Ibero-American Symposium on Computer Graphics - SIACG (2006)
P. Brunet, N. Correia, and G. Baranoski (Editors)

ArTbitrating JaVox: Evolution Applied to Visual and Sound Composition

A. S. Moroni[1], J. Manzolli[2] and F. Von Zuben[3]

[1]Robotics and Computer Vision Division, CenPRA, Brazil
[2]Interdisciplinary Nucleus for Sound Studies, Unicamp, Brazil
[3]School of Electrical and Computer Engineering, Unicamp, Brazil

Abstract

ArTbitrariness is presented here as an initiative of upgrading the esthetical judgment through interactive evolutionary computation techniques and other population based techniques. Computational creativity will be approached, moreover evolutionary algorithms will be described and applied to simulate creativity. ArTbitrating JaVox, an evolutionary interactive environment for artistic production in visual and sound domains will be presented. Besides, the features of the evolutionary systems, that were incorporated in ArTbitrating JaVox, VOX POPULI, in sound domain, Art Lab and Shape, in visual domain, will be described.

Categories and Subject Descriptors (according to ACM CCS): J.5 [Arts and Humanities]: Fine arts; H.5.5 [Sound and Music Computing]: Systems.

1. Introduction

"Can computers themselves be creative, as opposed to merely producing an apparently creative performance, whose originality is wholly due to the human programmer?" The first person to denounce this apparent absurdity was Ada, Lady Lovelace, the friend and collaborator of Charles Babbage. She realized that Babbage´s "Analytical Engine" – in essence, a design for a digital computer – could, in principle, "compose elaborated and scientific pieces of music of any degree of complexity or extent." But she insisted that the creativity involved in any elaborate pieces of music emanating from the Analytical Engine would have to be credited not to the engine, but to the engineer. As she put it, "The Analytical Engine has no pretensions whatever to *originate* anything. It can do [only] *whatever we know how to order it* to perform." [Bod96]

But what is creativity? Creativity is a puzzle, a paradox and some say a mystery. Artists and scientists rarely know how their original ideas come about. They mention intuition, but cannot say how it works. How could science possibly explain fundamental novelties? Sometimes, creativity is explained as the combination of familiar ideas in unfamiliar ways. In other cases, it involves the exploration, and sometimes the transformation, of conceptual spaces in people´s minds [Bod96; Bod98]. Other authors affirm that an idea or product that deserves the label "creative" arises from the synergy of many sources and not from the mind of a single person [Csi96].

Turing recognized the importance of creativity, whatever the definition of intelligence, when he attempted to answer Lovelace´s objection in his seminal paper "Computer Machinery and Intelligence", the same paper in which he introduced his famous test for machine intelligence [Tur50]. Turing suggested that objections to the possibility of computers being creative of the type put forward by Lady Lovelace were based on a common misunderstanding of the nature of reasoning of the mind, resulting in an over-statement of the powers of rational thought. In particular, Turing pointed out that "a person knowing a set of facts and rules about the world does not mean that the person immediately knows all of the implications of applying the rules to the facts".

Turing suggested that a better variation of Lovelace´s objection would be that a machine could never *take us by surprise* but he then proceeded to declare that computers often surprised him because of his own faulty understanding of what he had *ordered them to perform*. In making this argument, Turing tried to show that the engineer would be no more responsible for the creativity of a machine than the machine itself because the engineer could not predict the creative behaviour at design time.

Turing´s argument did not provide much information about the possible processes involved in creative thinking but it did highlighted the importance of emergence, novelty, and surprise in computational models of creativity. [SG02]. Until today, the answer to Lovelace´s question seems to be *no* [Sea80, Bod96], but computational creativity is enormously interesting and potentially important. The psychology of creativity can benefit from artificial intelligence and computer science *precisely because* – as Lovelace pointed out – *a computer can do only what its program enables it to do*. On one hand, computational concepts, and their disciplined expression in programming terms help us to specify generative principles clearly. On the other hand, computer modeling helps us to see, in practice, what a particular generative system *can* and *cannot* do [Bod96].

Nowadays, computational creativity covers a wide spectrum of subfields. At one end of the spectrum lies the study of creativity as a psychological and social phenomenon, by computational methods; at the other end lies the study of computational assistance for creative people. Many applications related to the area of computational creativity can be found in [BC02; Bet99; GT99]. Here, we try to bring and model some kind of creativity into computers to permit the user/artist and computer work together interactively, producing results that could not be produced independently.

Next, we define ArTbitrariness and its context. Then, we present the systems that gave rise to the concept of ArTbitrariness, VOX POPULI, in sound domain, and Art Lab, in visual domain. Aspects of music computing are commented. Evolutionary computation and systems applied to the visual domain are depicted. ArTbitrating JaVox, and its features will also presented. Finally, initiatives of automating criticism and the conclusion are presented.

2. What is ArTbitrariness?

Since that Lovelace posed her question several authors tried in some way to bring creativity into computer systems by applying different techniques, like production systems, shape grammars and, more recently, genetic algorithms or evolutionary computation [Coh99; McC91; TL92; TL99; SG02; Sim93; Sim99]. Evolutionary algorithms have been applied to a wide range of creative design problems [Bet99] with such a great success that resulted in some researchers to speculate that they modeled creativity [Gol99], although most commentators are very cautious and do not make such claims without some reservations [BC02].

