PROPAGATING FIGURES

A genetic algorithm approach

TSUNG-HSIEN WANG
Graduate Institute of Architecture, National Chiao Tung University, 1001, Ta Hsueh Road, Hsinchu, 30050, Taiwan
Michael@arch.nctu.edu.tw

Abstract. In order to make the computer media get involved into the process of evolving design rather than merely serve as a presenting tool, this study aims to propose a computer-aided design (CAD) tool that could show a great capacity of evolving design and be particularly utilized throughout the early conceptual phase. How to accomplish this objective will start from taking advantage of the cognitive point of view regarding emergent subshapes to assisting designers in reconfigurating/restructuring shapes and figures computationally. Ultimately, by means of using genetic algorithm, this research could conduct a CAD tool, facilitating designers in the very early conceptual phase. In other words, this tool could be employed by designers in generating subshapes or reconfigurating shapes and subshapes during the process of developing concepts.

1. Emerging subshapes cognitively and computationally

Since design process has been investigated considerably, several crucial orientations are proposed. The most well-known of them are problem-solving and seeing-moving-seeing models of design process (Simon, 1981; Schon, 1992). These two theoretical models provide the consequential researches a very significant foundation. Especially, after the seeing-moving-seeing model having been proposed, there are numerous explorations relevant into visual thinking in design process occurring. Afterwards, visual thinking becomes a significant direction in the domain of design thinking. Whereas visual thinking has been regarded as a predominant phenomenon in design process, abundant following researchers endeavour to examine how visual thinking takes place inside designers and how these visual feedbacks affect them, particularly in those investigations of the early conceptual phase(Gross, 1996). Further, this visual reasoning process can also be used to articulate the communicational phenomenon between designers and drawings (Soufi, 1996).

With amply explorations in visual thinking, one critical phenomenon, which we termed it shape emergence, in this visual thinking process was proposed and pertained to the characteristic features of shapes, uncertainty and ambiguity (Mitchell, 1992; Soufi, 1996; Suwa, 1999; Stiny, 2001; Testa, 2001). These characteristics of shape emergence account for the reason why and how subshapes could be extracted from the existing shapes. For example, Liu suggested that shapes could be categorized into explicit, implicit, closed and unclosed shapes and so on and further proposed that the quantity of subshapes pertains to the designer's experience. In brief, these phenomena signify a predominant relationship between emergent subshapes and the conceptual development process.

Sequentially, from computational point of view, shape grammar could be simplified as a mechanism to generate shapes or subshapes by means of symbolic notations and rule-applications. As rules were applied in computations, shapes manifested the mutual relations specifically (Stiny 1993). Therefore, subshapes could be recognized from the boundary of primary shapes. In addition to these finite shapes, Gero (1993) proposed another data-driven symbolic model and explored more emergent subshapes that are not necessary to be embedded into the primary boundary. Liu (1996b) further suggested a connectionist system trained to recognize these emergent subshapes, including explicit, implicit, closed or unclosed ones. What is more, abundant researches re-examined emergent subshapes from various perspectives, i.e., Oxman (2002) proposed a binary perspective which comprises perceptual and cognitive components to specify the emergence in design process and Knight (2003a, 2003b) tackled the issues of emergence and ambiguity in shapes from the grammar perspective.

All these researches provide an integral foundation to inspect this significant phenomenon of shape emergence in visual thinking not only from cognitive but also from computational point of views. At the same time, what the more important thing is that shape emergence has an inseparable relationship with creativity as well and still not to clear up yet (Soufi, 1996;Ueda, 2001; Oxman, 2002; Knight, 2003).

