Changed the impact of the attractor on the nodes to 100. And I would like to begin with a very weak force of the attractor on the nodes. So I decided to calculate the force by dividing it by multiplication of the radius with 150. That gives a very subtle difference which may be hard to see. But for me it has just enough impact to be noticed.

Than I doubled the amount of node counts from 50 to 100. And I have no idea where those noisy patterns are coming from. I have also no idea if I like them.

I have set the node-damping to 0.01. Damping is used to slow down the nodes. In fact the setting I now use means that the damping is so low that the nodes (almost) never stop moving. They continue to move for a minute or 2, maybe 3. Some of them may bounce back from the boundaries of the display window. I also increased the radius of the attractor (the mouse button) to 200 pixels.

I Increased the grid to 700 x 700 pixels. Set the node damping to 0.001. Also increased the xCount and yCount amount of nodes to 200. Lowered the alpha level of the fill which draws the real nodes.

Reduced the xCount and yCount again and I increased the size of the nodes to 4 by 4 pixels. Which makes them better stand out from the background.

I have picked up the original settings of the program from the Generative Design book code. Resetting the program to its original numbers gives me too chaotic images. So I reduced xCount and yCount to 40 nodes. The grid-size is now 700 x 700 pixels. And I have defined a function key for adding some red color. I also decreased the effects by reducing float  $f = 1 / \text{pow}(s, 0.001) - 1$ . This gives a very small movement to the nodes.

Used red, magenta, blue, cyan, green and yellow to render a more colourful image. I increased the power by which the base is raised from 0.001 to 0.005. Increased the size of the node from 1 x 1 to 2 x 2 pixels.