The concept of ArTbitrariness arose from the attempt of computationally emulate creativity applied to artistic production in the visual and sound domains. Two composition systems were developed, VOX POPULI, in sound domain, and Art Lab, in visual domain. Interesting results appeared from both. Emergent questions are: what criteria could be applied to *automate* when looking for *creative* composition? What *assures the quality* of a composition? How to recognize an *interesting result*, or how to supply a system with an *automatic judgment capability*?

All these questions addresses aesthetical appreciation. Are there rules that guide aesthetical appreciation? In music, for examples, chords are associated to consonance rules. Because if aesthetical appreciation would be governed only by subjective opinion, it would not be possible to obtain (partially) automatic shapes of artistic production, with some aesthetical value, without a complete integration of the user with the machine. On the other hand, if general rules did not allow the maintenance of a set of liberty degrees of expression, there could be complete automation, in spite of the possible complexity of design.

Since none of the extremes seems to appropriately describe the artistic production process, one may conclude that there is space to automate the exploration of the liberty degrees of expression through a man-machine interaction, such as in the attendance of general rules. In short, the liberty degrees can be modeled such as optimizing problems of combinatory mathematics. The general rules can be mathematically formalized and inserted in computational systems, as constraints or directions to be followed by the machine. The freedom of expression will be understood here as an exploratory search for the best combination of the free attributes among all possibilities. This scene is characterized by the existence of a huge number of possible solutions, or possible combinations of the free attributes.

After the proposition of a search space that contains the possible solutions, a search tool is applied to look for promissory regions in the space, in which there are possible good solutions or combinations of free attributes with more aesthetical value than others from less promissory regions.

There are very strong search algorithms. Among the factors, that justify the choice of evolutionary computation techniques, is the fact these algorithms apply *population search techniques*. But, independently of this, the search algorithms require the definition of an individual evaluation for each solution. The automation of the evaluation process requires that the machine be able to deterministically evaluate the aesthetical quality of each individual in the current cycle, or *generation*.

Instead of delegating this task to the machine, or to give the machine the evaluation capability, what is done here is to bring about an interaction with the artist/user, in such a way that the automatic solutions are presented to the artist and that he/she evaluates the solutions according to his/her subjectivity. In this sense, ArTbitrariness is interpreted as an iterative interactive optimization process. The main objective of arTbitrariness is to prevent to leave to the artist what (already) can be optimized and to prevent to leave to the machine what cannot be optimized (yet) [MZM02] . It can be said that arTbitrariness addresses an arbitrary point between subjectivity and objectivity, with its associated potential of automation as presented in figure 1.

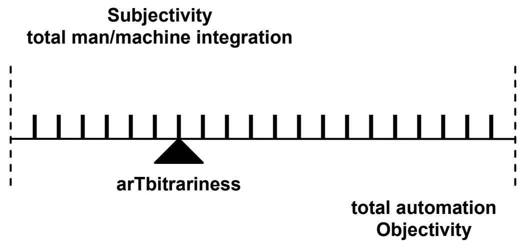


Figure 1*: arTbitrariness as an arbitrary point between subjectivity and objectivity.*

Next, the evolutionary environments, VOX POPULI, applied to artistic production in sound domain and Art Lab, applied to artistic production in visual domain, which gave rise to the concept of arTbitrariness, will be described.

3. VOX POPULI: an evolutionary environment for sound production

In the last decade a generation of music computing researchers has been discovering that by using simulated evolution techniques it is relatively easy to obtain novelty, often complex novelty, but it is correspondingly difficult to direct the flow that novelty takes. The challenge faced by the designers is how to bring more structure and knowledge into the evolutionary loop, while trying to take people out of the evolutionary loop [TW99]. In other words, it moves the pointer of the "arTbitrariness weighing-machine" to the direction of the total automation.

