Can't See the Forest: Using an Evolutionary Algorithm to Produce an Animated Artwork

Karen Trist, Vic Ciesielski, and Perry Barile

RMIT University
GPO Box 2476V, Melbourne, Vic, Australia
{karen.trist,vic.ciesielski,perry.barile}@rmit.edu.au

Abstract. We describe an artist's journey of working with an evolutionary algorithm to create an artwork suitable for exhibition in a gallery. Software based on the evolutionary algorithm produces animations which engage the viewer with a target image slowly emerging from a random collection of greyscale lines. The artwork consists of a grid of movies of eucalyptus tree targets. Each movie resolves with different aesthetic qualities, tempo and energy. The artist exercises creative control by choice of target and values for evolutionary and drawing parameters.

Keywords: Evolutionary programming, evolved art, new media art, software art, animation, algorithmic art, AI, genetic programming, genetic art, non photorealistic rendering.

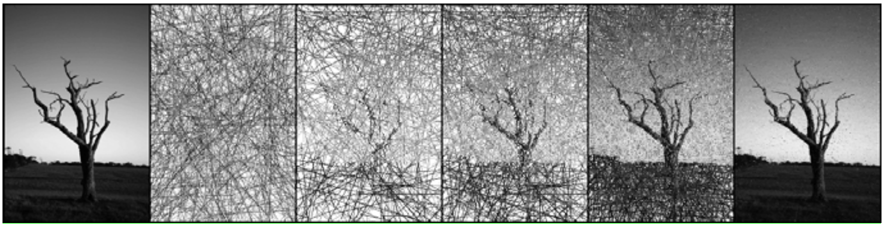
1 Introduction

This paper describes the construction of an artwork using a software system for evolutionary art [1] based on genetic programming [5]. The artwork, 'Can't see the forest', consists of a grid of animated drawings of trees created by using a software program we call 'The Shroud' (after the Shroud of Turin), for its capacity to produce a sort of ghost-like rendering or trace of a target image. We explore this evolutionary art process by describing the created work and the artistic and technical considerations that we encountered in its production. The goal of this collaboration between artist and computer scientist was ultimately to produce an engaging animation suitable for exhibiting in an art gallery.

The Shroud software produces visually engaging animated movies drawing grey level straight-line strokes at various angles and of finite lengths that give a variety of likenesses to a photographic or drawn target image. Figure 1 shows a sequence from a run showing the target with a series of frames of the subject gradually evolving[1]. The movies are of two main kinds: (1) start with a canvas on which a large number of lines are drawn and keep redrawing the canvas until a recognisable subject emerges (see Figure 1), and (2) start with a blank canvas and keep adding lines until a recognisable subject emerges (see Figure 2). In both cases the animations engage the

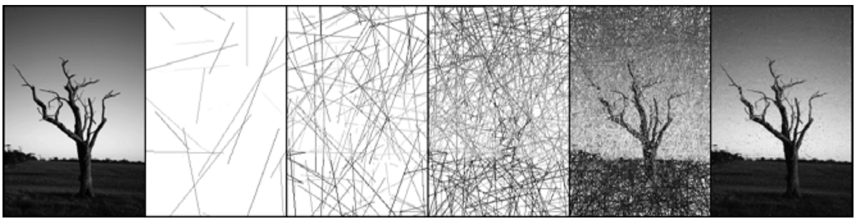
[1] Animations of some of the panels are at http://evol-art.cs.rmit.edu.au/npr_theshroud/

F. Huang and R.-C. Wang (Eds.): ArtsIT 2009, LNICST 30, pp. 255–262, 2010.



Target Generation 1 Generation 24 Generation 350 Gen 1,500 Gen 50,000

Fig. 1. Sequence from a run of the evolutionary algorithm



Target Generation 1 Generation 59 Gen 530 Gen 1,366 Gen 99,900

Fig. 2. Sequence from a run of the evolutionary algorithm starting from a blank canvas

viewer's attention as the viewer waits to see what will emerge. The challenge for the artist is to use the programs to produce animations that are not too fast or too slow, that have interesting targets, that keep the viewer in suspense for long enough, but not too long, and to assess to what level of clarity the target will be revealed.