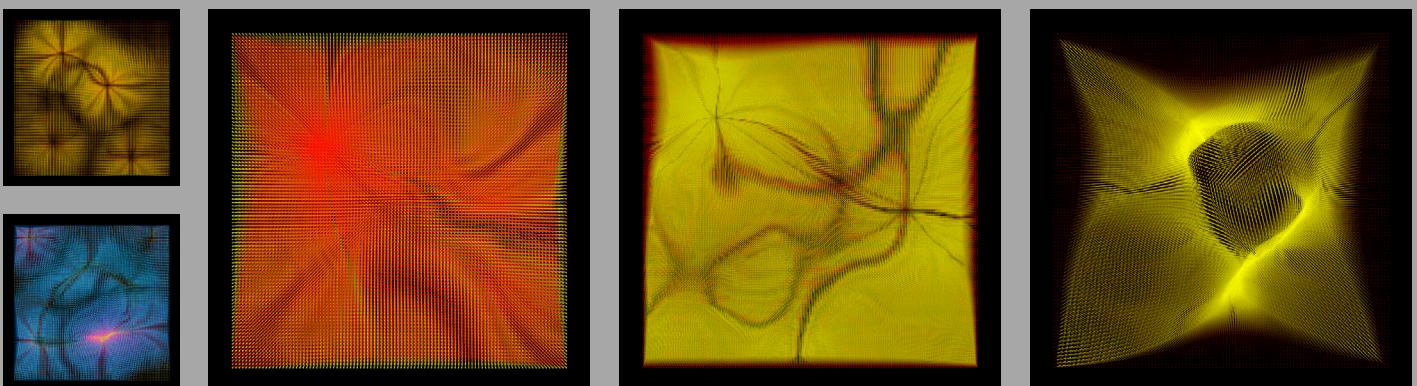
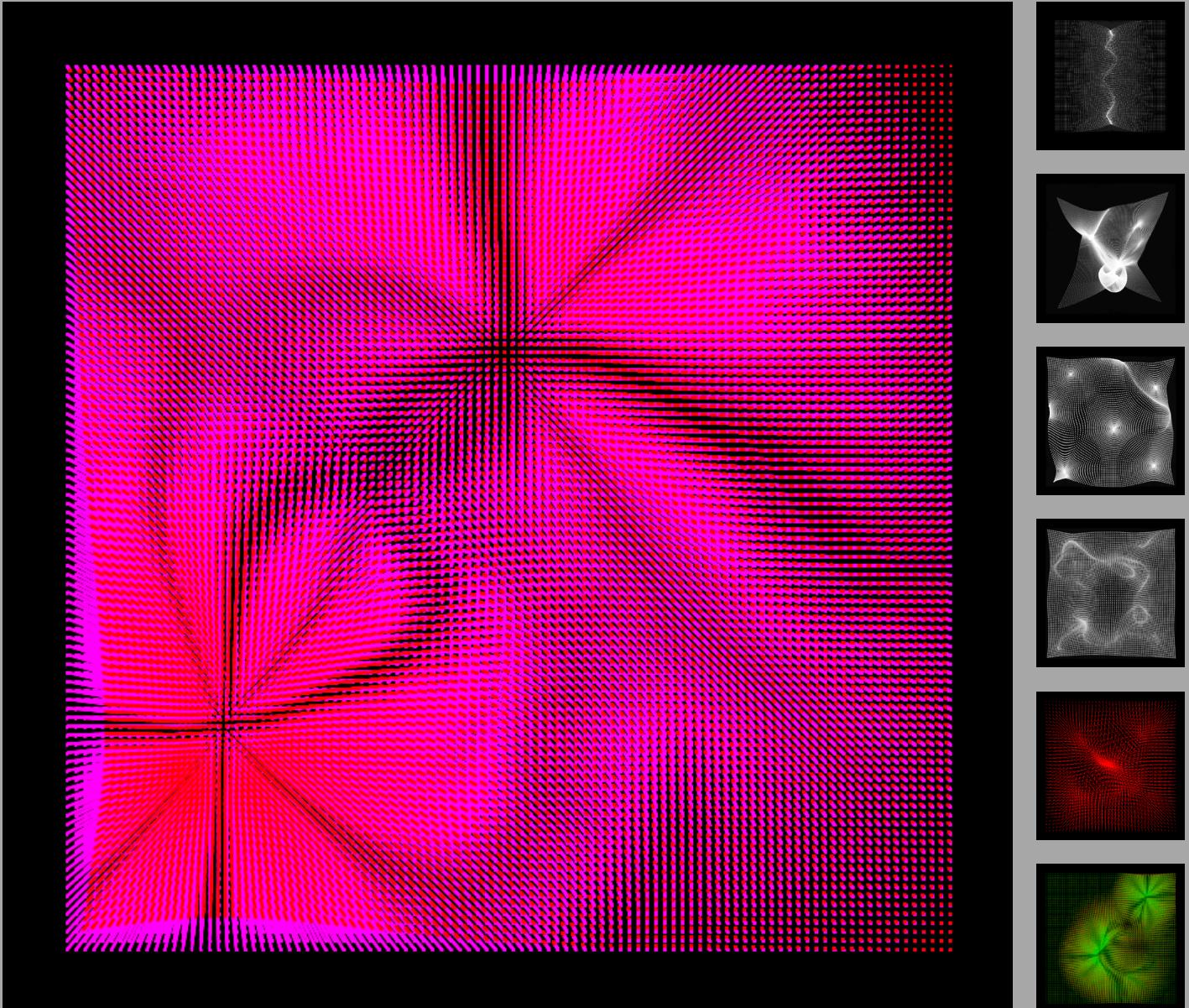
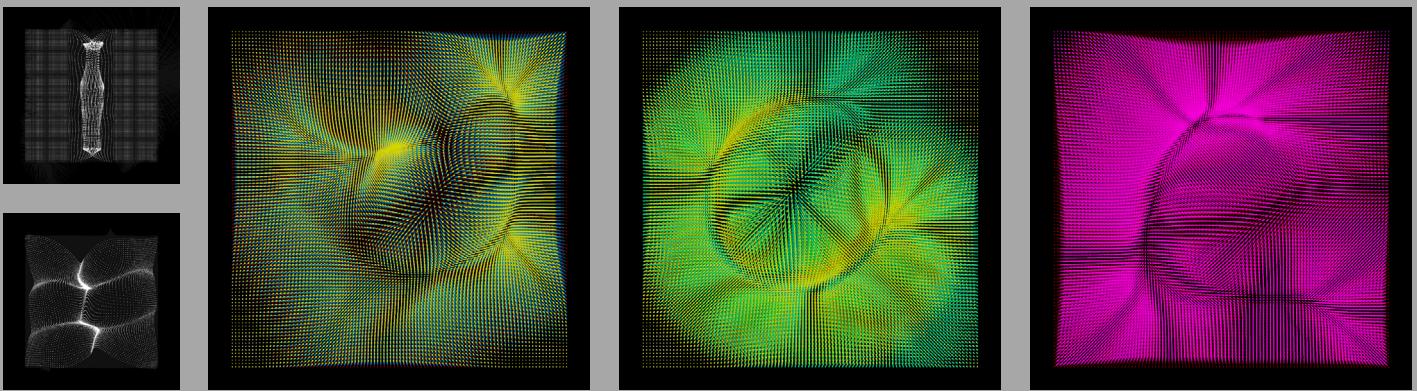
Because I did not really know what was going on I reduced the amount of nodes to 10 x 10. But that did not give me the results which could give me an answer. So I continued to work with the previous settings. Just for fun I tried what black could do to the image. Not very much. It made the image quit rough. All the refinement was lost. Tried white. That did not work either. It bleached-out the image. Increased the attractor radius from 200 to 300. And that worked a lot better.

It seems that (when you decrease the effects by reducing float  $f = 1 / \text{pow}(s, 0.001) - 1$ ) this setting gives the best results. But what happens if I make it a negative number. Well... not much. I put it back to a positive number.

I increased the xCount and yCount to 200. Damping is still on 0.001. The nodes are 1 x 1 pixel. The influence radius is on 300. And  $f = 1 / \text{pow}(s, 0.005) - 1$ . But why I get those TV-screen-like images. I really do not know. And I also don't mind for now.

'The secret of my influence has always been that it remained secret.'

Salvador Dalí, 1904–1989, Spanish surrealist painter.



MyCodeHistory: 29 May 2015

This program is about finetuning the attractors strength And this goes for attraction and repelling as-well. But as usual I started with changing the display size to 800 x 800. Imported the color setting of the previous attractor programs. Enlarged the grid-size to 800 x 800 pixels. And than something strange happened. A kind of soap bubbles appeared. I have no idea why they appeared but they look pretty 3D-ish. Not that I was searching for that effect. I made some color variations with them. But I think only the blue version worked well.