However, with the rapid development in artificial intelligence, more and more intelligent exploitations in the design researches arise to simulate how a human designer could do during design processes (Bentley, 2002). Such as so-called evolutionary computation, it submits a feasible way to emulate how the way designers think. Among these mechanisms, genetic algorithm, originally proposed by Holland, J.H., has been substantially applied into diverse disciplines. It was primitively inspired from the Darwinian principle of nature selection. Through closely mutual competition, those individuals who have better fitness would have a great probability to survive. Besides, from another respect inside genetic operations, no matter how gene is selected and further operated, it is manifest that between parents and offspring exists a crucial relationship of inheritance and similarity. (This relationship would be analogized to examine the relationship between shapes and subshapes afterwards.) In more details, the evolutionary process consists of the gene structure of a chromosome, being manipulated by several operational and control function sets, the fitness measure, the terminal criterion, etc. During such an evolutionary process, an optimal solution could be possibly generated and presented by means of the fitness survival. Particularly, those intangible and ambiguous problems have been proved to be solved successfully and efficiently by using genetic algorithm (Man, Tang and Kwong, 1999). On the other hand, it signifies the prospect of dealing with the issues concerning shapes, those exactly having a characteristic feature of ambiguity and uncertainty in design.

2. Issues on emerging and computing

Design, in a sense, could be regarded as a search based on visual processing. Particularly in the early conceptual phase, visual interactions happen frequently between designers and drawings. In general, concepts and contemplation in design are recorded and externalized by means of designers' drawings and sketches and further are utilized for sequentially analyzing and reconsidering. Moreover, especially these figures or shapes, existing in drawings or sketches, are all-time incomplete and ambiguous during the early conceptual phase. Multitude examinations also regard those figures would further have a profound influence on the following design. As is specified above, the principal issue addressed in this study is how to propose a mechanism for facilitating designers in generating figures and shapes from the visual thinking perspective, especially focusing on the early conceptual phase. On the side, whereas there is still insufficient in CAD tool that could provide shape generations automatically during the early conceptual phase, this generating mechanism expects to be further integrated into a modeling application to make this tool more accessibly.

2.1. EMERGENT SUBSHAPES IN EARLY CONCEPTUAL PHASE

Owing to the ambiguity and uncertainty of shapes during the developing period of design, subshapes could be extracted from those existing shapes. Mitchell (1992) regarded an emergent shape as "a shape that exists only implicitly in a primary shape, and is never explicitly input and is not represented at input time.” Such an ill-articulated or half-formed feature provides the following design infinite illusion and probability and makes design proceed. Else, in light of Liu’s (1995) statements, there are distinct differentiations between expert and novice designers in recognizing explicit and implicit shapes. The more the complex of shapes is, the more time and efforts are needed in recognizing them and he even proposes a special variable that is termed as “threshold of recognizing activation” (TRA). This parameter denotes how subshapes be recognized differentiating from their previous experience. Therefore, since emergent subshapes is such an important phenomenon in design process and, at the same time, proved the existence of such a limitation caused from experience, the leading objective of this CAD tool is going to be utilized to facilitate designers in propagating more emergent subshapes during the early conceptual phase.

2.2. TWO STEPS IN EMERGING SUBSHAPES

Minsky(1986) suspects that “ the way we perceive the world , from one moment to another, depends only in part on what comes from our eyes: the rest of what we see from inside our brain.” As a result of Minsky’s statement, it implies that there are two parts of shapes. One is from what we perceive from the external stimuli and the other is what we deliberate inside our internal transformations, those inside our brain. As the same situation during shape emerging, there are also two obvious steps regarding how to recognize a subshape (Liu, 1995). First of all, a designer looks at given shapes, so-called the external stimuli, and then restructures what they see and what they are contemplating in their minds, so-called the internal transformation. That is to say, the first step could be regarded as to see (perceiving stimuli) and the second one is to restructure (transforming).

