Nature Inspired Design

Evolutionary Art

Module Lecturer: Bob Hendley

Palita Lapanupat
Ming Zhu
Antonis Karayiannis
1. Introduction

Since the early days of computers, the idea of computer generated art was present. Later, the use of computers as tools had for creating art had a great impact on graphics, advertising, movies and various other fields. Even though sophisticated techniques were developed, either by recreating traditional drawing methods or by the use of virtual environments, computers were just a tool aiding development.

The first machine being able to draw actually dates back to the 19th century. At the time, an illusionist created an automatic mechanism capable of drawing simple images. It was called Zelda and it was a rudimentary mechanism made out of dented wheels. This was actually an achievement for its time and it managed to full many people that it had the ability to draw any possible image [1].

In our paper we will try to explore this “dream” of computer generated art by the assistance of evolutionary algorithms. Evolutionary algorithms are essentially one of the few techniques that can be used for the development of art by computers and maybe the best. In the pages to come we will examine fields that evolutionary art can be applied and various techniques that can be used. We will start by exploring two dimensional art, and the various ways to create it. Later on will go through three dimensional art, with a focus on landscape generation and animation mainly for character motion development. Finally we will take a look to fitness evaluation which is the main drawback of evolutionary art and try to find a way to overcome its problems.

2. Evolutionary Art in 2D

2.1 Genetic Algorithm [2]

In genetic algorithm, the list of parameters that represents each gene is called genotype. The genotype in biological systems can be DNA but in evolutionary art it can be many representations, for example, binary string. Each individual can be referred to phenotype. There is a population which contains a set of individuals or designs. Genetic algorithms usually have fitness function to measure the fitness of individuals. For the evolutionary art, it is difficult to evaluate the aesthetic value of the images. Therefore, these fitness values are measured by using human evaluation. This method is called as interactive evolutionary system. For generating a new generation, two individuals must be selected by the process based on the fitness. In this interactive evolutionary system, the population is generated randomly and the user will be the one selecting the interested
images to do reproduction process and generate the new generation. After selecting the two individuals or parents, crossover method is used to create an offspring, which takes some parts of gene from its father and mother. It also gets mutated by randomly changing some parts of gene.

2.2 Genetic Programming [2,3]

In evolutionary art, there is another way to represent the genotype. A mathematic expression has been used lately. Each individual is an image formula. It can represent as the tree structure that has the mathematic operators as a node. The tree nodes can be divided into two types of node, the function set and terminal set. The function set can be any mathematic operators and terminal set can be the variables or any constants. If the tree evaluates each pixel of image, the terminal set can be pixel coordinate positions, x and y. After generating the population and selecting parents, offspring is created by using crossover method. The two tree structures will be combined. For the crossover method, it usually exchange subtrees between the individuals. For mutation method, there is a change in the genotype. It may change from one function to another one or change from constant to any variables.

![Example of crossover](image1.png)

![Example of mutation](image2.png)

Figure 1. (a) Example of crossover (b) Example of mutation
2.3 Representation

There are many types of representation in evolutionary art. One of them is pixel representation. The pixel coordinate positions, x and y, are used. In genetic programming [3], each image has its own equation with the coordinate positions as variables, such as, \( x + 2y \). The tree structure has the function set, which are mathematic operator, and terminal set, which are general numbers, x and y.

For the grey image, when the values of x and y are close to zero, the result from the equation will be close to zero. This makes that pixel presents black color. In the other hand, the color of pixel will be lighter, if x and y increase as shown in the below figure.

![Figure 2. Grey image gotten from x + 2y](image)

For color image, each pixel contains an RGB (Red, Green, Blue) value. Therefore, a 3d-vector is used instead of the constant. For example, the equation, \( x + 2y \), changes the constant 2 to a vector [1,2,3]. The equation for this one will be [\( x + 1y \), \( x + 2y \), \( x + 3y \)].