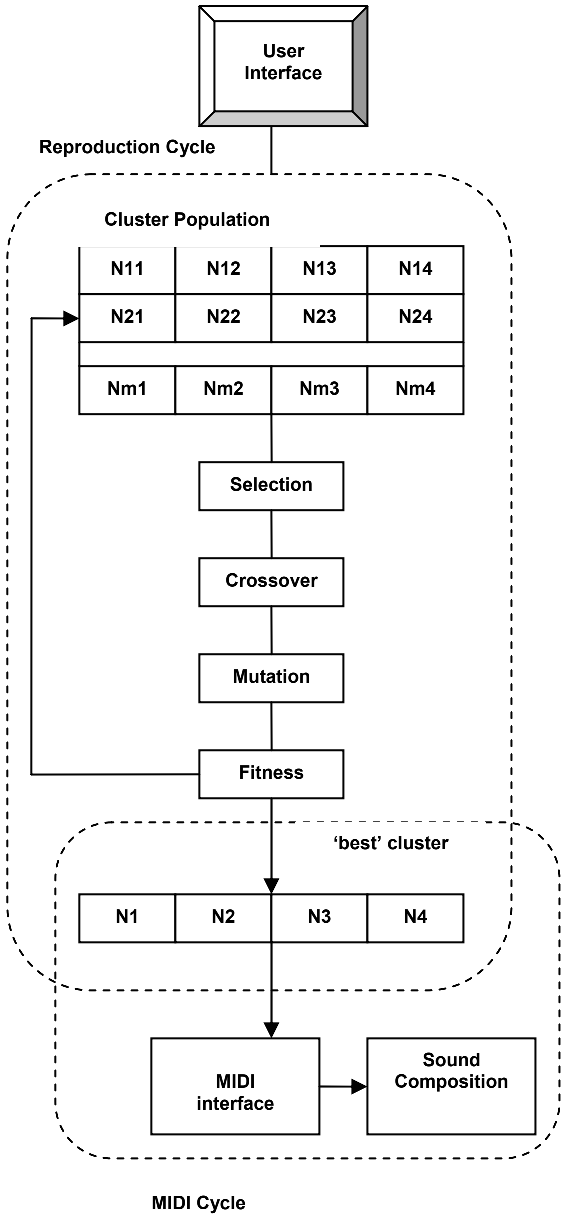


Figure 2: ***The evolutionary and interface cycles in VOX POPULI.***

But will it be really necessary or, at least desirable, to take people out of the evolutionary loop? This loop is a rather simple one: *generate, test, repeat*. Basically, in an evolutionary system, a cluster or *population* of things is created; the things are tested according to some criteria and the ones that are better according to the criteria are kept, and then the process is repeated by generating a new cluster of things based on the old ones. This loop continues for possibly many generations until the things that are being created are good enough according to the criteria being used. The complication comes when we have to specify what we mean by 'generate' and 'test'.

In natural evolution, what is being generated are individual organisms, through a process of genetic modification, usually either sexual combination or asexual 'cloning', both with some possible mutation. The criteria of success are the forces of natural and sexual selection, i.e. ability to survive and reproduce. Furthermore, in natural evolution there is no 'stopping point' when some criteria have been met; the test keeps changing as a consequence of ongoing evolution of other species as well. But what and how should those organisms be generated and tested when dealing with music composition systems? [TW99]

In VOX POPULI the population was made up of four note groups, or clusters, as potential survivors of a selection process. Melodic, harmonic and vocal-range fitnesses were used to control musical features. Based on the ordering of consonance of musical intervals, the notion of approximating a sequence of notes to its most harmonically compatible note, or tonal center, was used. The selected notes were sent to the MIDI board and could be heard as sound events in real time. This sequence produced a sound resembling a chord cadence or fast counterpoint of note blocks [MMZG00, MMZG02]. Two processes were running in parallel: a production process, the evolutionary cycle, supplying an individual from the population of clusters to be played, and a consuming process, the interface cycle, looking for a cluster to be played. Figure 2 shows the evolutionary cycle and the interface cycle in VOX POPULI.

VOX POPULI rises the computer and the mouse as real-time music controllers, acting as an interactive computer-based musical instrument. In VOX POPULI interface, an interactive pad supplies a graphical area in which bi-dimensional curves can be drawn. These curves are linked to the controls of the interface. One curve (red) links the melodic and octave range controls; and the other curve (blue) links the biological and rhythmic controls. Figure 3 and 4 shows different curves and their associated sound output. VOX POPULI interface was replicated in ArTbitrating JaVox, depicted in Figure 7.

VOX POPULI interface controls make use of nonlinear iterative mappings which give rise to attractors, defined as geometric figures that represent the set of stationary states of a dynamic system or simply trajectories to which the system is attracted. Modeling a piece as a dynamic system implies in a view in which the composer draws trajectories or orbits using the elements of each set [Man93]. Using different drawings, the composer can experience the generated music and conduct it, trying different "trajectories" or "sound orbits". The trajectories affect the musical fitness evaluation and the reproduction cycle of the genetic algorithm, that is being applied to the sound generation, as well as the MIDI interface cycle [MMZG00, MMZG02].

The next section describes Art Lab, an evolutionary environment applied to the production of abstract compositions in visual domain. General aspects of evolutionary systems are presented.

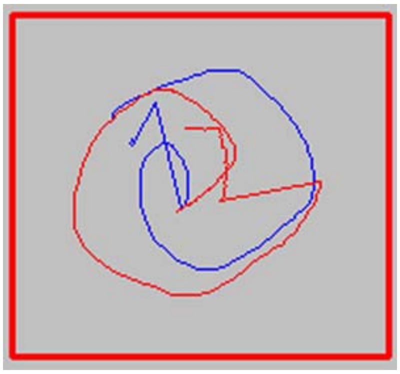


Figure 3: ***A simple curve generated by the Vox Populi pad interface***

Figure 4: ***The musical score generated by the curve on Figure 3.***

4. Art Lab, an interactive evolutionary system for abstract compositions

In 1991, Sims presented a novel approach for combining genetic algorithms with computer graphics. The system which Sims designed allowed users to evolve complex figures without concern for the mathematics used to generate the images. The interface was simple: given a number of initially random figures, the users selected the two which were the most interesting. These figures were used as ‘parents’ to produce a subsequent population of ‘offspring’, which possessed some attributes of both parent images. From the new population of images, two parents were selected, and the cycle continued. Through this iterated process of interactive selection, the images could become complex images which the users have evolved under their guidance [Sim91].

One of the most attractive aspects of this style of interactive graphics development is that it abstracts many of the cumbersome details of image production away from the users. The users are not required to know how the graphics are generated, how offspring are produced from two parent images or how the images are internally represented. The power of this method lies in the ability to direct the progress of evolution. Perhaps the easiest way to describe both how and why this process works is by analogy. By selecting some images to be the parents of the next generation, and not selecting others, the users create a bias in the evolution based upon their own likes and dislikes. The figures which are ‘strong’, with respect to the users’ tastes, are more likely to be selected as the parents of the next generation. Assuming that the parent ‘chromosomes’ (the internal representation of the images) have a means to pass their qualities to their children, the characteristics found in the parent images are also found in varying degrees in members of the subsequent populations.