Notwithstanding, symbolic and connectionist approaches have been employed on this issue tremendously. Yet, there is insufficient in recognizing subshapes. The former ones submitted an efficiently computational way in generating numerous shapes by means of notations and rule-applications and the latter one attempted to recognize shapes in terms of connecting vast artificial neurons from an opposite respect. Both of them, without doubt, address some advantages in recognizing or generating figures. However, the same problem of them is that their mechanisms in recognizing or generating shapes and subshapes could merely be used to solve either generating or recognizing section. In addition, symbolic processing takes too many steps and unnatural ways in generating a simple shape, and connectionist processing spends too much time in straining that might be lead to performance inefficiency. To recognize such an emergent subshape is no more than routine trivia of human designers. *Figure 1.* shows how symbolic processing generates a scheme by applying several rules.

Through skilful manipulations, it takes 14 steps and 6 rules to transform a rectangle into a simplified layout. Obviously, there exists one contradiction that seldom people generate their layouts from a simple rectangle but from the context of the design criteria, special needs to a restructured scheme and usually a simple geometry combination. On the other hand, designers always have to perceive some stimuli and then react. They could not just convert all these contemplations into rules and steps. Applying rules is one efficient way. However, it seems to be too arbitrary of rule-makers in manipulating rules to generate such a layout. On the contrary, these rules and steps should be manipulated more automatically. In addition, connectionist processing could neither be used to recognize a simplified layout from a pure rectangle. There exists one premise before connectionist processing proceeds: connectionist processing starts from modelling human brain function. Since that, it needs more clues in advance to respond to. In other words, this fashion could be merely used for recognizing shapes from those existed ones. Clearly, as stated above, shape-emergence consists of two major steps and connectionist approach is insufficient in the second step, a restructuring step. In sum, symbolism and connectionism are beneficial for generation and recognition respectively. However, there still needs one more sound mechanism available in solving both considerations.

In this research, genetic algorithm was proposed and used for simulating a process of gene evolution through restructuring its chromosome, including several genetic operations, the fitness measure and so on. From certain respects, the process of emergent subshapes is similar to gene evolutionary process. The relationship between genotype and phenotype is analogous to that between shapes and restructured shapes. The latter one, phenotype or restructured shapes, always inherited part or sum of parts of characteristics from the previous one (genotype or existed shapes). Sometimes, the discrepancy is so tiny that it’s difficult to be detected. To simplify this, the composition of a gene is analogized to be the first seeing step, an external stimuli, in emerging shapes and subshapes. This step decides what part of given shapes is concerned. Sequentially, propagating chromosomes of gene is served as the second restructuring step, an internal transformation. Ultimately, this study would propose how emergent subshapes could be extracted from the original ones more deeply by using genetic algorithm and further testify the efficiency and validity of implementing genetic algorithm into a CAD tool.

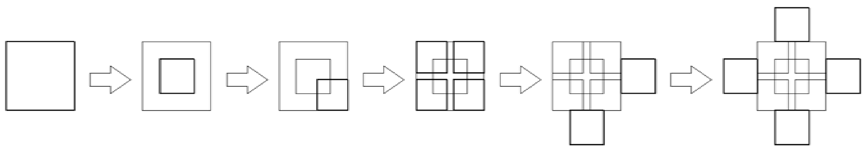


Figure 1. A simplified processing of generating parti of Villa Rotond.

In conclusion, the major objective of this research is providing a CAD system for designers during the very early conceptual phase. Moreover, this tool is going to be used for augmenting the search space of emergent subshapes of designers by means of using genetic algorithm in recognizing and reconfigurating shapes and subshapes. In terms of the expansion of the emergent behavior, designers are stimulated and could evolve further design.

3. Methodology and steps

In this research, in order to achieve the objective of providing a CADsystem recognizing and reconfiguring shapes by using genetic algorithm, there are several principal procedures as follows. The first step is to construct the knowledge representation of this system. In this section,there include two major sub-procedures. One is the data structure of input and output, and the other is to submit the structure of a chromosome presentation. This structure of a chromosome would be displayedby a string of values in binary form. On the side, this binary string will further be used in the sequential operational process. Following is to specify this system structure explicitly and concisely.