There is an example for evolutionary art by using pixel-representation, which evolves 2D textures [4]. The function set of this is a texture formula that is combined from the mathematic operators and some special operators, such as, cloud, marble, noise. These special operators create a wide variety of textures. The example of texture formula is shown in the following figure.
Another type of representation is by using a tree-coding which corresponds to any shapes, as can be seen from evolutionary of Mondriaan Art [5]. The Mondriaan Art, which was created by Pieter Cornelis Mondriaan, contains geometric shapes using only straight black lines and using only five colors, which are white, black, red, blue and yellow, as shown in the below figure.

For each image, it is represented by a tree structure and two numbers for the height and width for the image. The tree structure is also same with pixel-representation that has the function set and terminal set. However, the function set is not mathematic operators. The functions are horizontal H and vertical V that tell how the image splits up. For the terminal set, it indicates the color which fills in that part of image. A example of this tree structure is shown below.
For evolutionary of Mondriaan Art, it takes a long time to get a good image and also requires responses from human. It is hard to find fitness function that tell how good or bad of the image.

2.4 Fractals

Benoit Mandelbrot explained that a fractal is “a rough or fragmented geometric shape that can be split into parts, each of which is (at least approximately) a reduced-size copy of the whole.” [6] Fractals are objects that have self-similarity at various scales. When looking at little pieces of that object, each piece has the same shape like itself with identical geometric shape.
This figure, called as the Sierpinski triangle or Sierpinski gasket [7], shows a standard strictly self-similar fractal. When you draw a triangle that has equal sides, divide the triangle into four sub triangles which has $\frac{1}{2}$ height and $\frac{1}{2}$ width from the original one and repeat this process.

An interesting example is the Mandelbrot set [8]. It is a mathemetic set which does not contain the real number but complex number. This complex number has a real part, which is an ordinary number, and an imaginary part that a number with a special character called i, such as, $2 + 4i$. This Mandelbrot set tests on equation $Z = Z^2 + C$, where C is a constant number. The below figure shows the Mandelbrot set which is the black shape in the middle and the colors in the image are not related to the set. These colors help to highlight parts of this set.

![Figure 7. The Mandelbrot set](image)

This Mandelbrot set also has a self-similarity property by zooming in a round feature that can be shown in the following figure.
3. Evolutionary Art in 3D

3.1. Modelling 3D Terrain

Generate Landscape using genetic programming as a new research area which involves genetic programming, landscape ecology, visualisation techniques and spatial statistics. Coupled with rapid development in genetic programming, terrain generation has becoming a hot topic in this area. Terrain generation is used in a broad range of applications in many fields. Such as computer animation, digital art, video games, architecture etc. This chapter mainly represents how 3D landscape generating techniques are developed in this area. Section 3.1.1 introduces the common method to representing terrains, its advantages and disadvantages. The ideal algorithm for generating terrain, other algorithms and the genetic algorithm are presented in section 3.1.2. Section 3.1.3 presents different types of terrain modelling techniques. Section 3.1.4 presents a new application which overcome some common problem and can produce multiple different features.

3.1.1 Terrain Representation

A terrain, a digital ground surface, can be represented in many ways. The most common used method is height maps. A height map is a scalar function of two variables, every a pair of coordinates are related to an elevation value h, as shown in the equation.
It is used to store values that for displaying in 3D graphics, in terrain, the height maps are finally converted into mesh.

\[ h = f(x, y) \quad (1) \]

In practice, a height map is implemented as a two dimension, rectangular grids of height values. It is represented as a 2D arrays filled with elevation values. A height map contains one channel interpreted as a distance of displacement or “height” from the “floor” of a surface and sometimes visualised as luma of a grayscale image, with black representing minimum height and white representing maximum height. See the example figures, figure 9 and figure 10 [10].