Having artists working with computer scientists produces some interesting cultural and technical issues. For example, in genetic programming from a computer science perspective, the desire would generally to be to find an optimal result quickly and utilising the fewest resources. However, because we are here concerned with generating perhaps unusual results based on subjective, aesthetic criteria, the focus is more on ways for the programmer to give the artist increased creative control, rather than just finding the best level of fitness to a target image as quickly as possible.

The finished artwork consists of a seemingly infinite loop of animated 'drawings', with no clear start, middle or end. Each run converges more or less towards the target image with its own starting and ending points on the main loop. The duration of the individual loops also vary (being factors of the main loop). The overall effect is to produce a dynamic piece with a balance of resolved and semi-resolved renderings of the target images over time. See Figure 3 for an example of a random sequence.

2 Related Work

A wide variety of evolutionary approaches have been used to generate art works, for example genetic algorithms, genetic programming and ant colony optimisation. A number of these art works have been on a botanical/flora/plant growth theme. In Panspermia

[9] a simulated botanical environment was built using a genetic model of growth. Sims later expanded this work with Genetic Images [8] in which viewers were able to interactively re-sequence genetic markers used in the production of evolving images. In [6], a survey is presented of various evolutionary approaches to adapting L-Systems – simple languages that allow rules to be generated that model plant growth.

Many contemporary artists are producing work (installations, painting, photography etc) that responds to issues of sustainability in the environment. Patricia Piccinini's work 'Signs of Life' from the 'Plasticology' series consists of '50 monitors with computer generated plants constantly swaying on the wind' [7], inspired by the artist's ongoing exploration of 'bio-scientific practices of manipulation and alteration of living beings, of creating new worlds.' [2]. Lyndal Jones's 'Avoca Project' responds to the impacts of climate change in a novel way through intervention in the environment and a series of site-inspired artworks [4]. Our artwork was also inspired by the pressing problems evident today such as global warming, deforestation and habitat destruction.

Repetition is an important creative practice for artists, whether used in the process of working up ideas, (a painter might do many small studies of a subject before the final, full-size painting), or as an essential feature of the finished work. Impressionist painter Claude Monet produced many studies of repetition of a theme over his long career: of haystacks, cathedrals and of the water lilies in his garden at Giverny, 'in order to display minute discriminations of perception, the shift of light and colour from hour to hour on a haystack, and how these could be recorded by the subtlety of eye and hand' [3,p359].

Pop artist Andy Warhol also used repetition extensively in his many series of popular culture and iconic figures. However, his silk screen print grids are not about recording changes in light but his repetitions instead 'mimic the condition of mass advertising' [3,p359] speaking 'eloquently about the condition of image overload in a media-saturated culture'[3,p540]. Warhol produced variations on a single image, repeating 'these images until repetition is magnified into a theme of variance and invariance, and of the success and failures of identicalness' [10].

3 Artistic Inspiration

The artwork interrogates concepts of environmental sustainability through creating a digital dying 'forest' (through repetition in a grid format, which suggests a continuation beyond the frame) of evolving and devolving animated movies of dead eucalypt trees. The idea of using genetic programming to create this work seemed an appropriate way to explore the effects of deforestation and global warming on our fragile and continually evolving environment in Australia. It also seemed apt that the programs were represented as tree structures, while the targets are also literally trees. The eucalypts photographed are naturally hardy and long-lived, however many have succumbed to drought in the last few years and can be seen in paddocks on farms and in forests, weathered silver and ghost-like.

