Automated Daily Production of Evolutionary Audio Visual Art — An Experimental Practice

Tatsuo Unemi
Department of Information Systems Science
Soka University
Hachiōji, Tokyo 192-8577 Japan
unemi@iss.soka.ac.jp

Abstract

Evolutionary computing based on computational aesthetic measure as fitness criteria is one of the possible methods to let the machine make art. The author developed and set up a computer system that produces ten short animations consisting sequences of abstract images and sound effects everyday. The produced pieces are published on the internet using three methods, movie files, HTML5 + WebGL, and a special application software. The latter two methods provides viewers experiences of a high resolution lossless animation. Their digest versions are also uploaded on a popular web service of movie sharing. It started October 2011. It is still in an experimental level that we need to brush up, but it has not always but often succeeded to engage the viewers.

Introduction

As similarly as the evolutionary process in the nature has produced huge number of complex variations of unique species on the earth, evolutionary computing has a capability to produce unpredictable designs by the computer. As the nature often provides us experiences of beautiful audio visual stimuli, the computer has a potential capability to produce beautiful images and sounds if we set up a combinatorial search space in the machine that contains masterpieces. We can find a lot of technical variations for such approach under the name of "Generative art (Galanter 2003; Pearson 2011)." The design of computational aesthetic measures is very important to realize an efficient search in the huge space. It would act as a skill of a genius photographer who can find amazing scenery in the nature to be captured by his/her camera. It is easy for the computer to generate huge number of audio visual patterns by exhaustive search, but almost all of the products would be trash without an appropriate measure.

Though development of the computable models for aesthetic measures comparable with the human artists is on the long way of challenge in the research field of computational creativity, some of the methods has already been examined in the experimental activities by a number of researchers, such as (Machado and Cardoso 2002; Ross, Ralph, and Zong 2006; den Heijer and Eiben 2010). The author has also developed an experimental system of evolutionary computing that automatically produces art pieces, combining ideas of preceding researches and his own ideas (Unemi 2012a). Owing to the recent improvement of computational power of graphical processing unit (GPU) on the personal computer, it became possible to use this type of system for realtime production of non-stop sequence of short animations on site (Unemi 2013). At the same time, it is also possible to set up a machine to make automatic production everyday without any assistance by human.

This paper introduces the author's project named "Daily Evolutionary Animation" that started October, 2011. The following sections describe a summary of evolutionary process, aesthetic measures employed, daily production process, and a public showing on the internet. In the final section, we discuss future extensions along this project.

Summary of evolutionary process

The author developed SBArt (Unemi 2009) originally as a tool to breed a visual evolutionary art using a mechanism of interactive evolutionary computation (Takagi 2001). The first version that runs on UNIX workstation was released in the public domain on the internet in 1993. It is based on a similar mechanism of the pioneering work by (Sims 1991) that uses a tree structure of mathematical expression as the genotype. The expression is a function that maps (x,y,t) coordinate of spaciotemporal space to a color space of hue, saturation and brightness. The spacial coordinate (x,y) is used to indicate the pixel in the image, and the temporal coordinate t indicates the frame position in the movie. Each expression is organized by the terminal symbols and non-terminal symbols. A terminal symbol expresses a value of three dimensional vector. It is a constant containing three scalar values or a permutation of three variables, x, y and t. A non-terminal symbol is a unary or binary operator that takes three dimensional vector for each argument and result value. We prepared nine unary functions including minus sign, absolute value, trigonometric functions, exponential functions, and so on; and ten binary functions including addition, subtraction, multiplication, division, power, and so on. Two selective functions that return one of two arguments choosing by comparison of the first elements are effective to compose a collage of different patterns. Each genotype is used to draw the phenotype by determining the color values distributed in a volume of movie data. The computational cost depends on the resolution of both space and time because it must calculate a three dimensional value for each pixel.

As an extension of the system, an automated process of evolution was implemented as described in (Unemi 2012a). Evolutionary process is conducted in a manner of minimal generation gap method (Satoh, Ono, and Kobayashi 1997) that produces only two offsprings from randomly selected parents in each computing step. The genetic reproduction is done in a style of genetic programming (Koza 1992) using subtree exchange for crossover and symbol replacement for mutation. To prevent infinite extension of the length of genotype through the iteration of genetic operations, the maximum number of symbols in a single genotype is restricted within 120. The fitness values are calculated based on aesthetic measures described in the next section.