Continuing this analogy, Darwin’s theory of survival of the fittest plays an integral role in explaining how subsequent populations become closer to the users’ preferences. The users’ preferences are the basis of the fitness function. Through a number of generations, the characteristics which the users do not find interesting will not be selected for recombination. Only the images which contain characteristics which the users find interesting will be selected; therefore, only these will influence the composition of subsequent populations [BPJ99].

In Art Lab this approach was applied to the generation and evolution of *geometric abstract frames* [MZM02]. Art Lab´s interface controls permits the user to generate set of four frames, at each time, of the common graphic primitives available in any programming environment: lines, boxes, arc, circles, ellipses, miscellaneous. Basically, each picture had a chromosome associated to it - its genotype – that could be briefly described as follows:

picture =

(width, height, background color, form1, form2,…)

where each form has its specific attribute fields, all randomly generated, corresponding to the fields of the associated geometric primitive.

At each iteration, a population of four abstract visual compositions is generated and presented to the user for evaluation. The user can attribute to each one a grade – the fitness - from 0 to 10, the default was 0. After the evaluation, the user can evolve the population of pictures. Only those frames with fitness > 0 are considered for evolution, and four new frames are created from the old ones.

For this to happen a selection function is applied. One method for doing this assigns a probability of being selected to each individual in proportion to their relative fitness. That is, a solution with a score of 10 is 10 times more likely to be chosen than a solution with a score 1. This proportional selection is also sometimes called *roulette wheel selection* because a common method for accomplishing this procedure is to think of a roulette wheel being spun once for each available slot in the next population, where each individual has a slice of the roulette wheel allocated in proportion to their fitness score [MF98].

This approach is applied to Art Lab, where the individual is an abstract composition, as well as to VOX POPULI, in this case, to select the ‘best cluster’ according to the melodic, harmonic and vocal range criteria. Once the individuals are selected from the new population by using the roulette wheel, a crossover operator is applied upon them. The chromosomes of two successive compositions in the population are taken. Since in Art Lab the chromosomes may differ in length because they could have a different number of geometric forms inside them, a random number r is generated such that $1 \leq r \leq l$, where l is the length of the shortest chromosome. Both chromosome strings are cut and crossed at the l point. Below, if $1 < r < n$, *picture3* and *picture4* resulting from the crossover operator may be:

picture1 = (width1, height1, background color1, object11,…, object1n, … object1m)

picture2 = (width2, height2, background color2, object21,…, object2n)

picture3 = (width1, height1, background color1, object11,…, object2n)

picture4 = (width2, height2, background color2, object21,…, object1n, … object1m)

After the application of the crossover operator, a mutation operator is applied over the population, according to a mutation probability. Usually, in nature, this probability is very small but since evolutionary algorithms only exist in a computer, there is no reason to rely on real biological constraints. For example, in the vast majority of sexual organisms (not including bacteria), mating occurs only between pairs of individuals, but evolutionary approaches can rely on 'mating' or 'blending' more than two parents, perhaps using all of the individuals in the population of each generation to determine the probabilities for selecting each new candidate [MF98]. The user designs the mutation operator, according to the application. In VOX POPULI it changes a MIDI note from an individual cluster. At Art Lab, the mutation operator can change the size of the frame, the background color, a geometric primitive or an attribute of a geometric primitive.

Figure 5 shows the evolution of two abstract compositions. On the top are the parent compositions and below are the descendents. Note that the descendents present characteristics of both parents.

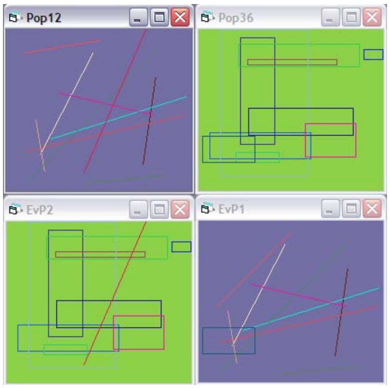


Figure 5: ***Evolution of abstract images at Art Lab.***

The best known evolutionary algorithms is the genetic algorithm. A genetic algorithm that uses human judgment to determine fitness is called an interactive genetic algorithm (IGA), in reference to its interactive interface. This interface typically promotes the presentation of the individuals in the current population for the user's evaluation. In visual domains, where each individual typically appears as an image, all individuals are presented at once, often reduced in size so that the entire population can be viewed simultaneously [TW99]. The user can compare and contrast the images concurrently and assign the fitness of each individual relative to all the others.

Evolutionary computation is a search. There is an interesting observation to be made regarding the other search techniques: each relies on a single solution as the basis for future exploration. Evolutionary algorithms are a recent and rapidly growing subset of the search methods that, instead of working with one solution at a time in the search space, consider a large collection or population of solutions at once. A commonly used term in this context is "optimization", which just means "finding" the best [BC02]. In computer science and in artificial intelligence, when a search algorithm is applied, the computational problem is defined in terms of a *search space*, which can be viewed as a massive collection of potential solutions to the problem. Any position, or point, in the search space defines a particular solution and the process of search can be viewed as the task of navigating that space [KC88].

In VOX POPULI, each chord is a point or solution in the 'chord space'. In Art Lab, each abstract composition is a point in the composition space. In Art Lab, when desired to intensify a local search, a memetic algorithm can be applied [Mos99]. Memetic algorithms are a marriage between a global search based in populations and a heuristic local search to each one of its individuals. A mutation operation, for example, is a local search since that it works with a single "current best" solution - in opposite to a population of solutions - upon which it will try to improve for the next step

The features of VOX POPULI and Art Lab environments are being merged in ArTbitrating JaVox, applied to sound and visual production. Next, ArTbitrating JaVox will be described.