The artist lived in Tasmania during the heyday in the 1970s of clear felling for wood-chipping in the old growth forests. The artist had been dismayed to see windrows of the leftover trees bulldozed into heaps for eventual burning. In the Shroud program, when the tree size is small there are very few lines drawn and the aesthetic effect is very like a game played by the artist when she was a child called ‘Pick Up Sticks’. It was played by letting the sticks fall like spaghetti into a saucepan - holding them upright and then letting them fall into a heap. The trick was to pick up as many sticks as you could without making the others move. The simple, straight lines drawn in the Shroud software reminded the author of this childhood pastime, only then for her consciousness to shift back to the tragedy of the piles of logs in the Tasmanian forests. In ‘Pick Up Sticks’ the black sticks earned more points. In the wind-rows of bulldozed trees they would all be burnt to black.

Artistic possibilities were suggested during the process of software testing. Normally an artist drawing would use a rough sketch as a beginning, which would then usually be worked up into a more finished piece. However, as a result of doing a large series of runs and then playing them alongside each other on the computer screen as they evolved differently and at different rates, the artist was inspired by the creative and narrative possibilities inherent in combining both the ostensible ‘failures’ and the ‘successes’. Thus the software experimentation drove the idea of using a grid of multiple animations for the final artwork with some not fully resolving.

The artist was not interested in simply reproducing the target images in another form – animated drawings. She wanted to extend the creative possibilities by utilising the well-established art practice of repetition. The software was ideal for creating a digital dying ‘forest’ as it allowed for almost limitless versions of the target. However, the concept of regeneration is also present in the process, as the animations on a loop continually evolve towards the tree targets and then dissolve into random chaos, only to be reborn again.

4 Shroud Program

4.1 The Evolutionary Algorithm

Evolutionary algorithms are characterized by a population of potential solutions that improve over the course of the evolutionary process. In our case an individual is a program that draws an image on a canvas and the measure of improvement, or fitness, is resemblance to a target image. See [1] for a complete technical description.

In the first generation the population of programs is created randomly. Each program is able to generate a sequence of greyscale brush strokes on the canvas. The program is represented as a tree structure (not to be confused with the tree images that are the targets of the animations). The parameters of brush strokes such as line length, intensity and angle are determined at this time. The basic evolutionary process proceeds in the following manner:

a) Wipe the canvas clean and draw the output of one program on to the canvas.
b) Compare this rendered canvas to the target image. A rendering with a greater similarity to the target image is assigned a high measure of ‘fitness’, while a rendering with lesser similarity to the target image is assigned a lower measure of ‘fitness’.
c) Do this rendering, perform comparison to the target image and assign fitness to each program, wiping the canvas clean prior to each rendering.

At the end of this process the programs are listed in order of fitness. The programs now enter a phase called the mating phase. Programs are combined with one another in a process similar to crossover in biological reproduction. Parts of two programs are combined and two ‘offspring’ are produced and then placed into the population of the next generation. The selection of mating partners is based on the fitness of individual programs. Offspring are also generated by a process of mutation in which a random change is made to a single parent. Programs with higher fitness have a greater probability of being selected for mating. This is a process similar to Darwinian evolution through natural selection. Parts of each parent go into the offspring. The theory is that the building blocks (the parts of programs that render brush strokes) of programs with higher fitness will be propagated into a new generation, while building blocks that do not promote high fitness will disappear from the ‘gene pool’. The evolutionary process is terminated when a pre-specified number of generations has been completed or a pre-specified level of similarity to the target image is reached. Whenever a program is found that has a higher fitness than the best program in previous generations, a frame of the final animation is generated. There are a number of parameters which control the evolutionary and drawing processes. These are shown in Table 1.