What will happen when I reduce the attractor radius to 10? The circles get very small. And the images arre getting less interesting. So I increased the attractor radius again to 400 pixels and let the program run. Painted some points on the display window. And after a minute some pop-art-ish graphics did appear. Imported the file in Photoshop and adjusted the levels. Not bad... although some banding occurred in the image.

As easy as the first and second variations where coming up so difficult it was to get something sensible in the third part. Most of the time its best to go back to the original settings when you get stuck. So I changed the grid-size back to 800 x 800. XCount and yCount are on 401. Than I just pointed with the cursor in the display window. After a few seconds theses spiral things shaped themselves.

I did not change anything in the program. The only difference with the previous program is that I pointed twice in the display window with my cursor. I think the images are a bit too harsh.

Made the grid-size 2000 pixels. Which sounds weird because my display size is just 800 x 800. XCount and yCount are on 101. Put the attractor strength on 10 and the ramp of the attractor on 0.04. I also made random movements in the display window with the cursor to get interesting patterns. Although the movement is very minor and slow the image looks quit balanced.

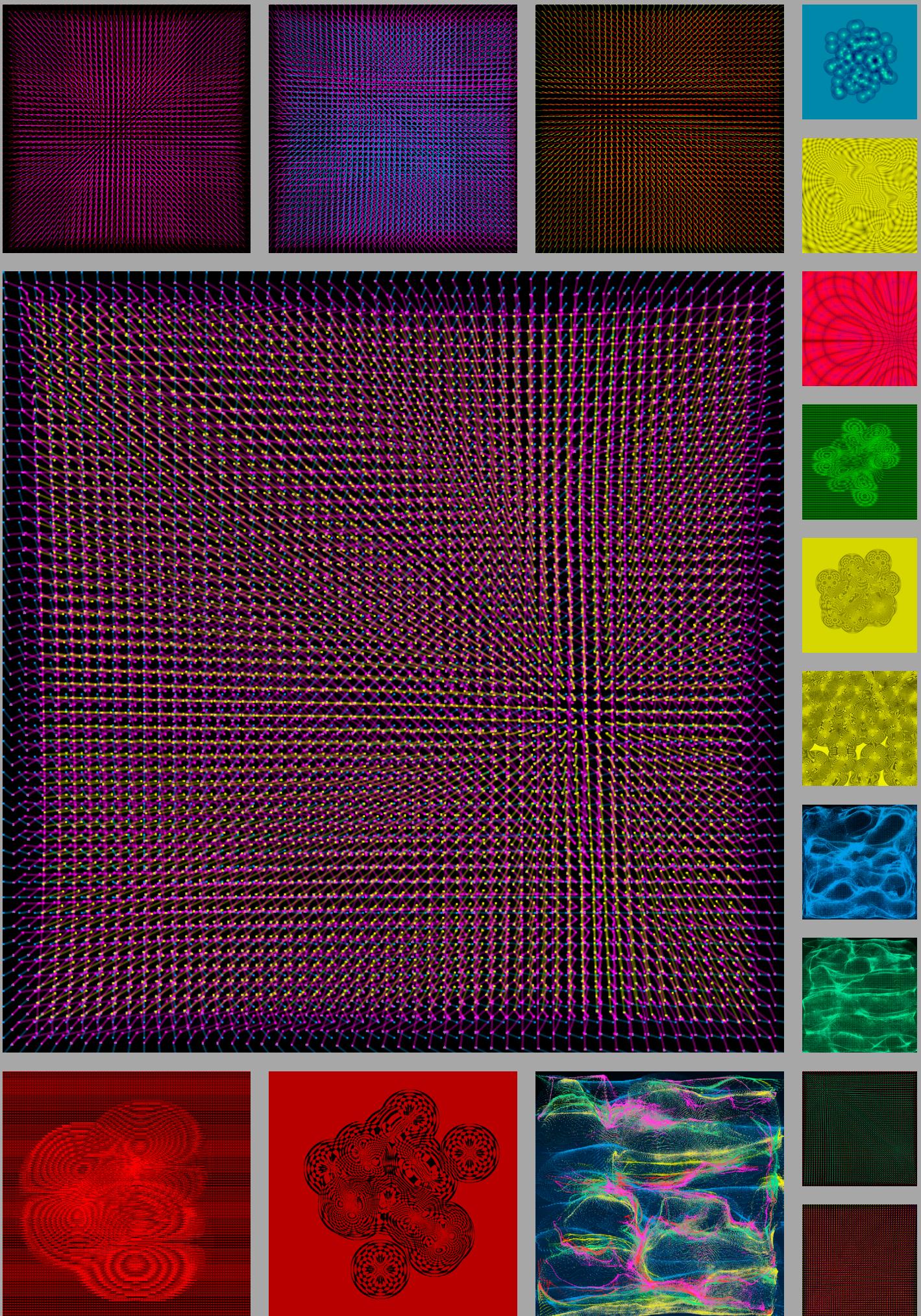
I think that the time has come to check what those variables are really doing. I start with adjusting the attractor strength. When you put it on zero nothing will happen. Which is quite obvious because there is no attraction involved. Because I do not know if there are limits to the attractor strength I changed it to 100. The rectangles are 3 x 1 pixels. And when I touch the display window with the cursor it creates flower-like objects. Again a bit too harsh and there is some interference going on. But it looks interesting.