5. ArTbitrating JaVox: Merging Evolutionary Environments

In ArTbitrating JaVox, the features of both environments, VOX POPULI and Art Lab, initially developed in Visual Basic, are being merged in a Java application. Like in Art Lab, ArTbitrating JaVox has facilities for the automatic generation of abstract compositions, that can be translated to sound trajectories, like in VOX POPULI. Figure 6 shows both evolutionary visual and sound cycles. In the visual cycle, visual compositions can be created and evolved.

When the Play button at ArTbitrating JaVox interface is pressed, the user can hear the sonorous composition generated by means of the abstract visual composition by JaVox. Now, there are two evolutionary cycles, in sound domain and in visual domain. Moreover, the abstract compositions in the visual domain can result in trajectories for production in sound domain. Like in VOX POPULI, while the blue and red trajectories are traversed, the x and y coordinates of each curve are assigned to the melodic, biologic, rhythm and octave (mel, bio, rhy and oct) controls that guide the fitness evaluation. It is possible to see the controls moving.

Figure 7 shows an abstract composition generated by ArTbitrating JaVox. Figure 8 presents the associated sound trajectories generated by the interactive pad control in the interface of ArTbitrating JaVox.

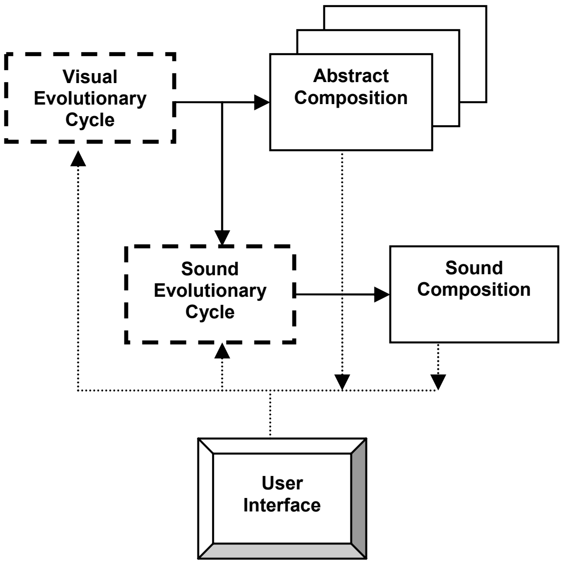


Figure 6: ***ArTbitrating JaVox schematic diagram.***

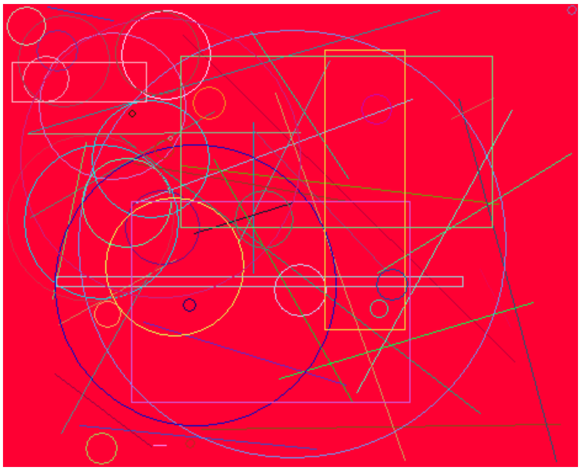


Figure 7: ***Abstract composition generated at JaVox.***

New problems arise: how to associate the visual features with sound features? Here, we retake the question placed in section 1: if a machine could create, what would it create? Somehow it seems that the basic vocabulary of forms of a computational system would be the elementary geometric forms. Certainly, this is a completely *arbitrary* decision, but once taken to exercise the computational creativity in the visual domain, the second decision is to choose Kandinsky to study. It influenced the fact that, besides being an abstract artist, Kandinsky tried to relate the visual and sound domains. In his famous book, *Concerning the Spiritual in Art*, Kandinsky [1977] stablishes a parallel between color, form and music. Kandinsky´s more adjusted work to our requirements is presented in figure 9.

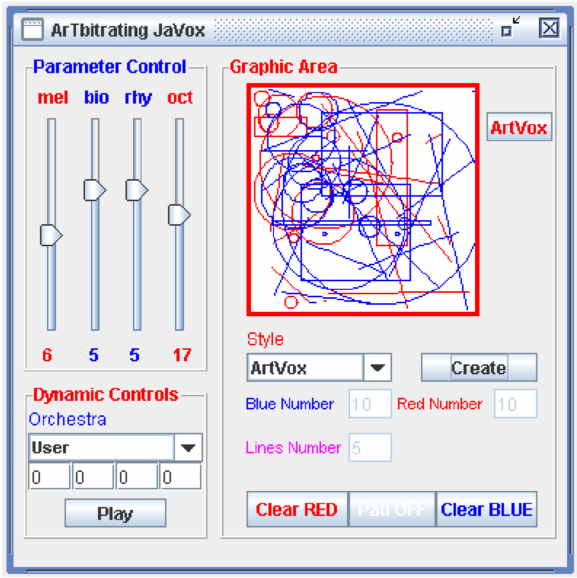


Figure 8: ***Sound trajectories generated from the abstract visual composition at ArTbitrating JaVox.***

Departing from this composition, visual objects are being created in an auxiliary environment, Shape, in Java. In figure 10, Kandinsky's "strange" compositions are presented.