Table 1. Parameters of the evolutionary and drawing processes

Program Tree Size	The maximum number of strokes that can be drawn. Larger programs draw more strokes.
Number of Generations	The number of generations to execute before stopping.
Fitness Target	Stop when the fitness reaches this value.
Write Nth Best	Rather than rendering every best individual discovered as a frame, only write every Nth.
Fitness Write Threshold	Only write a frame if the fitness has improved this amount since the last frame written.
Accelerated Convergence	In the basic program a pixel on the canvas is always changed based on values in a draw function. If this parameter is set a pixel is only changed if the new value would be closer to the target value.
Incremental Rendering	Rather than starting with a blank canvas the rendering starts with the best canvas from the previous generation.

5 Producing the Work with ‘Shroud’

The artist exercises creative control by choice and preprocessing of targets and by choosing values for the evolutionary and drawing parameters described in Table 1. Some combinations of parameters lead to unusual and unexpected effects.

5.1 Selection and Preparation of Suitable Target Images

The trees were photographed so as to include minimum background distraction so the tree stands out clearly against a light-coloured sky. The artist tried, where possible, to keep the scale and the horizon line reasonably constant for continuity between individual animations within the final piece. The target images chosen needed enough contrast and strong outlines so that even in the small tree size animations we would be able to make out the beginnings of a 'drawing' of a tree, even though very few lines are drawn.

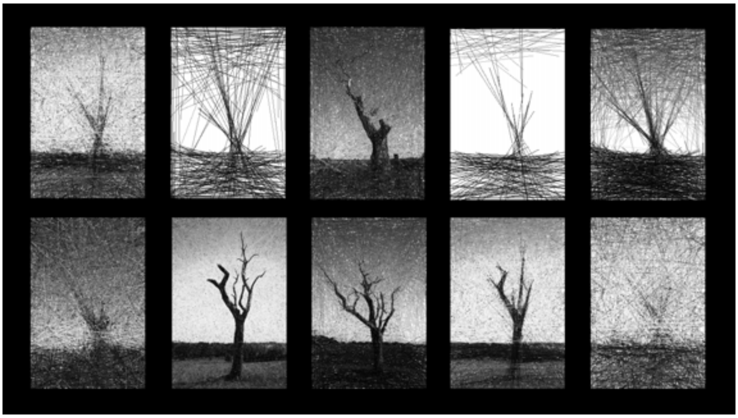


Fig. 3. Random sequence from artwork

5.2 Choices for the Artist

a) How close to the target should the final image be? In some situations photorealism is desired. In other situations a fuzzy, suggestive abstraction is desired. The level of photorealism is controlled by a number of parameters acting in concert. Large values for *Max Generations* and *Program Tree Size* and turning on *Incremental Rendering* will give more photorealism. Beyond a certain number of generations there is often minimal improvement. However, too few generations will often not produce anything resembling the target, particularly with smaller tree sizes.

b) At what rate should the target emerge? Usually the artist's goal is to have the target emerge at a rate that will keep the viewer's interest. If the target emerges too fast there is no mystery or anticipation. If the target emerges too slowly the viewer loses interest. The rate at which the target emerges is primarily controlled by the parameters *Write Nth Best* and *Fitness Write Threshold.* Smaller values of these parameters will give longer animations or animation segments and more frames to be viewed before the target is revealed. Setting *Accelerated Convergence* and *Incremental Rendering* will generally decrease the time before the target is revealed. Setting a large *Program tree size* and turning on *Accelerated Convergence* can result in the target being revealed in the very first frame.

c) Should the target be built up from a blank canvas adding more and more lines or should it emerge slowly from a large number of randomly drawn lines? A small tree size and *Incremental Rendering* will result in the former effect while a large tree size without *Incremental Rendering* will give the latter effect.

d) How much energy should be in the animation? Some animations exhibit a considerable energy in that lines are continually being drawn and redrawn. In other animations the drawing of the lines is not as obvious and the target slowly emerges without such vigorous line drawing on the canvas. Generally small program sizes and *Incremental Rendering* turned off give the former effect while larger program sizes and *Incremental Rendering* turned on give the latter effect.

