



CHALMERS
UNIVERSITY OF TECHNOLOGY



The Blind Watchmaker Revived: Evolving Human Faces

Master's thesis in Complex Adaptive Systems

Alexandre Cabrita

The Blind Watchmaker Revived: Evolving Human Faces

ALEXANDRE CABRITA



Department of Space, Earth and Environment, Physical Resource Theory
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2019

The Blind Watchmaker Revived: Evolving Human Faces
Alexandre Cabrita

Supervisor: Claes Andersson, Department of Space, Earth and Environment, Physical Resource
Theory, Chalmers University of Technology

Examiner: Claes Andersson, Department of Space, Earth and Environment, Physical Resource
Theory, Chalmers University of Technology

Department of Space, Earth and Environment
Physical Resource Theory
Chalmers University of Technology
SE-412 96 Gothenburg

Cover: Generations 1, 9, 11, 21, 31, 37, 48, 51, 61, 71, 75, 78, 89 and 100 of an evolution of the
face model using the interactive evolutionary algorithm developed.

Typeset in L^AT_EX
Gothenburg, Sweden 2019

Abstract

Inspired by the work of Richard Dawkins "The Blind Watchmaker", an interactive evolutionary algorithm has been created to evolve human faces. The face model used was built from the freely available software MakeHuman. The interactive evolutionary algorithm was written using Blender's Python interface since this allowed the control of the face through scripting. After encoding 14 different facial features in a chromosome, it was possible to semiautomate the evolution of the face, leaving the user with the power to guide the evolution. The result is a program that randomly generates an initial population of faces, allows the user to select the preferred one, creates a new population through random mutations and lets the user terminate the running program as soon as a face represents a final solution. The processes of user selection and random mutations allowed the modeling of different faces having distinct facial features.

Contents

1	Introduction	1
2	Theory	3
2.1	Evolutionary Algorithm	3
2.2	Evolutionary Art	4
2.3	Face modeling	6
3	Methodology	7
3.1	Face representation	7
3.2	Interactive evolutionary algorithm	8
4	Results	11
5	Discussion	15
6	Conclusion	17
	Bibliography	20
	List of Figures	22

Chapter 1

Introduction

Nature has long influenced artists in the creation of their work, with beautiful landscapes, flowers and herds of animals serving as sources of inspiration. Nowadays, with the advent of modern computers, new ways of representing and depicting nature have also been born. This can be seen, for example, through realistic virtual models of landscapes in movies or pictures of galaxies around us. However, in most cases, the computer is only used as tool to create and display images and nature the source of inspiration of the art work. Nonetheless, there are areas of research attempting to use the computer's ability for automation, to construct art work that results from operations mimicking natural processes. One example of that is **Evolutionary Art**.

Evolutionary art is the result of creating or manipulating images using evolutionary algorithms, meaning that in the presence of an evolutionary pressure, an image can be evolved using the same mechanisms present in natural evolution. Therefore, art work becomes the result of successive steps of selection, reproduction and mutation, combined with a set of rules and instructions written in computer code. Since this technique of art creation flows in a rather automatic fashion, the artist itself is usually left with the power to steer the evolution, becoming the selective agent behind the entire process. However, even selection can be automatized if, for example, a suitable fitness function can be constructed [9].

One of the earliest works in the field has been accomplished by the biologist Richard Dawkins. In the book **The Blind Watchmaker**, Dawkins showed that it was possible to construct a computer program that evolved simple two-dimensional objects using mechanisms of selection and mutation [3]. After consecutive evolutionary steps, Dawkins observed that more complex structures resembling living organisms would eventually emerge. These structures, which looked like insects, spiders, foxes and other animals, were named **Biomorphs**. Even though Richard Dawkins wrote this book to make a case for the power of natural evolution, where gradual mutations and natural selection are the driving forces behind the emergence of complex life, the Biomorphs marked the beginning of what has been later called evolutionary art.

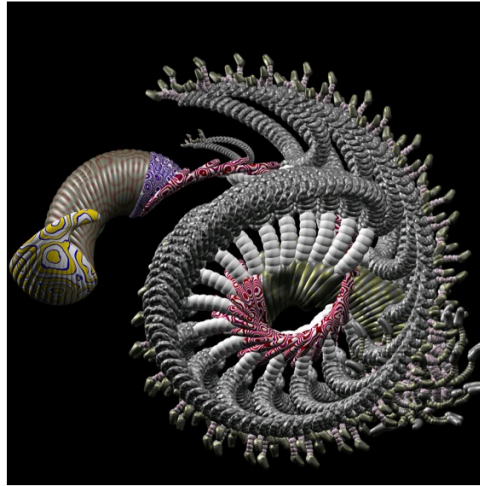


Figure 1.1: Image created by William Latham using the Mutator program [19].

Another pioneer in the field was William Latham. Coming from a background in art, Latham managed to join a strong interest in the evolution of shapes with computer code to develop a new style of complex sculptures that emerged from simple objects such as cones, spheres and tori [8]. As in the Biomorphs' case, the shapes would gradually change each iteration through evolutionary operators and a set of rules. However, the level of complexity of Latham's images exceeded any previous work, managing to create highly intricate three-dimensional figures through programs he worked on such as **FormGrow** and **Mutator** [19], see figure 1.1.