Figure 9. A height map
Figure 10. The same height map converted into a 3D mesh with anim8or

**Flaws and Virtues**

The height map has a main drawback which is when it comes to represent structure that has multiple heights for the same pair of coordinate \((x, y)\), it shows poor ability to handle it. Which means heights maps are not suitable for representing cases like caves, cliffs and other structures in which there are vertical surface but not only horizontal surfaces. Another issue is that it cannot handle a terrain with complex details easily for the reason that it only has a finite uniform resolution. The trade-off between high resolution and high scale features becomes critical.

The big advantages of height maps are it has a simple structure which gives an ability of optimising the operation such as rendering and object collision detection. Height maps can be treated as a luma grey scale image if the value of it is normalised. It means
image processing and other computer techniques can be used to modify and analyze terrains generated using height map, which makes it much easier when generate terrain models.

### 3.1.2 Terrain generating algorithm

A list of key points of an ideal algorithm for generating terrains was proposed by R. Saunders. Such as it should have low requirement for human input, easy to control, fast enough to adapt real-time application, flexible enough to support new types of terrains, and capable of generating recognisable terrains with different features.

Most common methods can not generate terrains automatically and have a high dependency on human skills. It takes huge amount of time and efforts to get an acceptable result. On the other hand, other methods which are able to generate terrains automatically only produce a narrow range of terrain types. But evolutionary art can solve those issues.

### 3.1.3 Terrain generation

Currently the traditional terrain generation techniques can be divided into three main categories: measuring, modelling and procedural. Measuring means generate terrain using data that are derived from real-world measurements, thus it can produce the most realistic terrains compared with other techniques. The case modelling is generating terrain using 3D modelling tools such as Maya and 3D Max, etc. So it has the most flexibility in the price of time consuming and labor costing. The last one, procedural method is to generate terrains using programming, this method also can produce high realistic terrain but it require the designers to have a deep understanding and experience in physical laws in order to use those tools effectively.

Nowadays, as a new trend, evolutionary techniques are applied to many important fields. Teong ong et al. proposed an evolutionary design optimisation technique to generate terrains and they gained remarkable achievements. What they have done is they applied genetic algorithm to transform height maps in order to conform them to required features. Because of the equivalence between height and grayscale image, it is possible to apply the same principle of evolutionary art to terrain generation.
3.1.4 Genetic Terrain Programming

As addressed above, most terrain generating techniques have their flaws and virtues. The strength of this new technique is it not only overcome the weakness of the other common methods but also can allow the designers to create aesthetic terrains.

3.1.4.1 General view

This application uses aesthetic evolutionary design with genetic programming, where the phenotypes are represented as height maps. By using an evolution guided approach, the users can choose what kind of solution they want according to their aesthetic appeal. The diversity of solutions is realised by changing the function set and terminal set within the programming tree. Through this interactive evolutionary method, both realistic and aesthetic terrains can be generated incorporated to user’s specific requirements. Finally the outcomes are mathematic TP expression, which can be used to automatically generate with different LODs (Level of Details) but with the same features that specified by the designers.

3.1.4.2 GP algorithm

In this GP, they used a syntax tree rather than lines of code. (Fig 11) The initial populations are generated randomly from the available primitives, with tree depth limited with 6 and the initial population size with 12. Number of generations is decided by the designer, who can also decide which individual to choose to generate next population according to his requirement.

![Figure 11. Tree representation max(x+x, x+3*y)](image-url)
If two individuals are selected then 90% sub-tree crossover and 10% mutation are applied. Otherwise if just one individual is selected, only mutation is applied with 50% probability. The program can be stopped any time due to the decision of user. Fitness is defined by the terrain feature and aesthetic requirement, and the best individual is chosen by the designer.

3.1.4.3 GenTP

This GenTP is developed by Miguel Monteiro de Sousa Frade, it is developed with GPLAB, an open source toolbox for MATLAB. This software mainly has three functional modules: the main interface, analyse module and the generation module.