I have increased the smooth setting to 8. Hope this helps to get the images a bit less noisy. The grid-size is reduced to 800 x 800. XCount and yCount are on 801. Which seems to be silly but if you put it on 800 you get an unwanted pattern in the background. Why? I don't know. Anyway I have increased the strength of the attractor to 500. And by pointing the display window at several positions I get these nice patterns which interfere with each other.

I wondered why I get circle-like patterns while I draw with rectangles. So I replaced the rectangle with an ellipse. And than I still get circles but it takes longer to draw them. You also get less of them. So I changed that program-line back to rectangles. And I increased the attractor strength to 1000. Instead of pointing at the display window I now drag slowly over it.

XCount and yCount are now on 401. The grid-size is 800 x 800 pixels. The attractor strength is on 1000. And the attractor ramp is on 0.06. Made slow horizontal movements with the cursor in the display window. It displays low tech pixels in an unpredictable way.

I increased the impact radius of the attractor to 1600 pixels. That is double the size of the display window. The nodes themselves are now 2 x 2 pixels. 1 x 1 Pixels made them hard to see. And the strength of the attractor is on 4. The ramp of the attractor is on 0.09. This means that there is very minor movement. And the trails left by the nodes are very short. Combined in a few colours this gives very interesting patterns of dots and lines.



MyCodeHistory: 3 June 2015

I inverted the fills and background color. Changed the grid size to 800 x 800 pixels. XCount and yCount are on 801 (xCount and yCount is the amount of nodes). Left all settings as they were but made a few variations with them. Ah... the pdf-file turns out to be just a black document. Changed the grid-size back to 600. When I reset everything the pdf-file does contain an image but it doesn't look like the one which is generated by Processing when it's set to generate a png file. Checked the pdf-file in Photoshop and it contains no image. So I do not worry about that and continue with working with png's. Made same variations with these grey moiré patterns.

Than I decreased the attractor radius of impact to 50. That is the influence size on the nodes. The nodes themselves (xCount and yCount) are 701 x 701. Which means you get a finer pattern in the display window. In fact it is just the same pattern as in the previous sketch. Their just a bit smaller.

I decreased the attractor radius of impact (influence size on the nodes) to 500. Did not change anything else. And than you get these moiré patterns. A bit more harsh and somewhat less refined. There is a strange side-effect. When you look at the preview in an OSX-window the preview does hardly look like the image you see when displayed at a 100% size.

Increased the radius of impact for the attractor to 800 pixels. XCount and yCount are on 201. I use rectangles made of 8 x 4 pixels. Made a few variants with different rectangle sizes. Interesting that these patterns are not depending on the shapes they are created with. It does not matter what shape you use as long as the shapes are small.

Decreased xCount and yCount. Grid-size has also increased. The attractor strength is 4 times as high (in the negative numbers) and the attractor ramp is doubled. Interesting to see that ellipses which enter the borders are piling up to each other. This gives a kind of rough photo-edge. But in the end I decided to clean it up because I found it disturbing for the final image.

Made a version which is the inverted variation of the previous sketch. It does give a complete different image. This leads to some variations with 3 ellipses of different size and position. It also generated some rubbish on the borders. So I increased the display-size. Maybe it is necessary to clean it up in a later stage. Made five variations with five different cubes under different angles. Cube front left, cube front right front outside, cube front inside, cube front transparent.

Made five variations with lines only. They are not perfect but I think this imperfection is not a negative thing in this case. And finally I made four patterns with different objects. Two flower and leaflike patterns. One with honeycombs and used some ellipses thinking about flowers in this time of the season.