Karl Sims, inspired by the work of Richard Dawkins, also started creating his own art work through these biological inspired mechanisms. In his projects, Sims decided to develop pictures containing abstract forms that would be obtained through the evolution of certain genotypes using selection, reproduction and crossover [16], see figure 2.1.

The field has since progressed, and applications have been found in design [10] and architecture [7]. Other artists and scientists, such as Jon McCormack [1], Steven Rooke [17] and Penousal Machado [15] are also good examples of individuals who developed work in the field of evolutionary art.

Another research area of great interest in computer graphics is the computer representation and animation of human faces. In fact, the first three-dimensional parameterization of a human face was first developed in 1972 by Frederick Parke [14]. Since then, the field has considerably progressed and it has found many applications in the creation of real looking characters for the film and game industry [4].

The goal of the work presented here was to combine the field of evolutionary art with the computer representation of human faces. More specifically, a computer model of a human face has been used together with an evolutionary algorithm, to shape and obtain new faces. In the chapters ahead, the reader can find the theory, the methodology and a discussion of the work realized.

Chapter 2

Theory

2.1 Evolutionary Algorithm

An **Evolutionary algorithm** is a type of algorithm that works, even if vaguely, through the same mechanisms occurring in biological evolution. Within this class of algorithms, the most common is called **Genetic algorithm**. As in natural evolution, the processes of selection, reproduction and mutation are present in a genetic algorithm and, even though these are much simplified as compared to the real natural processes, they can solve a wide range of optimization problems that are hard to pose in a way that permits the use of formal methods [18].

In a genetic algorithm, in order to optimize an arbitrary function $f(x_1, x_2, \dots, x_n)$, it is first necessary to encode the values of the variables (x_1, x_2, \dots, x_n) in a string of numbers called chromosome. Following the same biological terminology, each position in the chromosome is thus called a gene. There are many schemes to encode the information in the chromosome, however, it is common to use a binary encoding, where each gene can take either the value 0 or 1 [18].

The initialization of the algorithm is done through the creation of a set of chromosomes that make up a population. Usually, each initial gene value in a chromosome is set at random and this group of starting chromosomes is called the first generation. In analogy to nature, there is a one to one correspondence between a chromosome and an individual. The individual here is the set of variables (x_1, x_2, \dots, x_n) that represent the function $f(x_1, x_2, \dots, x_n)$. In order to get to the individual it is necessary to first decode the information given in the chromosome. There are multiple ways to decode this but one common way to do so, is to split a chromosome in equal parts, read each part as a binary number and convert it into a decimal number [18].

When a population has been obtained, its members must be evaluated so that a fitness function can be applied. Depending on the problem and on the existence of an objective function, this

evaluation can be done in different ways. The main idea is to attribute to each individual a fitness value measuring how well it represents the solution of the problem at hand. Therefore, the fitness is the quantity to be maximized in a genetic algorithm [18].

To obtain the next generation, the population must go through the processes of selection, reproduction, mutation and replacement. The selection step is detrimental for the success of the algorithm as it is the required step to assign the parents of the next generation. One might be tempted to think that selecting only the individuals with the highest fitness is the best way to ensure that the optimal solution is found. However, it is usually good practice to ensure some stochasticity in the selection process. Thus, even though individuals with high fitness values are favored, chances are that other individuals can also prosper. Two common selection schemes are roulette wheel selection and tournament selection [18].

Selection is then followed by reproduction. If cross-over is realized, the genetic material of the selected individuals is then mixed. Usually this is done by picking a random point of the chromosomes of two selected individuals and divide them into two parts. The first part of one chromosome is mixed with the second part of the other chromosome and the second part of the first chromosome is mixed with the first part of the second chromosome. Moreover, the newly generated chromosomes must go through the process of mutation. Even though mutations in nature correspond to errors in the replication of DNA, this step is fundamental in a genetic algorithm to ensure that the entire search-space can be explored. Therefore, each chromosome goes through a set of mutations where each gene can randomly change its value according to some predefined probability. The replacement of the old population with the new one can be done entirely by substituting all previous individuals with the new ones or it can be done partially, where some individuals from the previous generation can remain [18].

The processes of selection, reproduction, mutation and replacement are then repeated for multiple generations until a satisfied solution for the problem at hand has been obtained. For a more detailed description of each step and more information about evolutionary algorithms one can check [18].

2.2 Evolutionary Art

The processes described in the previous section are powerful tools to solve a wide range of mathematical problems in optimization. However, they could also be used to evolve shapes and images and therefore, create **Evolutionary Art**. Although it is sometimes possible to construct a fitness function to conceive evolutionary art [9], most cases require a subjective judgement. In the absence of a mathematical description of subjectivity, it is usually required that a person commands the path of the evolution according to his or her personal preferences.



Figure 2.1: Example of Evolutionary Art created by Karl Sims [16].

This combination between an evolutionary algorithm and a constant assessment by a user, results in a type of algorithm named **Interactive evolutionary algorithm**.