The main interface is where user can get an overview of terrains and then select individuals to make next generation. However what is inconvenient is that it is very difficult to see all the features of terrains especially the details of terrain, due to the angle displayed. To get over this problem, analyse module is introduced to this application. In the analyze module, it allow users to have a more detailed view of the terrain selected. User can rotate, zoom in or out and change the resolution. In this case, user can pick the specific terrains which their requirement. The generation module is used for the generations of height maps, the height maps can be saved as VRML 2.0 files so it can be imported into other 3D software such as 3D Max or Maya. Below are some figures for the GUI of this software.

![Figure 12. GUI for the main interface](image-url)
4. Evolutionary Art and Animation

Evolutionary art with animation follows all the principles mentioned in our paper up until now. Either we are talking about fractal evolution or a shape representation algorithm that produces art or any other possible kind, animation is possible. All we have to do is incorporate time, one more temporal dimension into our algorithm [11].

Animation as an art first started in the classic days of Disney and Warner Brothers using animation cels. In the later years computational techniques where introduced and animation was aided from physics simulation, body interaction, motor control and a lot others to help motion generation. In these kinds of systems the animators operate at a task level where they set some general details to achieve a goal and the character (an autonomous agent) takes care of the details needed to achieve the task [12].

In our field of interest this is where evolutionary animation found a solid ground to develop. There is absolutely no surprise that evolutionary algorithms entered the world of animation and especially for characters as they are good at finding solutions at large search spaces and since they are biologically inspired the results produced are more realistic when compared to any other approach. The outcome of the algorithms cannot be considered directly as art but as as a tool that aids the development. An approach of this kind can allow either an interactive fitness evaluation or an automatic one. The first one
depends on human evaluation and the second evaluates the agent on how well the task is performed. An example of this kind of character animation performed is Breeve which is based on Artificial life. In Breeve creatures created evolve their body and behavior which usually controlled by a neural network [13].

![Figure 14. An artificial creature in Breeve.](image)

5. Automatic Fitness Evaluation.

For evolutionary art, fitness evaluation seems to be the weakest link. In all the successful algorithms up until now, fitness evaluation was done entirely by humans. Humans ability to evaluate art in its every aspect extremely better than any algorithm right now, but unfortunately it is extremely slow as well. This creates a bottleneck for communication by relating the human world with the computers. In the next few paragraphs we will research the true meanings of aesthetics, study the beautiful through mathematics and by this way hope to establish the knowledge of what art is and how computers can evaluate it.

Following the notion that mathematics is an art we can see that there is a structural beauty hidden in them. Mathematics follow a logic so that each conclusion is drawn from previous statements to create some kind of proof. Each one of those proofs is a real piece of art that if we somehow convert it using a visual interpretation, the beauty will become more obvious and at the same time we will be closer to what we consider traditional art [14].
If art and beauty can be seen through mathematical proofs and visual interpretation, we can use this as a medium that helps for the creation of art. A first very simple example of this can be the eutrigon theorem. The eutrigon theorem states that the area of any eutrigon (*white triangles*) is equal to the sum of the areas of the equilateral triangles on its legs, minus the area of the equilateral triangle on its hypotenuse [15]. If we describe this into an equation it is difficult to understand the aesthetics values hidden but the visual proof of this as presented below can be considered as aesthetic.

![Visual proof of the eutrigon theorem](image)