3.1. DATA STRUCTURE

In this step, two major sub-procedures are included. One is the data structure of this program, and the other is to submit the most basic operational chromosome of this system. As follows:

3.1.1. Two Kinds Of Vertices

First of all, I first try to represent these entire figures simply by vertices. For example, a line consists of two vertices and similarly, an enclosed plotlines-shape could be viewed as one circuit of four vertices. Therefore, after designers draw some figures in working environment proposed, this system would first interpret this figures into the primitive vertex data consisting of two dimensional coordinate values (*Figure 2a.*). (In order to specify the whole process concisely, this study takes two-dimensional figures as the example in this paper.)

After coding all vertices embedded in the original figure, this section is trying to reveal other invisible vertices, conducted by the intersections from extending lines (*Figure 2b.*). Therefore, the most fundamental data, vertices embedded (VE) and invisible vertices (IV), are completely obtained.

3.1.2. Operational Chromosome

After encoding the figures into coordinate values, there is still an indispensable step needed to be conducted in advance before having these data into the reasoning machine. That is to construct a basic operational unit, a chromosome for the latter genetic operations, and this unit comprises either vertices embedded or invisible vertices or both of them. In other words, these units are a set of four vertices; furthermore, what the most important thing is this smallest rectangle should never be intersected by any other extending lines (*Figure 3.*).

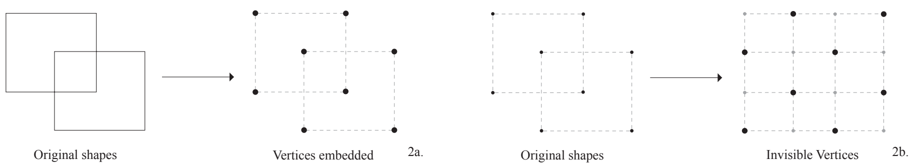


Figure 2. Vertices embedded (2a) in original shapes and invisible vertices (2b).

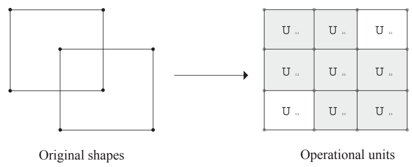


Figure 3. Operational units (chromosome).

Ultimately, all these data could be converted into a structure of a chromosome and further take advantage of a string of values in binary form to represent itself. Every bit of this binary string presents every operational unit respectively. As the initial stimulus could be coded in a chromosome with a binary string, the fundamental part of whole system is complete and sequentially manipulated by the genetic operations ceaselessly until the fitness solution is generated.

3.2. GENETIC OPERATIONS

In order to consolidate the fundamental genetic phenomena with the formulation of genetic algorithm (GA) more closely, this formulation comprises several characteristic operations, such as crossover, mutation, selection, fitness function, the terminal criterion, etc. All operations manipulated in this study are described below:

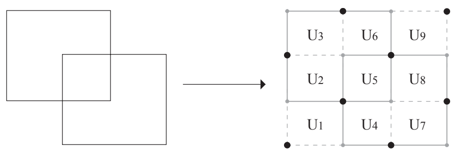
3.2.1. Condition Specification

The main problem of the program in this study is to facilitate designers in generating more shapes or subshapes during the design process. As a result, a gene-like mechanism is going to be used to propagate more shapes or subshapes through such a population-based search.

3.2.2. The Fitness Measure

In the beginning, I have to review the TRA values stated above in advance briefly. This value, a specific parameter of evaluating the threshold of expert designers from novice designers in recognizing subshapes, is sufficient to tell the discrepancy between expert and novice designers apart from the cognitive point of view. Furthermore, echoing this phenomenon, this study initiate with assigning weights distinct through the completeness of figures and the position of figures individually and further expects to simulate how designers recognize subshapes in design computationally.