Interactive evolutionary algorithms can be useful, for example, in design [18]. Besides the objective parameters that could be optimized when designing a product, such as weight, length, material density, etc., there are subjective features that require some type of optimization too. Properties like colour combination, shape of certain components and the overall visual-appeal of an object. These cannot be described in a perfectly defined mathematical landscape and require a person to constantly assess and steer the optimization process. Interactive evolutionary algorithms are then useful for these tasks since they merge the inherent optimization processes from evolutionary algorithms with the freedom to control the evolution path according to the user's preferences.

As in a genetic algorithm, in an interactive evolutionary algorithm the chromosome is the central component that encodes any properties of an image. Again, the first generation of possible solutions is usually realized by creating chromosomes at random, each having their corresponding images. The user selects the picture that is found to be more desirable and its chromosome serves as the parent of the next generation. The selected parent is then replicated and mutated, giving rise to a new set of chromosomes. The parent image is then placed in the center, surrounded by the images encoded by the mutated chromosomes. The processes of user selection and random mutations is carried out for multiple generations until the final work is achieved [18]. The algorithm so far described is the simplest form of an evolutionary interactive algorithm as it uses user selection and random mutations as the only evolutionary operators. It is possible however, to use crossover together with more complex techniques of fitness assignment in this type of algorithms. Nevertheless, it was decided that for this project, the simplest form an interactive evolutionary algorithm would be used.

This simple implementation was used by Richard Dawkins in the creation of the Biomorphs [3]. In this work, each image was defined by a chromosome of 9 genes and although the initial images showed no apparent similitude to any life form, after successive rounds of user selection and random mutations, they quickly evolved towards shapes resembling insects, spiders and animals.

Although the Biomorphs have been the first clear work of what is now called evolutionary art, many other scientists and artists applied the same evolutionary techniques to create art work, see chapter 1. Furthermore, evolutionary music has also been developed [12] and applications of interactive evolutionary algorithms have been found in fashion design [10], architecture [7] and robotics [6].

2.3 Face modeling

The field of computer graphics has long been interested in how to model human faces. As mentioned earlier (see chapter 1), the first parameterization of a human face has been achieved by Frederick Parke in 1972 [14]. Since then, many other models have been developed [4]. The field of face modeling can be further split into the geometric representation of the face and the animation of the face. Here, we are only concerned with first.

Realistic models of human faces require either 3D laser scans, multiple view pictures of a real face or a data set of human faces to build a model based on a linear combination of human faces [4]. However, for simplicity, the work realized here was done using a publicly available face model that could be manipulated using a free software called **Blender**.

Blender is a free open source software for 3D computer graphics. It is a software commonly used to create visual effects, video games, animations in films and models for 3D printers [2]. Its tools include 3D modelling, texturing, physics simulations, sculpting, animating, rendering and video editing. However, the main features that made Blender the choice software for this project was its Python scripting interface together with its ability to import **MakeHuman** models directly.

MakeHuman is another open source software developed to generate and model 3D humanoid models [11]. Its models can be easily exported and used in other computer programs such as Blender. MakeHuman starts with a base model of an androgen human that can then be manipulated to create different characters. In the work presented here, MakeHuman was used solely to extract its neutral model and exported it to Blender so that it could then be manipulated using Blender's Python interface.

Chapter 3

Methodology

3.1 Face representation

The face model used in this project was exported directly from MakeHuman where a humanoid model of the entire human body is immediately assembled upon starting the program, see figure 3.1. In order to work with this model on Blender it is first necessary to export it as a MHX2 file and install the necessary plugins that allow the use of MakeHuman models in Blender [13]. No changes were executed using MakeHuman and thus, the original and neutral MakeHuman model was the base for the rest of the project, see figure 3.1.

The first step after having the model imported to Blender was to remove the vertices in the mesh that did not correspond to the head. This was done manually by carefully selecting and removing mesh points. Only the vertices and polygons that made up the head were left untouched. Moreover, the eyes came disconnected from the rest of the head. However, it was more convenient to work with a single object than with two separate ones. Therefore, the eyes and the rest of the

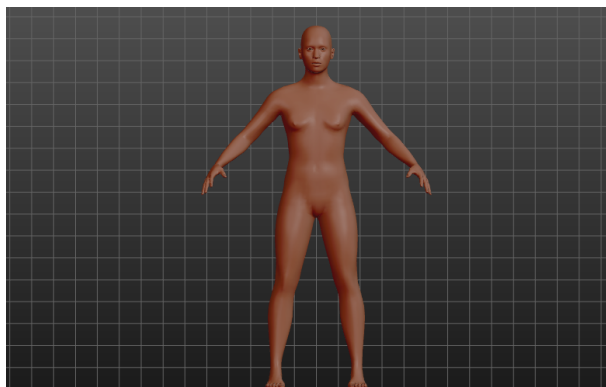


Figure 3.1: MakeHuman’s starting model. This was the base model for the face representation used in this project.



Figure 3.2: The base face after rendering the image.

head were joint into a single object. After rendering the image, one obtains the face showed in figure 3.2.

The second step was to understand how each component of the face could be independently manipulated. The solution to this problem was to use a Blender functionality called vertex groups. This allows an object to be broken down into multiple parts without losing its integrity. Therefore, assigning each part of the face to a separate vertex group was a fundamental step for an individual component modification. The process of selection was done manually by carefully selecting the vertices that made up each component.