It used to take some seconds to render a single frame image for movie production in 2001, but it became possible to render an animation in realtime by using the parallel processing of GPU. We revised the software so that it uses Core Image Framework by compiling the expression into Core Image Kernel Language to take advantage of GPU's power (Unemi 2010). It is a dialect of shading language GLSL in OpenGL working on MacOS X.

Aesthetic measures

It might be an ultimate goal of the research on computational creativity to implement a computable procedure that evaluates how a pattern is beautiful as a delegate of human critics. Many artists and scientists have been struggling with this difficult and interesting theme from several points of views as summarized in (Galanter 2012). It is obvious that the human's decision on aesthetics is depending on his/her own both private and social experiences, but it is also affected by physical functionalities of our sensory organs and fundamental signal processing in the brain widely shared among humans beyond the differences in cultures and races. Some of these measures in a level of perception should match with a mathematical theory of complexity and fluctuation.

We implemented three for each measure on geometric arrangement and on distribution of micro features for a still image, that is,

1. pseudo complexity measure utilizing JPEG compression,
2. global contrast factor in color image,
3. distribution of gradient angles of brightness,
4. frequency distribution of hue values,
5. frequency distribution of brightness and
6. average and variance of saturation values.

The detail of procedure for each measurement and auxiliary normalization are described in (Unemi 2012a). All of these procedures are relatively easy to implement utilizing well known technics of image processing.

The method 1. is a convenient approximation of complexity originally used in (Machado, Romero, and Manaris 2007). The evaluation is done by calculating a ratio between the compression ratio and the ideal value the user specified. 2. is a modified version of the factor proposed by (Matkovic et al. 2005). The original version takes a gray scale image to calculate the differences of brightness between each pair of adjacent pixels in multiple resolution, but we extended it to be applicable for a color image by replacing the difference of brightness with the distance in the color space.

For 4. and 5., there are a number of hypotheses and investigations on a frequency distribution of different types of features observed in phenomena happened in both nature and human society, such as pressure of natural wind, sound frequencies from a stream, populations of cities, note pitches of music, and so on. One of the well-known hypothesis is *power law* on which we can find a number of samples in (Newman 2006), for example. (den Heijer and Eiben 2010) is employing *Benford's law*, a similar shape of distribution with the power law, as one of the factors to measure the aesthetic value. We use a distribution extracted from one thousand snap photos of portraits and natural sceneries as the ideal distribution, that is approximately similar to the power law. 6. is a subject to be adjusted following the user's preference, colorful or monotone. We used a parameter setup for relatively psychedelic results at the start time, but changed it for more grayish results some months later, in order to make the results give the viewer weaker visual stimuli. The geometric mean among these measures is taken as the total evaluation of a single frame image.

To evaluate a movie, the aesthetic measure should be calculated from all of the pixels contained in the three dimensional volume of space and time of colors. However, it is still difficult to complete the calculation within an acceptable time for all of the data in the final product even using parallel processing on GPU. For example, half a minute of hi-definition movie contains approximately 2 giga pixels. To reduce the computational cost, we uses reduced resolution of 512×384 pixels for each frame image, and picks up only ten frames as the samples. In total, the number of pixels to be calculated is $512 \times 384 \times 10 = 1,966,080$. It is also important to combine an aesthetic measure on motion in animation. We employed a simple method of taking average value of absolute differences between colors of two pixels in the same position of consecutive frames in order to estimate how fast or slow the picture is moving. The point of motion measure is the inverse value of absolute difference with the ideal speed specified by human. The final evaluation is a geometric mean between the average point of still images and the average point of motion measures among sampled frames.

Automated daily production

The functionality of automated evolution has enabled not only an installation of automatic art but also automated production without an assistance by human. From October 6th, 2011, the system has been automatically producing ten movies everyday. The production procedure starts in the morning of Japanese Standard Time, continuing the evolutionary process from a random population until the completion of 200 steps of generation alternation. Starting from 20 randomly generated genotypes, children are added to the population until the population size reaches 80, then replacement starts. To prevent a premature convergence that often happens in search process in optimization, the population is refreshed by the following procedure for each 50 steps. It

1. picks up the best 15 individuals from the current population,
2. generates five random genotypes,
3. produces 20 individuals by crossover operation from individuals in 1. + 2., and then
4. starts the same process from these 20 individuals as conducted in the first step.