**Figure 15. The visual proof of the eutrigon theorem.**

Furthermore beyond Euclidean elements and simple geometry we find another example that represents art as a painting following hyperbolic geometry. The hyperbolic geometry first discovered by Nicolai Lobachevsky and Janos Bolyai offers an alternative to the fifth postulate of Euclidean geometry. As the fifth postulate indicates in Euclidean geometry, for a line *l* and a point *P* not on *l* there is only one line on *P* that does not cross *l* given that the two lines are parallel. In Hyperbolic geometry now the fifth postulate is modified so that there are at least two distinct lines through *P* that do not intersect [16]. This model of geometry has a very strong aesthetic influence to the human eye especially when using a variety of repeating patterns possible in the hyperbolic plane. The famous Dutch painter M. C. Escher uses the hyperbolic plane along with repeating patterns to create art with high aesthetics based on mathematics [14] as seen below.
Now after explaining the aesthetics in mathematics and their visuals interpretations we must try to understand how those can be used in fitness evaluation. Although art can be perceived in many ways even through feelings and emotions we have selected to present this field that focuses around mathematical aesthetics because it can be easily represented using computers. If we want to evaluate evolvable art one, we can now let ourselves into the hands of the beauty of mathematics to form a fitness landscape. That fitness landscape can be shaped by different formulas and each individual can be considered aesthetic the more it fits the credentials that prove those formulas. To explain this notion we will try to give an example that includes the two examples mentioned above. So in the case we have an algorithm that uses some form of shape representation to evolve art, considering a fitness landscape made from the eutrigon theorem and hyperbolic geometry as visualized by M. C. Escher, we can possibly have the following two individuals:

- **First individual**: A triangle contains circles which use some repeating pattern and follow the hyperbolic plane.
- **Second Individual**: A circle follows the hyperbolic plane and uses as a repeating pattern triangles that provide a visual proof of the eutrigon theorem.

If we now examine the two individuals we can see that the second one utilizes both the eutrigon theorem and hyperbolic geometry where the first one manages to utilize only hyperbolic geometry. So in a way we can evaluate the fitness by concluding that the
second individual is the best. It is noteworthy though that the use of one theorem might not be compatible with another, so our fitness evaluation can be a proportion of how close the individual is to the proof of a theorem.

Figure 17. Second individual - Hyperbolic geometry with eutrigon theorem as pattern.

Of course all these are easier said than done, we should keep in mind that fitness evaluation is the bottleneck of evolutionary art and not much essential research exists on it, so what we have mentioned above are really suggestions taken from other fields that have the credentials to apply in evolutionary art and maybe produce some good results on automatic fitness evaluation.

6. Applications.

6.1 Evolutionary Jewelry.

The use of evolutionary computation to design jewelry arises from the need for faster production rates especially in mass production. Traditional development requires design and craftsmanship which are both time consuming. Though CAD systems have been developed to aid the designers, those were not so successful so the field turned into Evolutionary Algorithms. The most common Evolutionary Algorithms are used in Evolutionary Jewelry is to evolve Iterated Function Systems (IFS) fractals for ornaments using hybrid methods between EP, ES, GA, and GP [17]. Because IFS are used to create
fractals the aesthetic characteristics of a design are the golden ratio, rotational symmetry, logarithmic spiral symmetry, mirror symmetry, complexity, compactness and connectivity, unpredictability and fractal dimensions [17]. An example of an algorithm for evolutionary jewelry design taken from [17] is the one below:

- **STEP 1**: Evolutionary process initializes by uniform-randomly selecting a set of individuals in the IFS chromosome library.
- **STEP 2**: Apply the mutation operator to the individuals following to the mutation probability and then fill all individuals and their mutated versions to the population.
- **STEP 3**: Apply the crossover operator to the population regarding to the crossover probability, to produce a new batch of offspring. Then add them to the population.
- **STEP 4**: In mapping process, map the individuals (genotypes) which are now in forms of chromosome strings of IFS codes to fractals (phenotypes).
- **STEP 5**: Go to the first evaluation—“Morphological Fitness Function”. Evaluate all individuals for their morphologies. Screen incompact and disconnected art forms out of process.
- **STEP 6**: Select the individuals considering their morphological fitness, which explain their compactness and connectivity.
- **STEP 7**: Go to the second evaluation—“Aesthetic Fitness Function”. The survival individuals are evaluated for their aesthetic fitnesses.
- **STEP 9**: Select a set of the best individuals regarding to their aesthetic fitnesses. The number of selected individuals depends on population size.
- **STEP 10**: Repeat STEP 2 to STEP 9 until the system achieves the termination criteria.