The individual images in Figure 3 are snapshots from animations that have been created with a variety of choices for the parameters described in this section.

6 Reflection on Challenges and Opportunities

The artist's main practice is still photography. Working with series of images to form a narrative is common practice in photography. However, the opportunity to work with subtle repetitions of an image and then animating the results into video format for presentation was creatively stimulating.

A challenge for programmer is that the evaluative process is done on subjective and aesthetic (rather than efficiency) criteria. It can be challenging for a programmer to evaluate success without using objective and clear criteria. Ideally, in evolutionary programming, the animation would evolve to closely resemble the target image in as few generations as possible. But for the artist planning a final artwork using many different animations in a grid or on multiple screens, it might be more desirable to be able to utilise the steps along the way in the artwork, as opposed to using only the final or 'best' result. An artist, for whom the process is often the most engaging part of the work, will often react creatively to accidents rather than having a pre-defined set of criteria for successful outcomes. The programmer saw himself in this process as a mediator between the artist and the machine.

Combinations of different choices can give quite a wide range of different effects, some desirable and others not. For example, large tree sizes and incremental rendering can give a too obvious resolving of the target in the first generation. Conversely, turning off *Accelerated Convergence* and *Incremental Rendering* can produce many thousands of generations and still not reveal the target.

7 Conclusions and Future Work

'Can't see the forest' is a work that has been created by exploratory and systematic use of a software system based on evolutionary algorithms. It adds creatively to the conversation around issues of environmental sustainability. The feedback loop established between the artist and the programmer functioned to quickly solve technical problems and to provide artistic inspiration through experimenting with the different iterations of the software as development progressed. This iterative process was very satisfying and fruitful creatively.

A major frustration for the artist with the use of the Shroud software is that some of the runs take hours or even days. In future work we plan to develop algorithms that will run much faster and give the artist much earlier feedback on the consequences of parameter choices. We also plan to give the artist more control over the length and thickness of the lines drawn, to work with coloured lines and images, and to look at the possibility of changing the parameters during the course of a run.

Acknowledgments. We thank our colleagues Dr. M. Berry, D. Keep and Dr. D. D'Souza for suggestions relating to this paper and RMIT's Design Research Institute for seed funding.

References

1. Barile, P., Ciesielski, V., Trist, K., Berry, M.: Animated Drawings Rendered by Genetic Programming. In: GECCO 2009: Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation, pp. 939–946. ACM, Montreal (2009)
2. Haraway, D.: Speculative Fabulations for Technoculture's Generations: Taking Care of Unexpected Country, Essay (2007), `http://www.patriciapiccinini.net` (accessed 24-April-2009)
3. Hughes, R.: American Visions: The Epic History of Art in America. Harvill Press (1997)
4. Jones, L.: The Avoca Project (2009), `http://www.avocaproject.org/` (accessed 24-April-2009)
5. Koza, J.: Genetic Programming: On the Programming of Computers by Means of Natural Selection. MIT Press, Cambridge (1992)
6. McCormack, J.: Evolutionary L-systems. In: Hingston, P.F., Barone, L.C., Michalewicz, Z. (eds.) Design by Evolution: Advances in Evolutionary Design, pp. 168–196. Springer, Berlin (2008)
7. Piccinini, P.: Signs of Life (1997), `http://www.patriciapiccinini.net` (accessed 24-April-2009)
8. Sims, K.: Artificial Evolution for Computer Graphics. In: SIGGRAPH 1991: Proceedings of the 18th Annual Conference on Computer graphics and Interactive Techniques, pp. 319–328. ACM, New York (1991)
9. Sims, K.: Panspermia (1990), `http://www.karlsims.com/panspermia.html` (accessed 27-April-2009)
10. Wilson, W.: Prince of Boredom: The Repetitions and Passivities of Andy Warhol (1968), `http://www.warholstars.org/andywarhol/articles/william/wilson.html` (accessed 24-4-2009)