However, in order to create these groups, it was first important to have an idea about which components could create visible changes in the overall face appearance. After multiple trials and thinking it was concluded that the different face components to be used were the nose, lips, eyes, ears, cheeks, chin, brow ridge, forehead and the overall size of the head. Some of the face components were assigned to two different vertex groups depending on their modes of change. For example, different vertices of the nose were used to change its length and its width. The last face component implemented was the skin colour. This was manipulated not through the creation of a vertex group but through the material properties of the object itself. In Blender it is possible to add different materials to an object. In fact, the model exported from MakeHuman comes already with a material/skin assigned to it. By creating an extra set of materials, each with a different colour, it was possible to toggle between them and change the skin colour at any time.

3.2 Interactive evolutionary algorithm

One essential step in order to proceed with any genetic algorithm is the construction of a suitable genetic encoding. In this case each gene in the chromosome was set to encode a different property

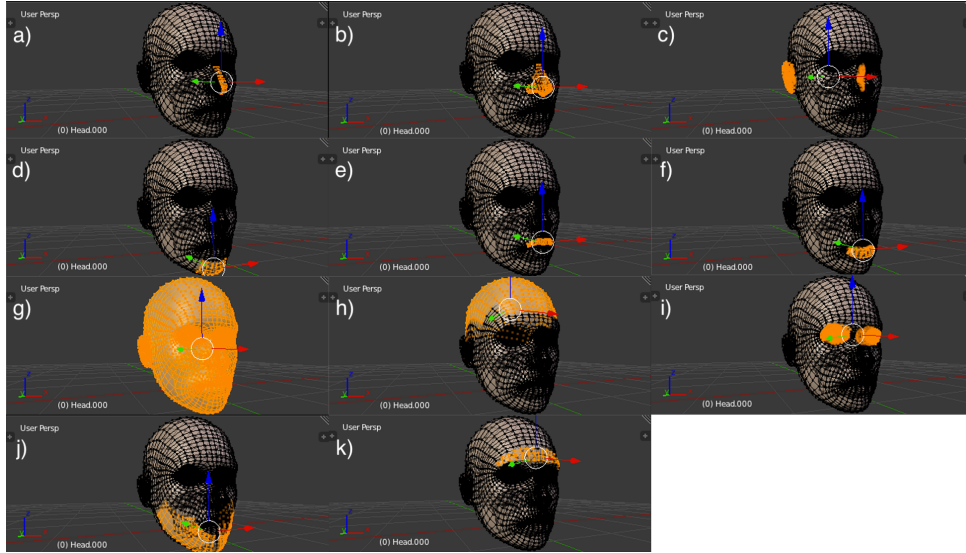


Figure 3.3: Vertex groups used to evolve the face. a) Nose (length); b) Nose (width); c) Ears; d) Chin; e) Upper lip; f) Lower Lip; g) Head; h) Head's upper part; i) Eyes; j) Jaw ; k) Brow ridge.

of the face, i.e. nose length, lip thickness, etc. Thus, depending on the value of a gene, its corresponding face component acquired different characteristics.

However, some constraints were set for these values. Each gene could be at minimum 0 and at maximum 1 and only numbers having one decimal place were allowed. Having this into consideration, it was then possible to encode the initial face through a chromosome. In a genetic algorithm it is common to set the initial population completely at random. Nevertheless, in this case, it was important to encode the neutral/initial face beforehand with suitable values since accumulative changes could cause the skin to rupture. Therefore, the initial gene values were set in such a way as to avoid any of the components to rupture the skin due to its large dimensions.

One could, for example, set all the initial values to 0 but this would allow changes in one direction only, i.e. either increase the component's initial size or decrease it. However, it was desired that some components could change in both directions. After trying different values for which no rupture would be seen and carefully evaluating the limitations of the mesh, the initial gene values were set to the ones presented in figure 3.4.

The genetic representation of the initial face together with the different vertex groups created allowed each component to be changed separately and evolve the face as a whole. The interactive evolutionary computation used is summarized in the algorithm 1 but a more detailed description of each step is given below.

The initial generation was created by randomly generating 8 chromosomes containing gene values within the predefined bounds. From those 8 chromosomes, 8 new faces were displayed on the screen. The neutral face was also included, working as the 9th face of the initial generation.

0.0	0.5	0.3	0.4	0.5	0.5	0.0	0.5	0.5	0.0	0.5	0.5	0.0	0.9
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

Figure 3.4: Initial gene values of the chromosome encoding the base face. These values encode the initial nose length, nose width, ear size, ear rotation, lower lip thickness, upper lip thickness, skin colour, head length, head width, chin size, size of the upper part of the head, jaw volume, brow ridge size and eye size.

From those 9 faces, the user would select the desired one to be the parent of the next generation. The parent chromosome would then be used as the template of 8 new chromosomes that would subsequently mutate. 8 other faces would be displayed around the parent picture and again, the user would select a face. This entire process would be repeated until a final solution was accomplished. Moreover, the mutation probability used was $p_{mut} = 0.2$, meaning that in the current version of the algorithm there were on average approximately three mutated genes in each chromosome per generation. Plus, the algorithm was constructed such that big jumps in gene values would happen with a low frequency. Therefore, the step size of a mutating gene was drawn from a normal distribution with mean $\mu = 0$ and standard deviation $\sigma = 0.2$. Due to this choice, each gene changed its values gradually (steps of 0.1 and 0.2) most of the times, while sometimes jumping to values either far smaller or far larger than the current one.