Shape is only beginning but has presented such interesting results that is becoming an independent visual environment. Object editing facilities were added to Shape in order to apply transformations of translating, scaling and rotation of the objects.

Conclusion

Evolution is now considered not only powerful enough to bring about biological entities as complex as human beings and conciousness, but also useful in simulation to create algorithms and structures of higher levels of complexity than could easily be built by design.

The concept of ArTbitrariness as an iterative interactive optimization process for upgrading the esthetical judgment through evolutionary computation techniques and other population based techniques for exploratory search have presented. The environment ArTbitrating JaVox, an evolutionary environment for visual and sound composition emerged from two other evolutionary environments, VOX POPULI, an interactive environment for computational composition, and Art Lab, applied to visual domain. The features of both environments, VOX POPULI and Art Lab are being merged in ArTbitrating JaVox, in order to enable the system to produce visual and sound compositions. Another environment, Shape, is emerging in order to test more sophisticated objects, to be included at ArTbitrating JaVox. ArTbitrating JaVox and Shape are available at:

http://www.geocities.com/artbitrating

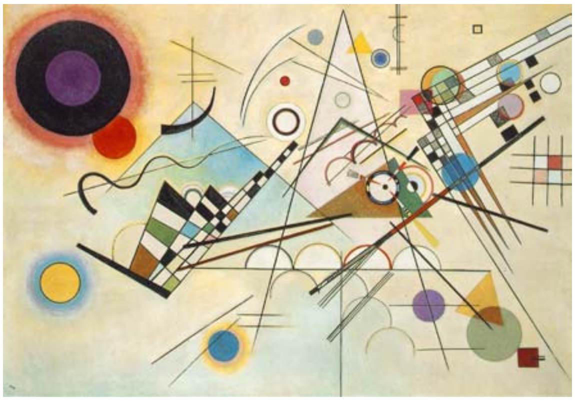


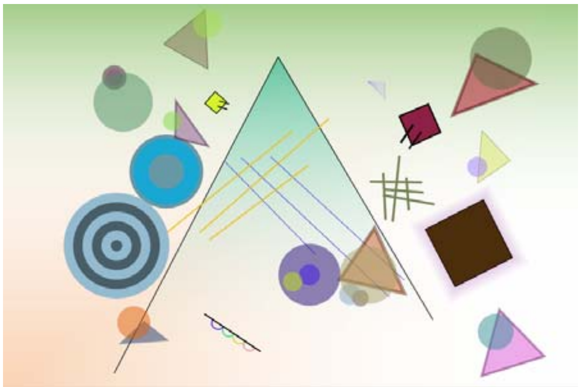
Figure 9. ***Composition VIII 1923 (140 Kb); Oil on canvas, 140 x 201 cm (55 1/8 x 79 1/8 in); Solomon R. Guggenheim Museum, New York***

Next Steps

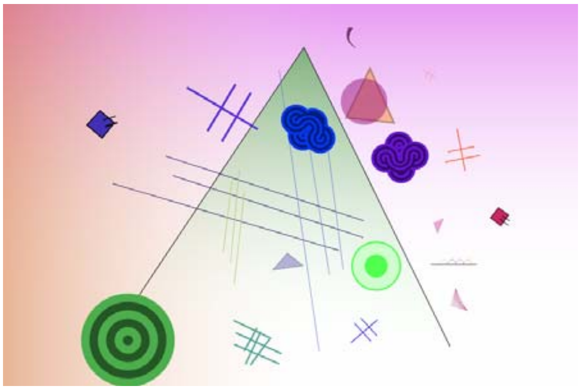
To further remove or at least transform the necessity of human interaction in the evolutionary process, some authors, working in the visual domain, have trained a neural network to replace the human critic in an interactive image evolution system [BPJ99]. The network watches the choices that a human user makes when selecting two-dimensional images from one generation to reproduce in the next generation, and over time learns to make the same kind of aesthetic evaluations as those made by a human user. When the trained network is put in place of the human critic in the evolutionary loop, interesting images can be evolved automatically. Using learning critics of this sort, whether applied to images or music, the artificial creators will end up being simplified because the structure will be acquired indirectly through trained fitness-evaluating critic that has learned its structural preferences from a user-selected training set.

Acknowledgements

We would like to thank Rafael Bocaletto Maiolla, Laface de Almeida and Daniel Gurian Domingues, who worked in the development of JaVox and ArTVox environments. We would like to thank PIBIC/CNPq program and CenPRA, for making this research possible. This research work is part of the AURAL project, supported by FAPESP process 05/56186-9.



(a)



(b)

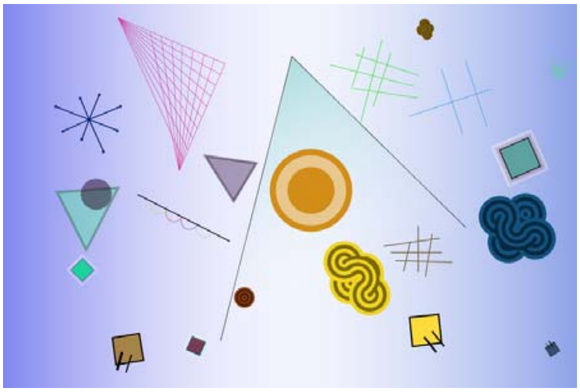


Figure 10. ***Pictures a, b and c, above, present Kandinsky's like objects generated at Shape auxiliary environment***

References

[BC02] BENTLEY, P., CORNE, D. (eds.): *Creative Evolutionary Systems*. Morgan Kaufmann, 2002.