Throughout the process, $20+2\times 200+20\times(200/50-1)=500$ (20 in the initial population, two children for each step and 20s in the refreshing procedure for each 50 steps) individuals are examined.

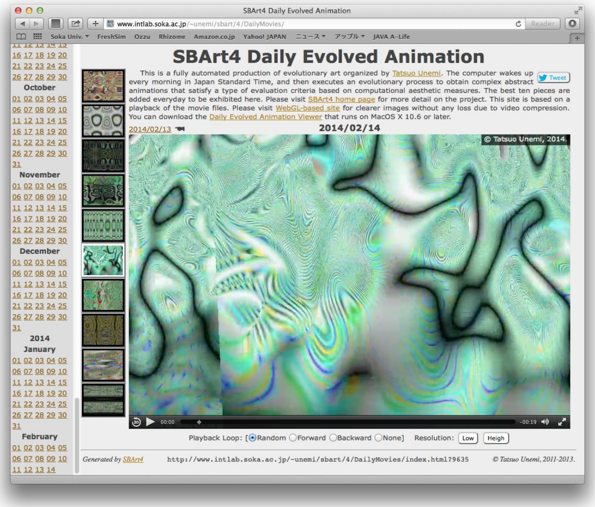
After the completion of 200 steps of evolutional process, the procedure selects the best ten individuals from the final population, and generates a source code of shading language for WebGL and 20-second movie files for each. A synchronized sound effect is also generated without any prerecorded sampled data but purely synthesized sound waves by combination of oscillation and modulation as described in (Unemi 2012b). The parameters of sound wave synthesis are the statistical factors extracted from frame images.

The main machine for the evolutionary production is an old MacPro 2006, equipped with two Intel Xeon dual core processors of 3 GHz, GeForce 7300 GT as GPU and MacOS X 10.6. as the operating system. The elapsed time necessary for the evolutionary process described above is approximately 90 minutes. It would be reduced in less than half if we could arrange it by a newer machine.

The entirety of the daily process is controlled by a program in AppleScript that accesses to application softwares, SBArt4 for evolutionary production, QuickTime Player 7 and X to convert the movie file format and to organize a digest movie, *curl* to submit the digest to YouTube, and *"t"* to announce the completion on twitter. The process is launched as a startup procedure after the machine wakes up at the scheduled time everyday. If no error occurs, the machine shuts down automatically.

Public showing on the internet

To complete the fully automated process by exhibiting the products on the internet, we built three types of user interfaces for viewers based on movie files, HTML5 + WebGL, and a special application software. In all of these methods, the animations are automatically played back in a sequence following the viewer's choice from three alternatives, random, forward and backward. The viewer also allows to directly select the date from the calendar shown in the graphical user interface, and choose one of ten pieces listed as thumbnail images to be played back. Figure 1 shows a sample of the web page to watch the animations distributed as a form of movie files.



SBArt4 Daily Evolved Animation

www.intlab.soka.ac.jp/~unemi/sbart/4/DailyMovies/

SBArt4 Daily Evolved Animation

This is a fully automated production of evolutionary art organized by Tatsuo Unemi. The computer wakes up every morning in Japan Standard Time, and then executes an evolutionary process to obtain complex abstract animations that satisfy a type of evaluation criteria based on computational aesthetic measures. The best ten pieces are added everyday to be exhibited here. Please visit SBArt4 home page for more detail on the project. This site is based on a playback of the movie files. Please visit WebGL-based site for clearer images without any loss due to video compression. You can download the Daily Evolved Animation Viewer that runs on MacOS X 10.6 or later.

2014/02/13 2014/02/14

© Tatsuo Unemi, 2014.

Playback Loop: [● Random ○ Forward ○ Backward ○ None] Resolution: Low Heigh

Generated by SBArt4 http://www.intlab.soka.ac.jp/~unemi/sbart/4/DailyMovies/index.html?9635 © Tatsuo Unemi, 2011-2013.

Figure 1: A sample of web page to watch the animations distributed in a form of movie files.