Thus, as all the operational units have been evaluated by the fitness measure above, they are ranked by the grade of completeness. That is to say, the more complete the shape is, the higher weight it will score. More specifically, evaluating the grade of the completeness of a shape is in light of the amount of lines comprising two kinds of lines, lines embedded and invisible lines. (Lines embedded are part of or the sum of parts of lines embedded in the original shapes. On the contrary, those invisible lines, which wouldn’t exist at the input time, are extended from the original shapes. These two lines are given different weights according to constructive ways individually.) For example, *Figure 4*. denotes different weights assigned to all these operational units. The weight of U_5 is higher than U_1 because the amount of lines embedded (LE) is much more than those of invisible lines (IL). Complying with this rule, all operational units get weights separately.



$U_5 = 4 > U_4 = U_2 = U_8 = U_6 = 3 > U_3 = U_7 = U_1 = U_9 = 2$

Figure 4. Weights.

In addition, the other essential contemplation is on the basis of the position of every operational unit. Those units located inside the boundary of original shapes would plus another positive value. Moreover, the fitness measure takes account of the complex of units. *Figure 5.* shows how these three fitness measures work. Eventually, a whole weight-assigning task is complete. In sum, the fitness measure proposed here is used to rank the differentiation of offspring for providing the future terminal criterion with reference. Therefore, this search could easily avoid long and ceaseless computation in advance.

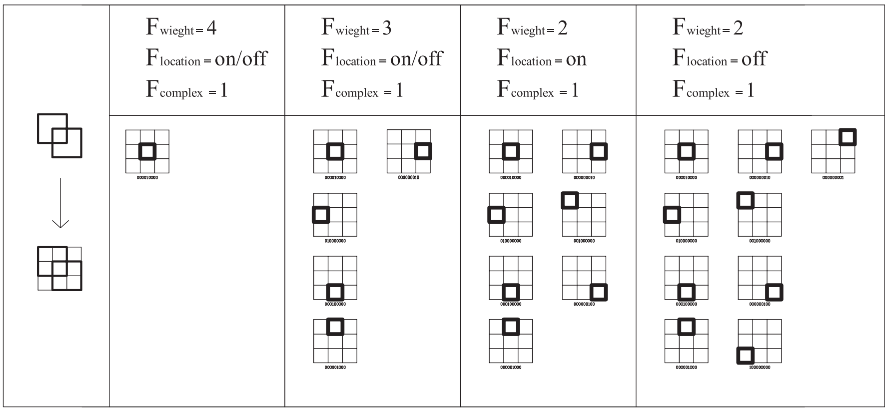


Figure 5. A demo shows how fitness measures work.

3.2.3. Selection Methods

During this section, I propose a mechanism to eliminate those parents with lower fitness in the mating pool and prevent those low-fitness offspring from being generated. Therefore, this step is often operated prior to genetic operations.

3.2.4. The Set of Genetic Operations

All genetic operations applied are addressed below. Three crossover methods and one mutation and one reproduction are discussed as follows:

1. One-point crossover: A single crossover point with the range [1, 9] could be randomly selected. (*Figure 6.*)

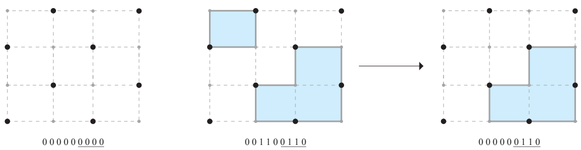


Figure 6. One-point crossover.

2. Multi-point crossover: From two to five points crossover is also allowed in this program. (*Figure 7.*)

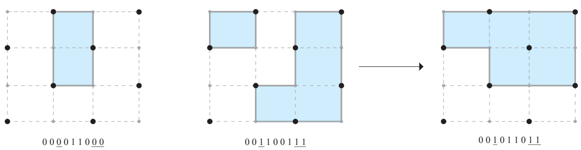


Figure 7. Multi-point crossover.