[Bet99] BENTLEY, P. (ed.): *Evolutionary Design by Computers*. Morgan Kaufmann, 1999.

[Bod96] BODEN, M.: *Dimensions of Creativity*. The MIT Press, 1996.

[Bod98] BODEN, M.: Creativity and artificial intelligence. In Elsevier Science: *Artificial Intelligence* 103 (1998) pp. 347 – 356.

[BPJ99] BALUJA, S., POMERLEAU, D., JOCHEM, T.: Towards Automated Artificial Evolution for Computer-generated Images in Griffith, N. & Todd, P. M. (eds) *Musical Networks: Parallel Distributed Perception and Performance*, The MIT Press (1999), 341 – 370.

[CSI96] CSIKSZENTMIHALYI, M. *Creativity: Flow and the Psychology of Discovery and Invention.* HarperPerennial, 1996.

[Coh99] COHEN, H.: Colouring without Seeing: a Problem in Machine Creativity, (1999). http://crca.ucsd.edu/~hcohen.

[Gol99] GOLDBERG, D. E.: The Race, the Hurdle, and the Sweet Spot: Lessons from Genetic Algorithms for the Automation of Design Innovation and Creativity in Bentley, P. (ed.) *Evolutionary Design by Computers*, Morgan Kaufmann (1999).

[GT99] GRIFFITH, N., TODD, P. M. *Musical Networks: Parallel Distributed Perception and Performance*, The MIT Press, 1999.

[KC88] KANAL, L., CUMAR, V. (eds.): *Search in Artificial Intelligence.* Springer-Verlag, 1988.

[Man93] MANZOLLI, J.: Non-Linear Dynamics and Fractal Geometry as a Model for Sound Synthesis and Real Time Composition. Ph. D. Dissertation, UK: University of Nottingham, 1993.

[McC91] MCCORDUCK, P.: Meta-Art, Artificial Intelligence and the Work of Harold Cohen. Freeman, 1991.

[MF98] MICHALEWICZ, Z., FOGEL, D. B.: *How to Solve It: Modern Heuristics.* Springer, 1998.

[MMZG00] MORONI, A., MANZOLLI, J., VON ZUBEN, F.J., GUDWIN, R. Vox Populi: An Interactive Evolutionary System for Algorithmic Music Composition. *Leonardo Music Journal,* 10 (2000), 49-54.

[MMZG02] MORONI, A., MANZOLLI, J., VON ZUBEN, F.J., GUDWIN, R. "Vox Populi: Evolutionary Computation for Music Evolution" in Bentley, P. (ed.) *Creative Evolutionary Systems*. Morgan Kaufmann (2002), 205 – 221.

[MZM02] MORONI, A., VON ZUBEN, F.J., MANZOLLI, J.: ArTbitration: Human-Machine Interaction in Artistic Domains", Leonardo 35(2) (2002), 185 – 188.

[Mos99] MOSCATO, P.: Memetic Algorithms: A Short Introduction in Corne, D., Dorigo, M. and Glover, F. (eds.) *New Ideas in Optimization.* McGraw-Hill (1999), 219 – 234.

[SG02] SAUNDERS, R., GERO, J.: *How to study artificial creativity.* ACM Press, 2002.

[Sea80] SEARLE, J.: Minds, Brains and Programs. *The Behavioral and Brain Sciences,* vol. 3, Cambridge University Press, 1980.

[Sim91] SIMS, K.: Artificial Evolution for Computer Graphics. SIGGRAPH'91 Conference Proceedings, 25 (1991), 319 – 328.

[Sim93] SIMS, K.: Interactive Evolution of Equations for Procedural Models. The Visual Computer, 9 (1993), 466 – 476.

[Sim99] SIMS, K.: Evolving Three-Dimensional Morphology and Behaviour in Bentley, P. (ed.), *Evolutionary Design by Computers*, San Francisco: Morgan Kaufmann, 1999.

[TL92] TODD, S., LATHAM, W.: *Evolutionary Art and Computers.* Academic Press, 1992.

[TL99] TODD, S., LATHAM, W.: The Mutation and Growth of Art by Computers" in Bentley, P. (ed.), *Evolutionary Design by Computers*. Morgan Kaufmann, 1999.

[Tur50] TURING, A. M.: Computing Machinery and Intelligence, Mind (1950), 59, pp. 433 – 460.

[TW99] TODD, P. M., WERNER, G. M.: Frankensteinian Methods for Evolutionary Music Composition in Griffith, N. & Todd, P. M. (eds) *Musical Networks: Parallel Distributed Perception and Performance*, The MIT Press (1999), 313 – 339.