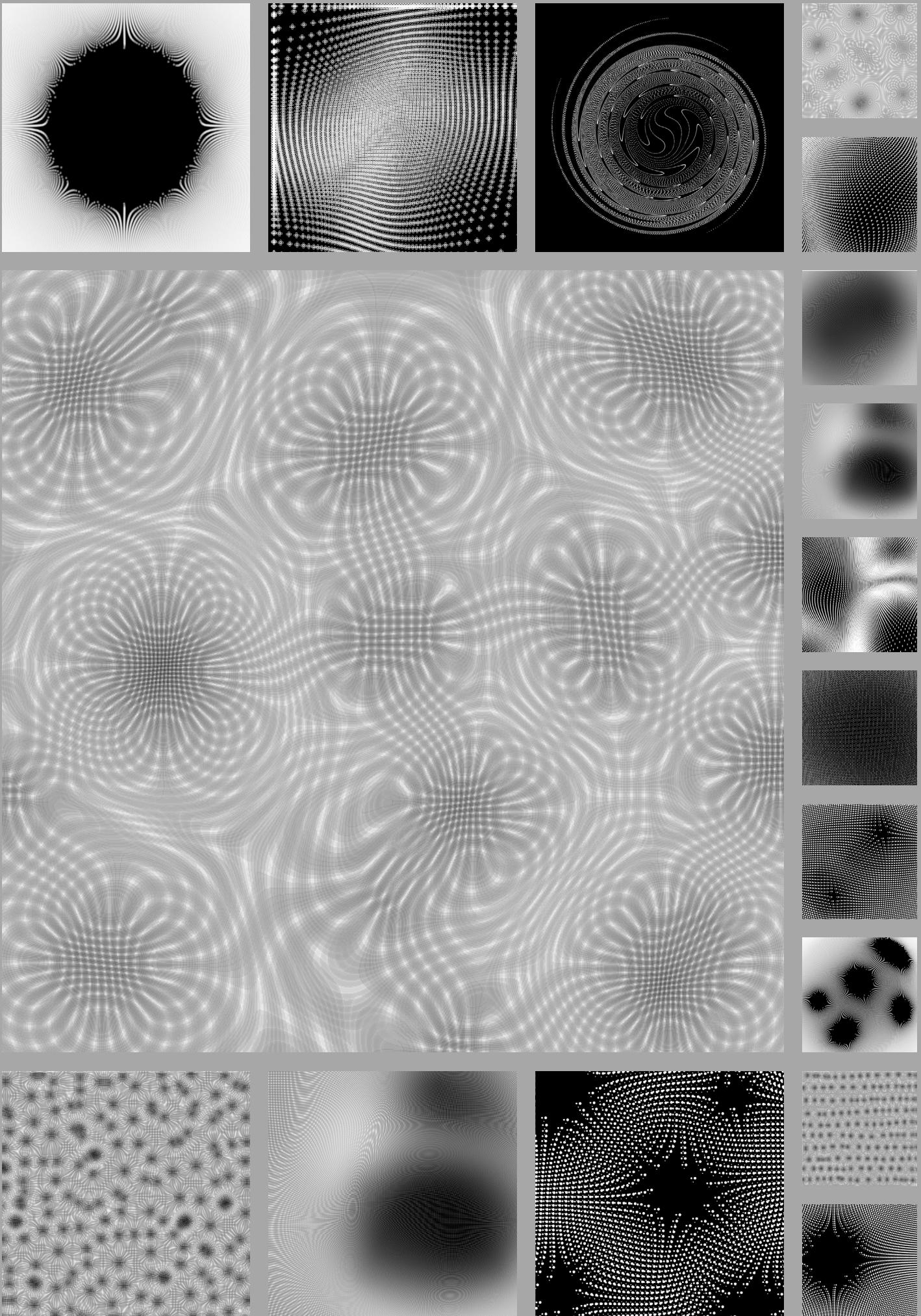
---

I think it is very help-full when learning (to program) that you are curious all the time.

---

Edison was asked to sign a guest book that had the usual columns for name and address, as well as one for 'Interested in.' In this last column Edison entered the word: 'Everything.'

Thomas Alva Edison 1847–1931, US inventor.



MyCodeHistory: 10 June 2015

It is quite interesting to see how these chapters lead you into very complex programs. It all starts very simple and during the using and modifying of the programs you get a better understanding how the last program (the tool) does its work. In this assignment I began with inverting the lines and background color. The lines are in fact build by very tiny ellipses. Changed the grid size to 1000 x 1000 pixels. That gives me more headroom to misuse the grid of nodes. The pdf delivers a wrong example of the Processing sketch. It draws the ellipses with black outlines. So I do not worry about that and continue working with png's. Made some variations with those swirly lines. In the last variation you can notice the separate dots very well. But it's nice that they still behave in line of each other. I also kept everything just black and white. Doing that keeps me concentrate on creating shapes without worrying if it's a good color combination. If it works in black and white it surely will work in color.

Made a few variations with the twirl. I think it is too frivolous. To anticipate on that I lowered the damping to 0.1. But that is giving me the opposite effect of what I would like to see. Maybe this is another opportunity to do something with it. I also replaced the ellipses by a rectangle. That makes the program faster and because the rectangle is only 2 x 2 pixels you don't notice that it is not an ellipse. I also increased the alpha channel to 32. Start with grey lines and because they are overlapping each other they create white areas. I think the compositions are best when you leave some areas of the horizontal line pattern untouched.

Reduced the attractor size and the attractor damping. The damping is now so low that it takes almost a minute to stop all motion. I think it's too low so I raised it again. The horizontal line pattern is now so dense that it almost seems to be a filled grey square. Increased the impact radius of the attractor. Decreased yCount because of the disturbances in the image. And I also increased the attractor ramp. The program generates very unusual images. The side effects are also very interesting. Looks like a sound wave has gone through the lines.

In the previous sketch I dragged random through the display image with the cursor. In these images I just pointed with the cursor at a few arbitrary places in the display window. I tried not to move or drag with it. Just keeping the cursor roughly in the middle of the display window.

In some examples I made very small circular movements with the cursor in the middle of the display window. That creates very rough circular images. I changed the force to the node speed. I divided dy and dx by f. As a final attempt I brought everything in to extreme high numbers. Which seems to be a good thing for creating strange images.