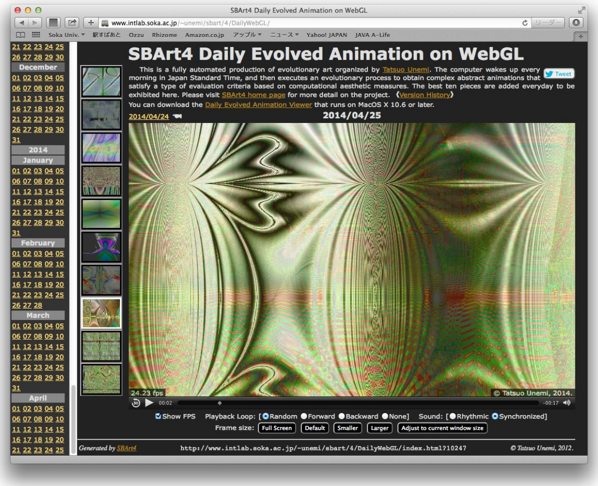
Movie files

Each of the produced movie files is compressed in both the H.264 and Ogg Vorbis formats in order to be adaptable for playback by popular web browsers, such as Safari, FireFox, Google Chrome, and Opera. These movies are accessible from `http://www.intlab.soka.ac.jp/~unemi/sbart/4/DailyMovies/`. Reorganization of a web site to adapt to the newly generated movies is also performed automatically just after the compressed movie files are uploaded to the web server. The daily and weekly digests of these movies are also posted to a popular site for movie sharing. A daily digest is a sequence of six-seconds excerpts for each movie, for a total duration of one minute. A weekly digest is a sequence two-seconds excerpts for each of the 70 movies produced in the last seven days. These digests are accessible at `http://www.youtube.com/user/une0ytb/`.

The daily process consumes an average of 346 MB of the storage in the web server everyday, which means that storing all of the movies produced over a number of years on a hard disk drive is feasible, because 126 GB for one year's worth of movies is not unreasonable considering the HDD capacity of currently available consumer products.

HTML5 + WebGL

A drawback of movie file is dilemma between quality and size. Usual environment of an internet user has no enough capability to display an uncompressed sequence of raw images. If we try to transmit uncompressed movie data of VGA (640×480 pixels) in 30 frames per second, the required band width is $640\times 480\times 30\times 3=27,648,000$ bytes per second. It is possible in a local area network with Giga bit channel, but difficult for usual connection beyond the continents toward a personal computer at home. The compression techniques widely used were designed for movies captured by the camera and/or cartoon animation. Because the evolutionary art might contains very complex patterns that is difficult to be compressed efficiently, such methods commonly used are not always effective for this project.


SBArt4 Daily Evolved Animation on WebGL

www.intlab.soka.ac.jp/~unemi/sbart/4/DailyWebGL/

SBArt4 Daily Evolved Animation on WebGL

This is a fully automated production of evolutionary art organized by Tatsuo Unemi. The computer wakes up every morning in Japan Standard Time, and then executes an evolutionary process to obtain complex abstract animations that satisfy a type of evaluation criteria based on computational aesthetic measures. The best ten pieces are added everyday to be exhibited here. Please visit SBArt4 home page for more detail on the project. 《Version History》

You can download the Daily Evolved Animation Viewer that runs on MacOS X 10.6 or later.

2014/04/24 2014/04/25

24.23 fps © Tatsuo Unemi, 2014.

☑ Show FPS Playback Loop: [● Random ○ Forward ○ Backward ○ None] Sound: [○ Rhythmic ● Synchronized]

Frame size: Full Screen Default Smaller Larger Adjust to current window size

Generated by SBArt4 http://www.intlab.soka.ac.jp/~unemi/sbart/4/DailyWebGL/index.html?10247 © Tatsuo Unemi, 2012.

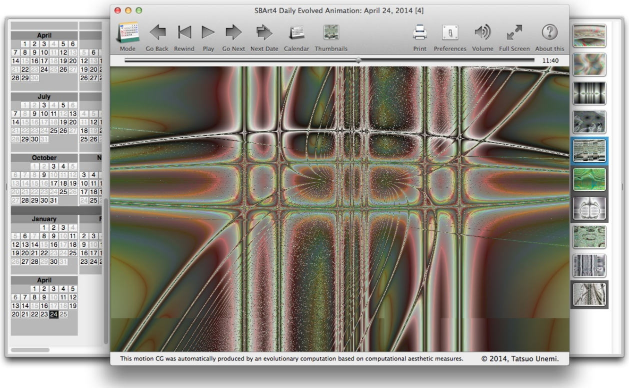
Figure 2: A sample of web page to watch the animations distributed in a form of shading code.