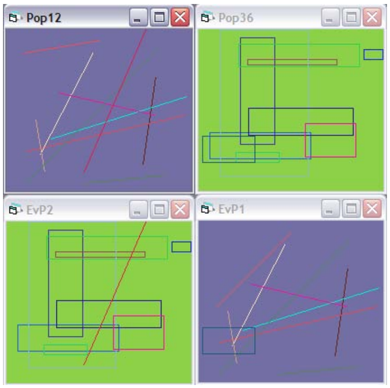


Figure 5: ***Evolution of abstract images at Art Lab.***

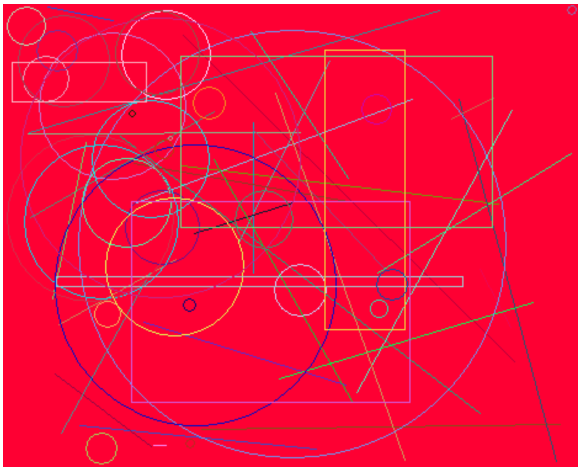


Figure 7: ***Abstract composition generated at JaVox.***

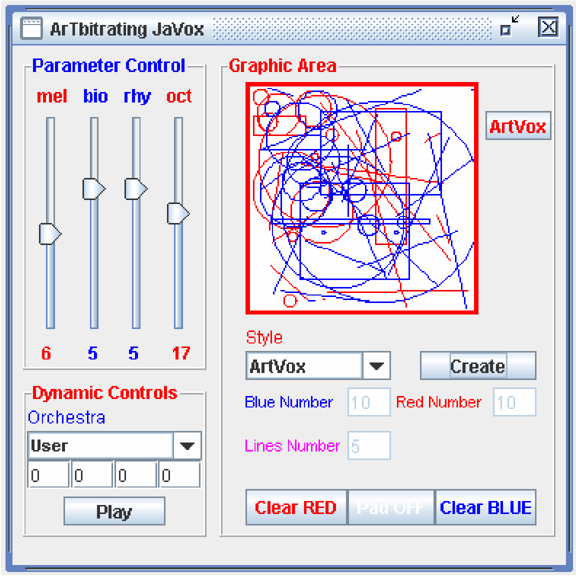


Figure 8: ***Sound trajectories generated from the abstract visual composition at ArTbitrating JaVox.***

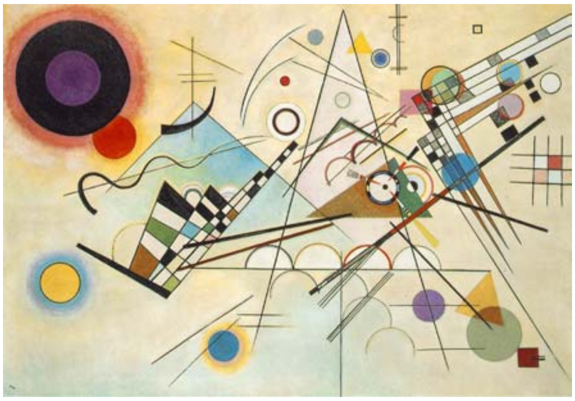
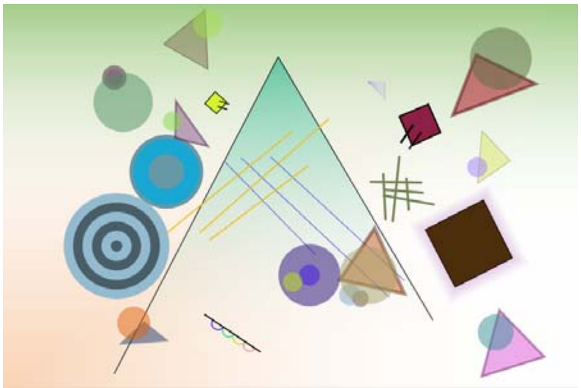
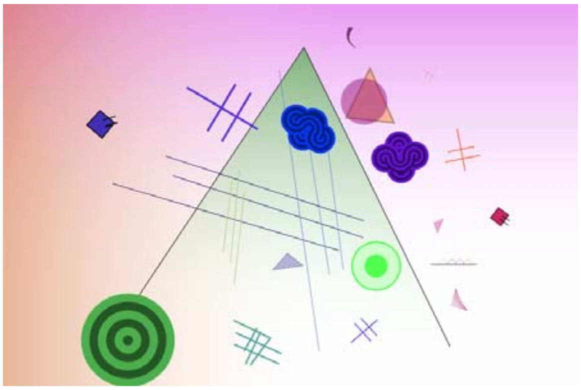


Figure 9. ***Composition VIII 1923 (140 Kb); Oil on canvas, 140 x 201 cm (55 1/8 x 79 1/8 in); Solomon R. Guggenheim Museum, New York***



(a)



(b)

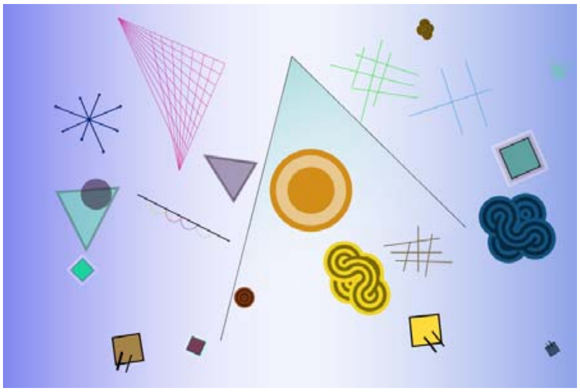


Figure 10. ***Pictures a, b and c, above, present Kandinsky's like objects generated at Shape auxiliary environment***