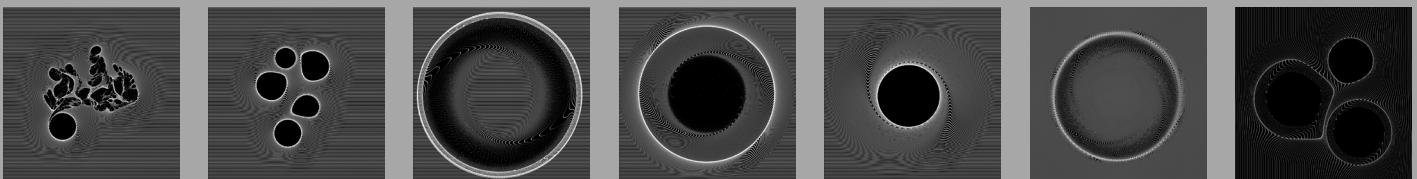
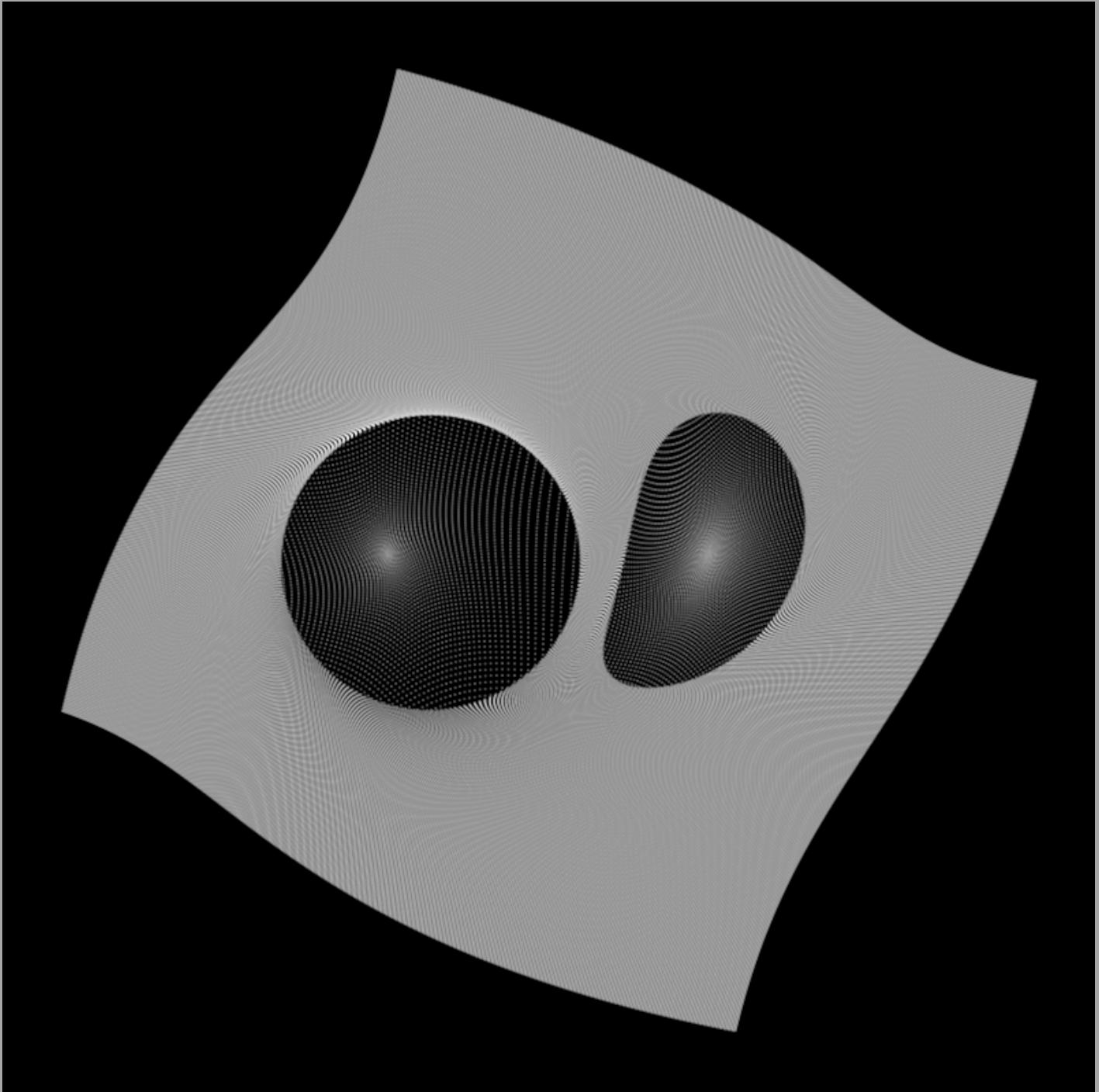
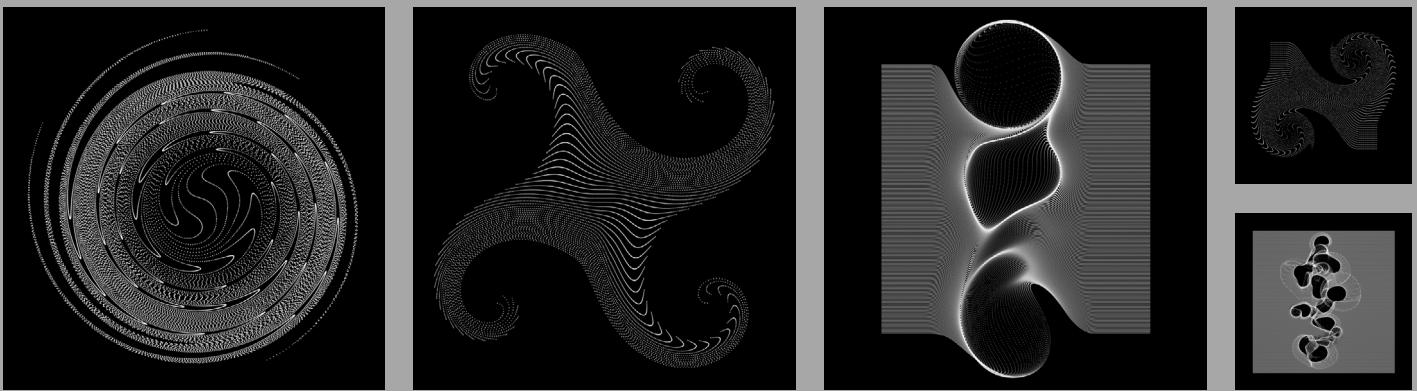
---

Sometimes it is good to concentrate on just one particular subject of what makes an image. In this case I concentrated on the shape of the object.