Algorithm 1. Evolutionary algorithm for jewelry design.

6.2 Electric sheep [18]

Electric sheep is a distributed computer project founded by Scott Draves. It evolves fractal animation that displays as a screensaver for each computer. Each animation contains 128 frames and is called as a sheep. Each image created by Electric Sheep is a fractal flame, a kind of iterated function system, and contains about 160 floating-point numbers. The electric sheep has a client/server architecture which the client or user can involve to this project. The client can create their own sheep and upload it to the server. Furthermore, they can download other sheep and use them as screensaver. They can also
vote for the sheep they like or decrease the rating for any sheep, which the votes will be the fitness for that sheep. The more vote it gets, the more chance it will survive.

Figure 18. Example of sheep died within 24 hours after got only 1 vote.

In genetic algorithm method, the parents will be chosen randomly from the population weighted by rating. After picking the parents, they will do crossover operation to create an offspring. There are two methods for this crossover operation. One method is that the offspring will pick genes from its parents alternatively. Another one does pair-wise linear interpolation between two parents. For mutation operator, there are several methods, such as, adding noise, adding symmetry.

Because a large number of sheep have been born, the sheep with the lowest rating is killed every time a new sheep is born. If some sheep have the same worst rating, the oldest will be killed.

Figure 19. Example of sheep that got high votes [19].
6.3 Artifacts

Interactive Evolutionary Computation has penetrated many fields like art, architecture, animation, medical science, robotics, etc. However, only a few efforts have been put in areas of designing artifacts like machines, vehicles or aircrafts [20]. The reason for this is because this filed is heavily rely on automated optimization process and too reluctance to accept new methods. Even so, accelerating interests have been attracted to this field. An Aircraft design example is introduced here to show how IEC will work in engineering.

When we refer to IEC, there are two categories: broad and narrow due to the definition by Takagi. [21] Narrow means the fitness evaluation is only accessed by human input while Broad has numerical fitness but human can use this to guide the evolutionary process. The advantage of this aircraft design here is it breaches the two historical categories mentioned above. It is true that many designers are making decisions through pure feel and intuitions rather than analysis. [20] While engineers are still heavily relying on analyze tools for fitness generation. This approach allow automatic fitness calculation through analysis as well as selection and fitness assignment by the human designers directly.[20] A process outline from [20] can explain better:

![Figure 20. Process outline.](image)
7. Conclusion

Through these few pages of our research we examined many of the aspects of evolutionary art as well as many of its applications. From an artistic point of view we can see that this kind of art can produce some very astonishing masterpieces. One can see the limitations though as many of the images created have a great resemblance between them according to the technique used to develop them. This is not necessarily a bad thing as all of the great artists have their own signature style that makes them different from the rest. Still, this proves we have a long way to go in order to achieve our desired goals.

On the other hand now, we can see that evolutionary art has been implemented by various different fields especially in 3D graphics and character animation. Outside the computer and software industry it also has a wide use in jewelry design development for ornaments where the development requires efficient out of the box ideas.

Furthermore we noticed in almost all of our examples that the development of art using evolutionary algorithms is time costly. That is the main drawback of evolutionary art and it is mainly due to interactive evaluation and the large search spaces our algorithms have to explore. We tried to address the problem of interactive evaluation with automatic fitness evaluation using mathematical aesthetics, a promising technique but not so widely used.

In our opinion future research should focus on ways to improve solutions provided to the industry. In this way the interest to the field will rise even more and with the wide use innovations will come. Nevertheless, through our research we tasted the fruits of the captivating world of evolutionary art and we can see that humans long lasting “dream” for computer generated art is already here and it looks promising.
References


[3] Qinying Xu, Daryl D'Souza and Vic Ciesielski, Evolving images for Entertainment, School of CS and IT, RMIT University, Australia.


[19] Spotworks LLC, Electric Sheep, Accessed online: http://community.electricsheep.org/about