Algorithm 1: Pseudocode of the interactive genetic algorithm used for the evolution of human faces.

1. Initialize the population by generating at random 8 chromosomes and displaying their corresponding images together with the base face on the screen.
 2. Select the desired image after careful evaluation.
 3. Create the next generation:
 - 3.1. Copy the parent image and place it unchanged on the screen, preferably in the centre.
 - 3.2. Make 8 copies of the parent chromosome and randomly mutate each one of them.
 - 3.3 Display the 8 new faces, preferably around the parent face.
 4. Return to 2 until any of the faces corresponds to a satisfactory solution.
-

Chapter 4

Results

Some snapshots of the final interactive program are shown in figure 4.1. This was the result of applying the interactive evolutionary algorithm previously described in the evolution of the face model presented in figure 3.2. An example of a face evolution over 25 generations is shown in figure 4.2. The starting face (top left) is a face with a big pointy nose, thick lips (specially the lower one), square head, its skin has a light brown colour and its ears are rotated outwards. The final face (bottom right), however, is bigger and rounder, the ears are larger and no longer rotated outwards, the skin is darker, and the lips are less thick. All generations in between show the steps required to obtain those differences. One can see that the overall face transformation is a gradual process, taking several generations to attain a significant difference in appearance. Observe, for example, how the head initially gets bigger and more elongated and as soon as it becomes broader it starts having a rounder shape. Also, note how the evolution does not always flow in the same direction. The colour of the skin, for example, becomes darker throughout the first 4 generations but at generation 9 it returns to a lighter colour. It only darkens again at generation 14 and from then onwards, it gets progressively darker. Another example of bidirectionality is the size of the eyes. Note that at generation 14 the eyes become wider but after 6 generations they return to their initial shape.

Finally, figure 4.3 shows 10 examples of faces obtained through evolution. It can be observed that all faces are distinct from each other while some share commonalities. In general, it is easily seen that the overall shape of the face can be quite distinct. There are elongated faces (a, d and g), round faces (b and c), square faces (e) and sharper faces with visible chick bones and salient chins (i and j). Some faces have a prominent brow ridge, such as face e and f and others have thinner lips, such as faces d and j. Others have wider noses and thicker lips (i). Furthermore, a wide variety of skin colours was achieved, such as a pale tone (a and i), a black one (b), a greyish colour (f and j), a pinker one (d), a tanned skin (e) and different shades of brown (c, g and h).