---

It is alleged that when Einstein and his wife visited the Mount Wilson Observatory in California, Mrs. Einstein pointed to a particularly complex piece of equipment and asked its purpose. Their guide said that it was used to determine the shape of the universe. 'Oh,' she said, not at all impressed, 'my husband uses the back of an old envelope to work that out.'

Elsa Lowenthal Einstein, 1876–1936,  
German physicist



MyCodeHistory: 14 June 2015

All programming techniques that were used in the previous programs of the original Generative Design Attractors chapter are combined in a tool. I did not modify this tool program very much. I just used it to create images. But it was pretty difficult to find the right settings. After some periods of tweaking I found some. It is still not very clear what everything does (and sometimes doesn't) because the program itself is very large. The program seems to behave very different compared to the other programs because when you initialise the program it does not use the new settings. Later I found out that these settings can be changed by modifying the different sets of the program (set1, set2, set3, set4).

In this second range of variations I tried to experiment with the layers. But unfortunately I got an error: java.lang.RuntimeException: java.lang.OutOfMemoryError: Java heap space. It seems that its better to keep the line-weight at 1.00 Otherwise the program will not run smooth. But I've made the grid vertical instead of horizontal. Did some images with vertical lines only. It seems that you can make the lines thicker when you are ready with the editing process. When the image is ready you can adjust the thickness of the lines without any problem (I thought). Except when you are working in curves mode.

mediocre image quality. I did not find the images smooth enough. First of all I thought maybe render everything on a larger size: 1600 x 1600 instead of 800 x 800. I might than resize all images back to 800 x 800 because that is the image size I worked in until now. But fortunately I found a setting which gave me a better resolution within 800 x 800. These images are very crisp and deliver a nice feeling. Had to increase the maximum available memory of Processing to 2048 Mb though. But until now that seems to be no problem. By the way it works only with lines. You could use curves but they make the tool very slow.

I made a horizontal setup. Tried to make a compact chaos in a complex grid. But I could not find what I wanted (or liked). And I was bothered by vertical lines which kept on popping up in the image. Just by coincidence I bumped into an extremely large attractor setting. And that might ask for even more layers in other colours. Added two extra colours but the images are not getting any better. I could add more colours but I would rather bring in some unexpected qualities. Found another way to create interesting moiré patterns. Made some variations on that theme.

The first attempts which make use of the new settings are giving me a very organic feeling. But I also think that some images are way too complex. Anyway... this could be the right time to use some color. And maybe some more layers. I only used two layers until now. Surprised! The pdf's are working! But the .png's are delivering transparency so I have to adjust those in Photoshop. Changed image creation to jpg in the tool. Made some justifications to the setParas and the initial settings of the sliders of ControlP5. I know now how to get the adjustment right.

---

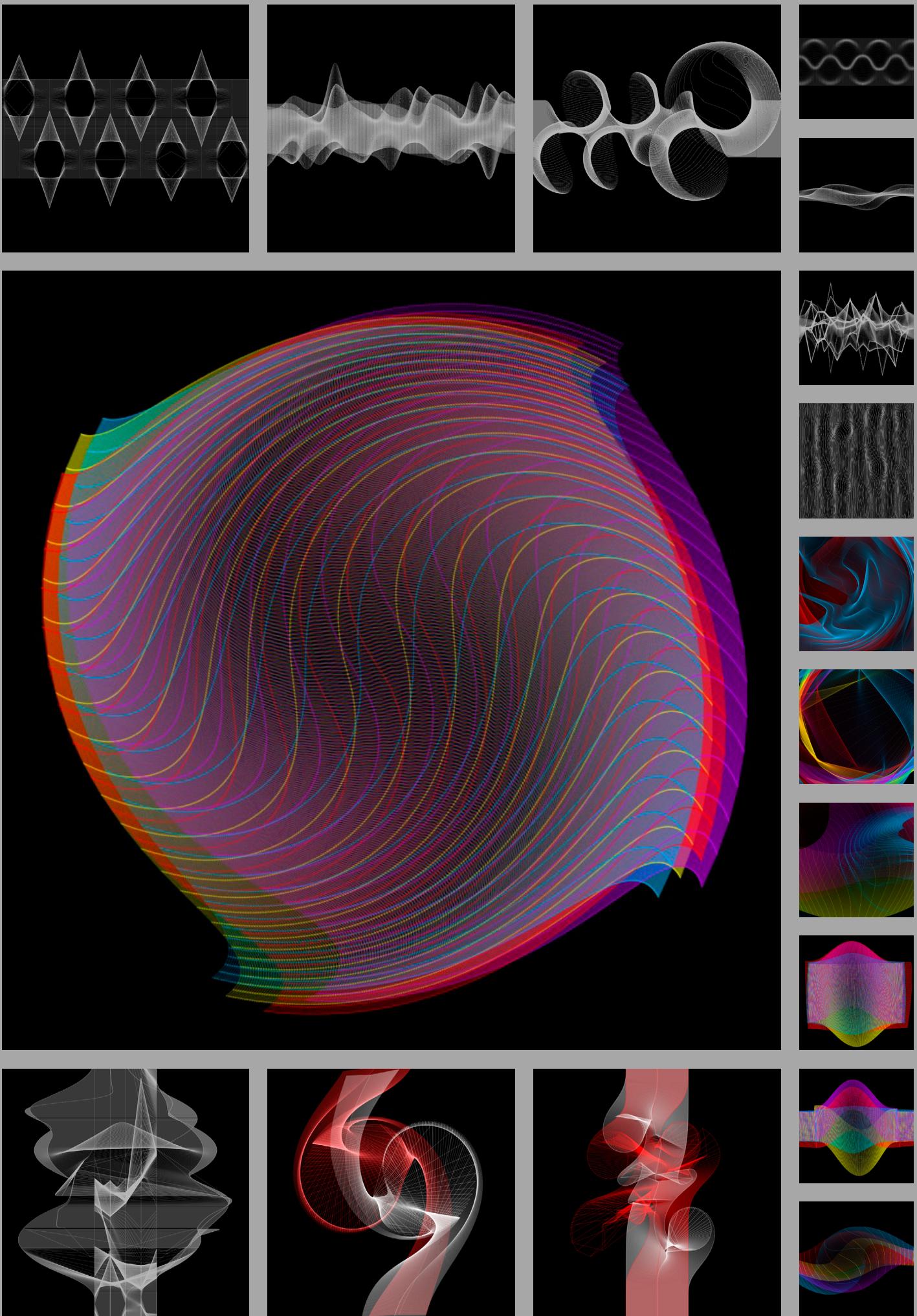
Looking back to this assignment I must admit that I haven't used the layers function very often. I used a maximum of 4 layers (each for a different color). And I have to stop working with this program before I will repeat myself. Although repetition seems to be very unlikely with this tool.