The latest web technology made it possible to let the browser render a complicated graphic image by downloading a script written in JavaScript. The newest specification of HTML5 includes some methods for interactive control of both graphics and network communication. In addition, WebGL is available to render a 3D graphics in a 2D rectangle area of *canvas* object utilizing shading language GLSL ES. It is possible to render an image without any loss by compression if the browser directly draw it based on functional expression produced through evolutionary production. Because SBArt4 is using Core Image Kernel Language to render each image as described above, it is relatively easy to generate a source code of GLSL ES from the genotype. An advantage of shading language is that it is possible to render arbitrary size of image without loss even if it's in the full screen mode of high DPI display. The fastest frame rate is depending on both the power of hardware and the efficiency of JavaScript execution on the browser.

An audio file in high quality is not so heavy in comparison with movie file. JavaScript controls the frame image alternation by checking the progress of audio playback. The average size of audio file in AAC compression is approximately 330 kbytes in 44.1 kHz as sampling rate, 16 bits as sample size, two channels and 20 seconds in duration, for each piece. Because the average size of shading code is 3 kbytes for each piece, the total amount of storage required for the web server is almost one 100th of the case in movie file. The service is available from `http://www.intlab.soka.ac.jp/~unemi/sbart/4/DailyWebGL/`. Figure 2 shows a sample web page to watch the animations distributed in a form of shading code.

Specific application software

In the method using WebGL described in the above section, it sometimes suffers computational bottleneck due to the hardware performance and browser's implementation for executable scripts. To take full advantage of the power of machine at viewer's side, it is the best way to distribute an application software optimized for viewing the products. We developed a software named *DEAViewer* runnable on OS X 10.6 or later, and are distributing it on Apple's App Store in free of charge. The basic mechanism is almost same with the case of WebGL, but the procedure of control part is directly executed on CPU by compiled machine code without any overhead of either compilation or interpretation of the code. It downloads the same information used in WebGL version, and slightly modifies the shading code to adapt to an efficient GLSL code. The more detail information is at `http://www.intlab.soka.ac.jp/~unemi/sbart/4/deaviewer.html`. It provides a viewer's experience of 30 fps lossless animation on 4K display. Figure 3 shows a sample image of a window of special application, DEAViewer, to watch the animations distributed in a form of shading code.


SBArt4 Daily Evolved Animation: April 24, 2014 [4]

This motion CG was automatically produced by an evolutionary computation based on computational aesthetic measures. © 2014, Tatsuo Unemi.

Figure 3: A sample image of a window of special application, DEAViewer, to watch the animations distributed in a form of shading code.

Future extension

Though it has already passed for two and half years and the number of produced pieces reached 9,500, but we have not conducted any analysis over them so far. In the author's intuitive reflection through those years, it often produces amazing pieces but sometimes not. Almost all of productions, except small number of erroneous failure, obtained higher fitness defined as a type of aesthetic measure we designed. This is a typical evidence why we need more research to pursue a human equivalent ability of evaluation even in a perception level for visual arts, because it suggests that the measures employed here might be necessary but not sufficient.

Of course, there are several candidates of aesthetic measures to be introduced, such as a composition based on golden ratio and/or rule of thirds. If we want to obtain an image that inspires something we know in the physical real world or in popular mythology, the composition is very important though it might be a long way to achieve. It is also necessary to consider not only on the perception level but also deeper level of understanding by combination of memory retrieval and conceptual inference connected to emotional move. It is of course a big issue in computational creativity to make a machine that creates emotionally impressive piece inspiring something in human mind connected with viewer's private life or social affairs.

An easier extension is on the method of combination among different measures. The system introduced here is using geometric means because we thought all of the measures should be necessary conditions. We should examine another style of combination such as weighted summation, minimum and maximum among them. More complex combination of these logical operations might be effective. It might be also interesting to introduce some methods developed in the field of multi-objective optimization (Deb 2001), as (Ross, Ralph, and Zong 2006; den Heijer and Eiben 2014) examined. An effective method must be introduced to produce pieces of wider variation, if we use more generations, or the eternal evolutionary process, for production.