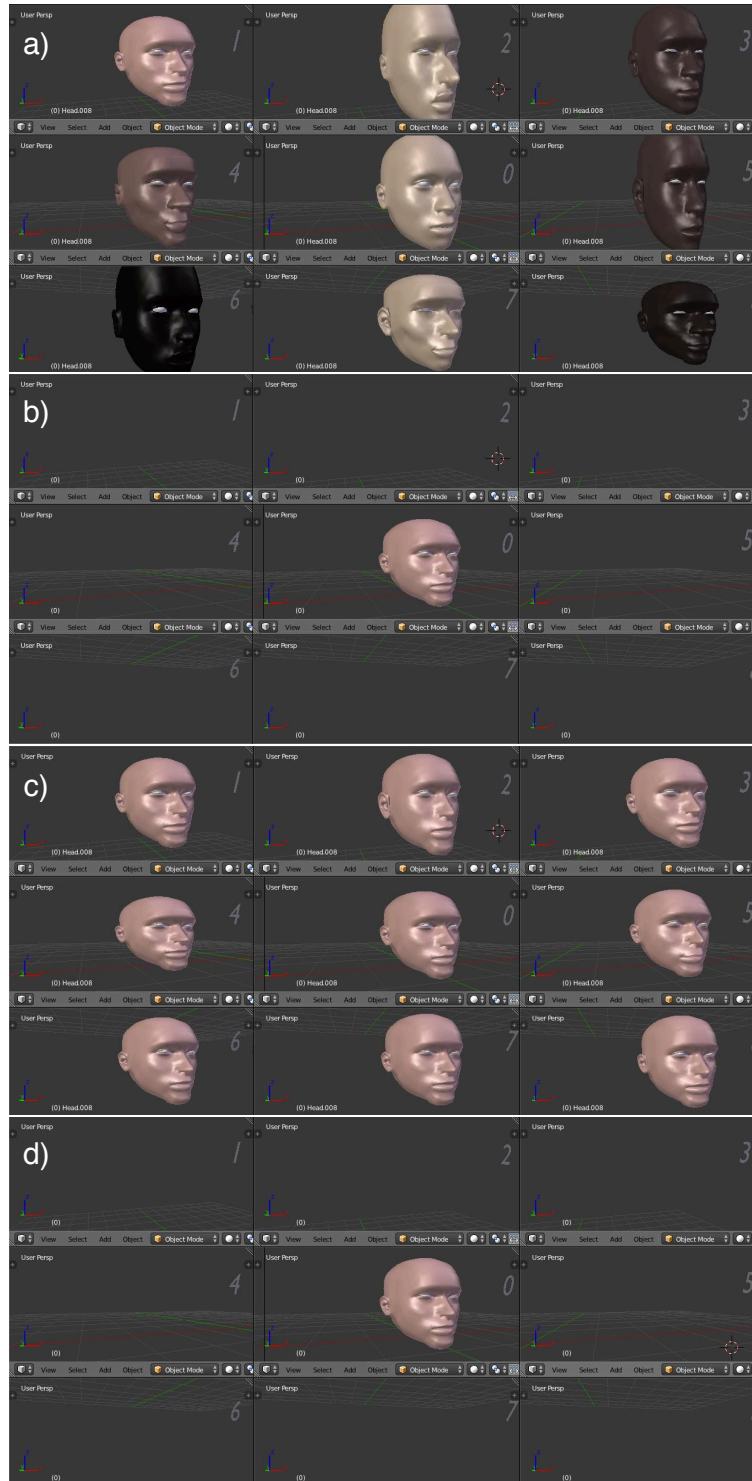


Figure 4.1: Example of the program flow for generation 0 and 1. a) Eight random faces are generated and displayed around the base face. b) The user selects the desired one (face 1 in this case). c) Eight new faces resulting from small mutations of the parent face are displayed. d) The selected face (face 7) is placed in the center and it will serve as a parent for the next generation.



Figure 4.2: Example of one evolution of 25 generations. Starting on the top left, the initial face has a big nose, thick lips and square head. The evolution proceeds towards the right side and it can be observed that the skin colour gets darker and the head starts to be more elongated. The final face is presented on the bottom right and it is a round face with darker colour, thinner lips and smaller nose as compared to the initial one.

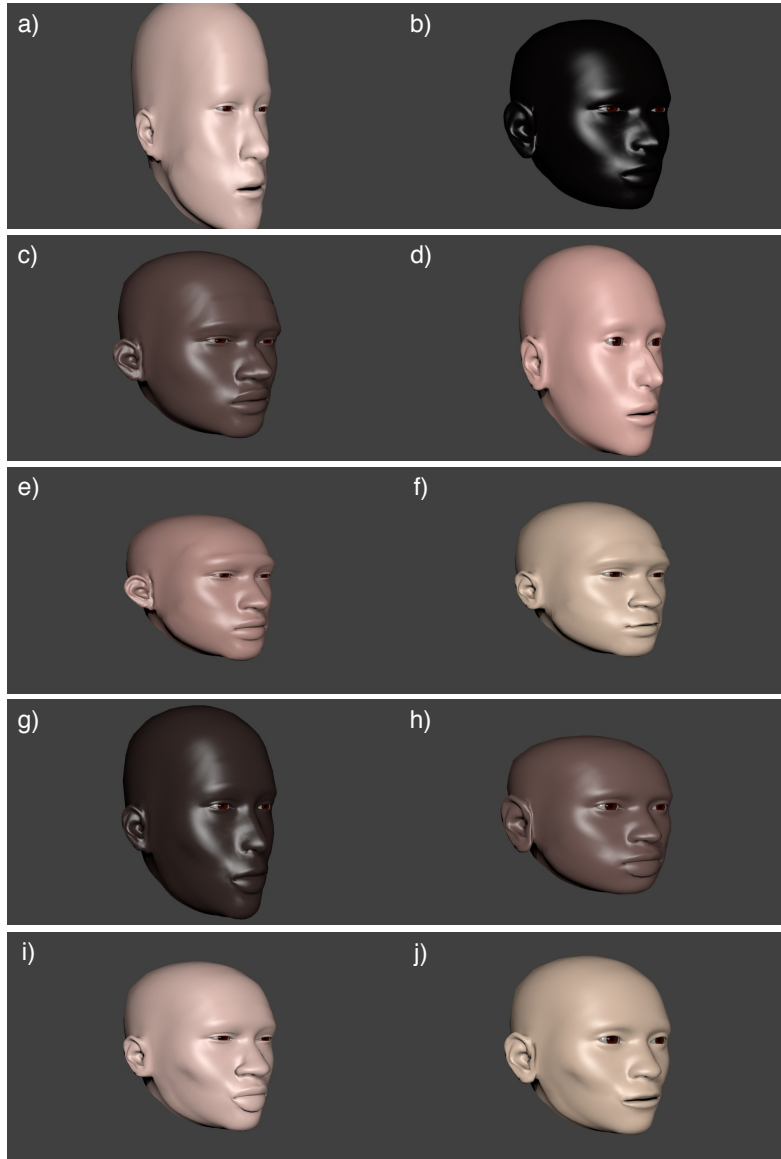


Figure 4.3: Ten examples of faces obtained through the evolutionary algorithm. It can be seen that a wide variety of different faces can be obtained. a) An elongated face with pale skin colour and thin lips; b) Round face with large ears and black colour; c) Round face with brown color, thick lips and small eyes; d) Elongated face with pointy nose and a pink colour; e) Small face with prominent brow ridge, small eyes and thick lower lip; f) Face with thin lower lip, prominent brow ridge and greyish skin colour; g) Elongated face with dark brown colour, small nose and thick lower lip; h) Small head with big ears, voluminous jaw and thick lower lip; i) Face with visible cheekbones, wide nose and small eyes; j) Face with thin lips, visible cheekbones and a greyish skin colour.