---

'The assumption that the new is much superior to older methods is widespread.'

David Edgerton: 'The Shock of the Old.'

Meanwhile I was a bit annoyed by the



MyCodeHistory: 18 June 2015

I did a lot of tweaking on the program's settings. But I just couldn't get the right the images from it. I added a zoom factor. Using that it is easier for me to position the cube at a size I really would like to see it. To get to know what is going on I lowered xCount, yCount and zCount to their lowest levels. Using the attractor it delivers just a random bunch of lines. Then I increased the gridStepX. Which, finally, did not make any sense at all. I found that xCount, yCount and zCount are counted twice. So when you move the slider to the number 1 you get a cube which exists out of 4 separate cubes. Another thing is that when I was finished with making the variations I noticed that I did not make much use of the attractor itself to modify the grid. I found that some settings of the grid were already interesting enough to make an interesting image.

After working with this sketch I noticed that I did not save my settings of the previous sketch. I was too much busy with creating. That happens once in a while. So I leave that for what it's good for and continue with the next variations. Still not happy with the image quality.

In the previous sketches I worked with a default setting which I have modified a bit. The next sketches I will work with the sets that are incorporated in the program. I give it a go with set1 (a setting that already was made available by the Generative Design people). First of all I think the line weight is too heavy. So I decreased that. And I increased the alpha channel. Added red to the set. Maybe a very obvious conclusion but I get the impression that you get the best images by doing as little as possible (which is (by the way) very difficult).

As I mentioned before I noticed that you can also make interesting images by not using the attractor at all. Just adding or subtracting grid-steps x, y and z gives interesting moiré patterns.

And it gives a lot of depth in the images. Because I found the image quality not good enough I rendered these images in a 2400 x 2400 resolution using Marius Watz's TileSaver. And it behaves still the same as in former sketches. If you run the program once it's working fine. But after refining the settings in the GUI it breaks the image down in four parts and puts it together again in a wrong order. Which I might eventually like even better although it's wrong. But I decide that later when I will put the images back together in Photoshop.

Continuing with set2. Found a setting with an improvement to the image quality. Should have known this much earlier. It's still not 100% but it is much better than what I had in the beginning. The thing I like seeing these variations is that they seem to be not computer generated. They have a kind of roughness in them which you see not very often.

Set3 is leading me to the individual pixels. This is really getting somewhere. At least I think I am working now on pixel level. That's why you might notice squares in some of the images. The program (or tool) evolves now more to a kind of structure or pattern generator than a tool which uses attractors in space. And at certain moments I do not know what I am looking at and at what angle and how close or how far. Good that I added that zoom function. It's very helpful.

Continuing with the last set (which is set4). Replaced the color-range by my color-range (which is an adaptation of the original one which was included in the tool itself). Working further with the color-range and trying to get some slightly small attractor manipulation in the image.

---

I think that the attractors in space should be used very careful. It is easy to make a big mess of it. It is very difficult to keep the chaos to acceptable limits.

---

Pope Boniface VII, thinking to employ Giotto, sent a messenger to the painter to ask for a sample of his work. With one continuous stroke of his hand, Giotto drew a perfect circle. When this was shown to the pope, he realised that Giotto was preeminent among painters of the time. (Professional artists do not find this anecdote persuasive evidence of Giotto's genius).

Giotto di Bondone, 1266–1337, Italian painter and architect.