Another extension we should try in not far future is on the aesthetic measure of motion in the temporal sequence of pictures. We introduced very simple method to estimate the speed of motion in order to reduce the computational cost, but it must be replaced with some statistical analysis based on a type of optical flow. The techniques to extract distribution of 2D vectors of flow in the motion picture are originally developed for detection of the camera movement and an object moving in the captured scenery. But it must be useful to measure the interestingness of motion.

To provide a test bed for the research on computational aesthetic measures, it might be valuable to develop a mechanism of software plug-in to add a third party module for evaluation. It will make it easier to examine and compare the researchers' ideas.

Conclusion

Our experimental project of automated daily production of evolutionary audio visual art was introduced above. We have a lot of tasks to be conducted toward the machine that produces impressive art pieces. The author hopes this project inspires some ideas for the artists and researchers interested in creativity of human and/or machine.

References

Deb, K. 2001. *Multi-Objective Optimization using Evolutionary Algorithms*. John Wiley & Sons.

den Heijer, E., and Eiben, A. E. 2010. Using aesthetic measures to evolve art. In *WCCI 2010 IEEE World Congress on Computational Intelligence*, 4533–4540.

den Heijer, E., and Eiben, A. E. 2014. Investigating aesthetic measures for unsupervised evolutionary art. *Swarm and Evolutionary Computation* 16:52–68.

Galanter, P. 2003. What is generative art? complexity theory as a context for art theory. In *Proceedings of the 6th Generative Art Conference*, 76–99.

Galanter, P. 2012. Computational aesthetic evaluation: Past and future. In McCormack, J., and d'Inverno, M., eds., *Computers and Creativity*. London, UK: Springer-Verlag. chapter 10.

Koza, J. R. 1992. *Genetic Programming: On The Programming of Computers by Means of Natural Selection*. Cambridge, MA: MIT Press.

Machado, P., and Cardoso, A. 2002. All the truth about NEvAr. *Applied Intelligence* 16:101–118.

Machado, P.; Romero, J.; and Manaris, B. 2007. Experiments in computational aesthetics: An iterative approach to stylistic change in evolutionary art. In Romero, J., and Machado, P., eds., *The Art of Artificial Evolution: A Handbook on Evolutionary Art and Music*. Berlin Heidelberg: Springer-Verlag. 381–415.

Matkovic, K.; Neumann, L.; Neumann, A.; Psik, T.; and Purgathofer, W. 2005. Global contrast factor – a new approach to image contrast. In *Computational Aesthetics 2005*, 159–168.

Newman, M. E. J. 2006. Power laws, Pareto distributions and Zipf's law. *arXiv.org* (cond-mat/0412004).

Pearson, M. 2011. *Generative Art: A practical guide using Processing*. Manning Publications.

Ross, B. J.; Ralph, W.; and Zong, H. 2006. Evolutionary image synthesis using a model of aesthetics. In *WCCI 2006 IEEE World Congress on Computational Intelligence*, 3832–3839.

Satoh, H.; Ono, I.; and Kobayashi, S. 1997. A new generation alternation model of genetic algorithms and its assessment. *Journal of Japanese Society for Artificial Intelligence* 12(5):734–744.

Sims, K. 1991. Artificial evolution for computer graphics. *Computer Graphics* 25:319–328.

Takagi, H. 2001. Interactive evolutionary computation: Fusion of the capacities of EC optimization and human evaluation. *Proceesings of the IEEE* 89(9):1275–1296.

Unemi, T. 2009. Simulated breeding: A framework of breeding artifacts on the computer. In Komosinski, M., and Adamatzky, A. A., eds., *Artificial Models in Software*. London, UK: Springer-Verlag, 2 edition. chapter 12.

Unemi, T. 2010. SBArt4 – breeding abstract animations in realtime. In *WCCI 2010 IEEE World Congress on Computational Intelligence*, 4004–4009.

Unemi, T. 2012a. SBArt4 for an automatic evolutionary art. In *WCCI 2012 IEEE World Congress on Computational Intelligence*, 2014–2021.

Unemi, T. 2012b. Synthesis of sound effects for generative animation. In *Proceedings of the 15th Generative Art Conference*, 364–376.

Unemi, T. 2013. Non-stop evolutionary art you are embedded in. In *Proceedings of the 16th Generative Art Conference*, 247–253.