Chapter 5

Discussion

As seen in the previous chapter, the current interactive evolutionary algorithm can be used to control the evolution of a human face and by doing so, obtain a range of faces having different facial characteristics. Therefore, the initial goal of the project has been achieved. However, there are some limitations with the current model that are worth mentioning and that could be explored on a further project.

The first one is the fact that the model so far does not allow for a change in eye colour. Eye colour is one of the most important features in a face and having a model that incorporates different eye colours would certainly be a big improvement. Second, the current face is completely hairless. The model used in this project does not have any hair, eyebrows or beard. Together with a diversity in eye colour, using a model able to switch between different hair styles would bring a bigger diversity to the number of obtainable faces.

Another problem with the current model arises from the fact that the different face components were selected manually. It was hence difficult to decide which vertices really corresponded to a certain component. This choice is of great importance as sometimes the integrity of the face could be damaged even by a subtle change in the choice of mesh points. It is possible that in some cases, another group of vertices could represent the wanted component better than the currently used. Moreover, it was difficult to assure that for each selected vertex on the right side of the face, its counter vertex on the left side would be also selected. This became even harder in parts where there was a high density of mesh points, such as the eyes. Perhaps, a more accurate method of assembling the vertex groups could be implemented.

In this work, an already made model of the face was used. It would be a great improvement if the face model could be created from scratch. This would allow for a better control and total understanding of the face parameterization and would also solve the encountered drawbacks.

Concerning the interactive evolutionary algorithm, the main limitation is user fatigue. In each iteration there were 9 images displayed on the screen for user evaluation. After 10 iterations the user had seen already 90 faces, possibly undermining the user's attention span. Furthermore, in each generation it is common to have multiple changes per face and some of those changes can be small enough to pass by unseen. One way to solve this issue would be to display, next to each face, the location and magnitude of each mutation. This would make facial changes easier to track and possibly reduce the user's effort throughout the entire computation. Another possible way to solve this problem would be to automate the selection step. Perhaps either through the assembly of a fitness function or by using a neural network.

Moreover, the evolutionary operators used were selection and mutation. Reproduction was done simply through the cloning of the parent's chromosome. Because of that, there was no mixing of genetic material between the different chromosomes available in each generation. In order to incorporate cross-over in the current model, it is first necessary to be able to attribute different fitness values to the displayed faces. Even if no fitness function can be found, it is still possible to create a point system where the user can score each face separately. For example, in each generation, the user can grade each face with a number between 0 and 5, being 0 the worst and 5 the best point given to a face. After each face is scored with a fitness value, they can pass through a round of selection which precedes the cross-over between their chromosomes.

Finally, it would be interesting to see if a possible match between the face model and a real face could be achieved through an evolutionary approach. Targeting a certain face would be the only objective way to access the quality of the entire model. In fact, such applications have been found in police investigations. EvoFit is a program created to construct pictures of criminals from the memory of a victim or witness using an evolutionary approach [5]. Moreover, besides targeting certain faces, this type of algorithm could be also used to evolve faces from imagination. Totally new and unobserved faces could be modeled. Perhaps even unrealistic or extraordinary ones, which could be useful for the creation of new characters in video-games and films. Furthermore, one does not need to restrict the evolution to faces only. It is definitely possible to expand the current evolutionary algorithm to evolve entire bodies.

Chapter 6

Conclusion

In 1986, Richard Dawkins published a work that came to be the foundation of what is now called evolutionary art. "The Blind Watchmaker" was a book written to explain how natural evolution is the driving force behind the emergence of complex life. To do so, in one of the chapters, the author introduces the Biomorphs. These figures, resembling animals and insects, were a product of a computer program designed to mimic the processes behind natural evolution. Selection and random mutations were in fact the main mechanism turning simple shaped images into more complex structures like those of life forms.

Since then, other genetic algorithms have been developed to create images, music and other types of art. Scientists and artists like William Latham, Karl Sims, Penousal Machado and Jon McCormack are clear examples of people that have been merging the technical and objective world of science with the more subjective realm of art.

The work presented in this report aimed to combine those two domains by constructing an interactive evolutionary algorithm to evolve a model of a human face. As in the program designed by Richard Dawkins, this interactive algorithm was written such that a user would be the main conductor of the face's evolution. Furthermore, the changes in facial features would be the product of random mutations. Hence, the model used in this work required the assembly of a genetic encoding of different face components into a chromosome. Moreover, the overall flow of the algorithm followed the one created by Dawkins. An initial display of different images would be showed in the screen after a random initialization of chromosomes and the user would then select the preferred image. The selected image would be successively mutated in order to display another group of images, which again, the user would use to select the desired one. After successive generations of user selection and random mutations the program would be terminated as soon as a final solution would be seen. In this specific case, evolution allowed the emergence of faces having distinct facial features from each other. Even though there were some limitations

observed, this work serves as a proof of concept that could be further explored in more advanced projects.

Bibliography

- [1] *Artworks*. URL: <https://jonmccormack.info/artworks/>.
 - [2] *Blender*. URL: <https://www.blender.org>.
 - [3] Dawkins, Richard. *The Blind Watchmaker*. Penguin, 1991.
 - [4] Ersotelos, Nikolaos and Dong, Feng. “Building highly realistic facial modeling and animation: a survey”. In: *The Visual Computer* 24.1 (2008), pp. 13–30.
 - [5] Frowd, Charlie D and Hancock, Peter JB. “Evolving human faces”. In: *The Art of Artificial Evolution*. Springer, 2008, pp. 189–210.
 - [6] Gruau, Frederic and Quatramaran, Kameel. “Cellular encoding for interactive evolutionary robotics”. In: *Fourth European conference on artificial life*. Vol. 4. MIT Press. 1997, p. 368.
 - [7] Hemberg, Martin, O’Reilly, Una-May, Menges, Achim, Jonas, Katrin, Costa Gonçalves, Michel da, and Fuchs, Steven R. “Genr8: Architects’ experience with an emergent design tool”. In: *The Art of Artificial Evolution*. Springer, 2008, pp. 167–188.
 - [8] Lambert, Nicholas, Latham, William, and Leymarie, Frederic Fol. “The emergence and growth of evolutionary art-1980–1993”. In: *Leonardo* 46.4 (2013), pp. 367–375.
 - [9] Lewis, Matthew. “Evolutionary visual art and design”. In: *The art of artificial evolution*. Springer, 2008, pp. 3–37.
 - [10] Lourenço, Nuno, Assunção, Filipe, Maças, Catarina, and Machado, Penousal. “EvoFashion: customising fashion through evolution”. In: *International Conference on Evolutionary and Biologically Inspired Music and Art*. Springer. 2017, pp. 176–189.
 - [11] *MakeHuman*. URL: <http://www.makehumancommunity.org>.
-

- [12] McDermott, James, Griffith, Niall JL, and O'Neill, Michael. "Evolutionary computation applied to sound synthesis". In: *The Art of Artificial Evolution*. Springer, 2008, pp. 81–101.
 - [13] *MHX2 documentation*. URL: <https://thomasmakehuman.wordpress.com/mhx2-documentation/>.
 - [14] Parke, Frederick I. "Computer generated animation of faces". In: *Proceedings of the ACM annual conference-Volume 1*. ACM. 1972, pp. 451–457.
 - [15] *Penousal Machado*. URL: <https://cdv.dei.uc.pt/authors/penousal-machado/>.
 - [16] Sims, Karl. *Artificial evolution for computer graphics*. Vol. 25. 4. ACM, 1991.
 - [17] *The Evolutionary Art of Steven Rooke*. URL: <http://srooke.com/>.
 - [18] Wahde, Mattias. *Biologically inspired optimization methods: an introduction*. WIT press, 2008.
 - [19] *William Latham History*. URL: <http://www.latham-mutator.com>.
-

List of Figures

1.1	Image created by William Latham using the Mutator program [19].	2
2.1	Example of Evolutionary Art created by Karl Sims [16].	5
3.1	MakeHuman's starting model. This was the base model for the face representation used in this project.	7
3.2	The base face after rendering the image.	8
3.3	Vertex groups used to evolve the face. a) Nose (length); b) Nose (width); c) Ears; d) Chin; e) Upper lip; f) Lower Lip; g) Head; h) Head's upper part; i) Eyes; j) Jaw ; k) Brow ridge.	9
3.4	Initial gene values of the chromosome encoding the base face. These values encode the initial nose length, nose width, ear size, ear rotation, lower lip thickness, upper lip thickness, skin colour, head length, head width, chin size, size of the upper part of the head, jaw volume, brow ridge size and eye size.	10
4.1	Example of the program flow for generation 0 and 1. a) Eight random faces are generated and displayed around the base face. b) The user selects the desired one (face 1 in this case). c) Eight new faces resulting from small mutations of the parent face are displayed. d) The selected face (face 7) is placed in the center and it will serve as a parent for the next generation.	12

-
- 4.2 Example of one evolution of 25 generations. Starting on the top left, the initial face has a big nose, thick lips and square head. The evolution proceeds towards the right side and it can be observed that the skin colour gets darker and the head starts to be more elongated. The final face is presented on the bottom right and it is a round face with darker colour, thinner lips and smaller nose as compared to the initial one. 13
- 4.3 Ten examples of faces obtained through the evolutionary algorithm. It can be seen that a wide variety of different faces can be obtained. a) An elongated face with pale skin colour and thin lips; b) Round face with large ears and black colour; c) Round face with brown color, thick lips and small eyes; d) Elongated face with pointy nose and a pink colour; e) Small face with prominent brow ridge, small eyes and thick lower lip; f) Face with thin lower lip, prominent brow ridge and greyish skin colour; g) Elongated face with dark brown colour, small nose and thick lower lip; h) Small head with big ears, voluminous jaw and thick lower lip; i) Face with visible cheekbones, wide nose and small eyes; j) Face with thin lips, visible cheekbones and a greyish skin colour. 14
-