3. Heuristic crossover: This means first converts the binary string into integer and further produce another binary string, offspring, by means of the following formula (Where β∈[0, 1], *Figure 8.*):

$$\text{Offspring} = \text{parent1}*\beta + \text{parent2}*(1-\beta) \tag{1}$$

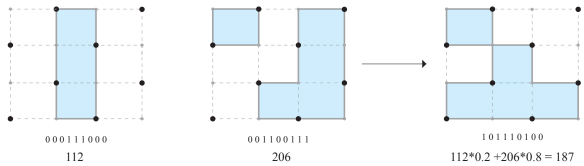


Figure 8. Heuristic crossover.

4. Mutation: During the genetic operation, mutation plays such an important role in providing the program with the possibility of making progress to an optimal solution. In general, this operation is seldom performed and chosen by random. (*Figure 9.*)

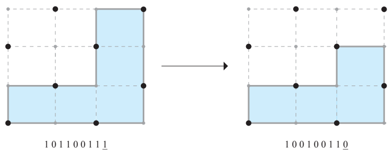


Figure 9. Mutation.

5. Reproduction: According the fitness measure of individual in the mating pool, every genotype has a chance to be duplicated to the next generation.

Whereas every operator has his own specific capability to alter the generations, how to negotiate with all these operators becomes the most significant matter in every run. Consequently, the operation rate is addressed to solve this problem. The alterable operation rate with a value between 0.8 and 1.0 is regarded as the probability of crossover. The other two asexual operations, mutation and reproduction, are normally performed sparingly and therefore the operation rate is approximately 0.1 or less than 0.1, a lower probability.

3.2.5. Certain Parameters of Controlling

In addition to the fitness measure, this program also suggests other two controlling parameters for users. One is the maximum value of the population size and the other is the maximum value of generation.

3.2.6. Termination

A crucial relationship between terminational criteria and the fitness measure is discussed here. In practice, the criterion of termination is exercised for determining whenever the program continue or terminate and further designating the result of the run. In other words, by means of terminational criteria could ascertain that offspring propagated have reached a plateau. Particularly in this study, this program is contrived to terminate either when:

1. a specific value of offspring, which might be assigned at the beginning of the run by users, is reached. This means a comparatively better fitness is achieved where the total weights of the population is equal or bigger than the criterion; or
2. the controlling parameters regarding the population size or generation size is reached.

According to the assignment of users of the termination criteria, this program would propagate satisfactory results, shapes or subshapes, as best as it could during a run.

4. Application and analyses

Taking advantage of the genetic mechanism proposed above, this program exercises a genetic operation concerning shapes emergence or reconfigurations of shapes and subshapes. The process of this program is specified through a flowchart below (*Figure 10.*). In addition, *Figure 11.* illustrates a sample run conducted to simulate what Petra did during a design process from the drawing first submitted by Schon and Wiggins (1992). Through serial genetic operations, including crossover, mutation, and so on, could this program generate the same output of Petra's.

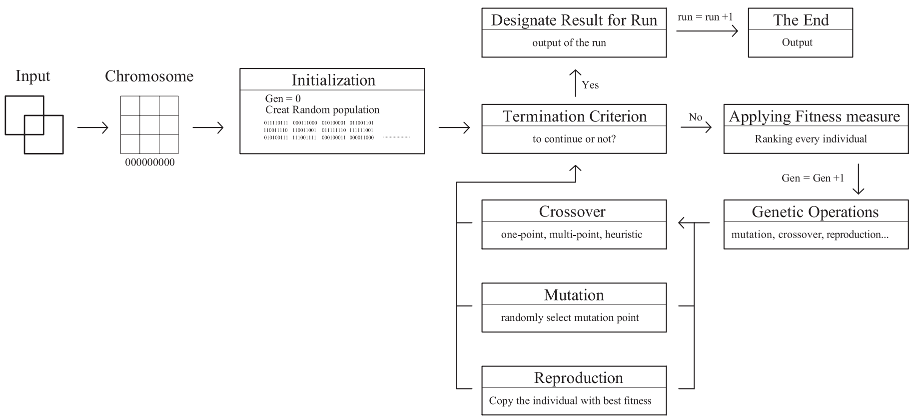


Figure 10. Flowchart of the program

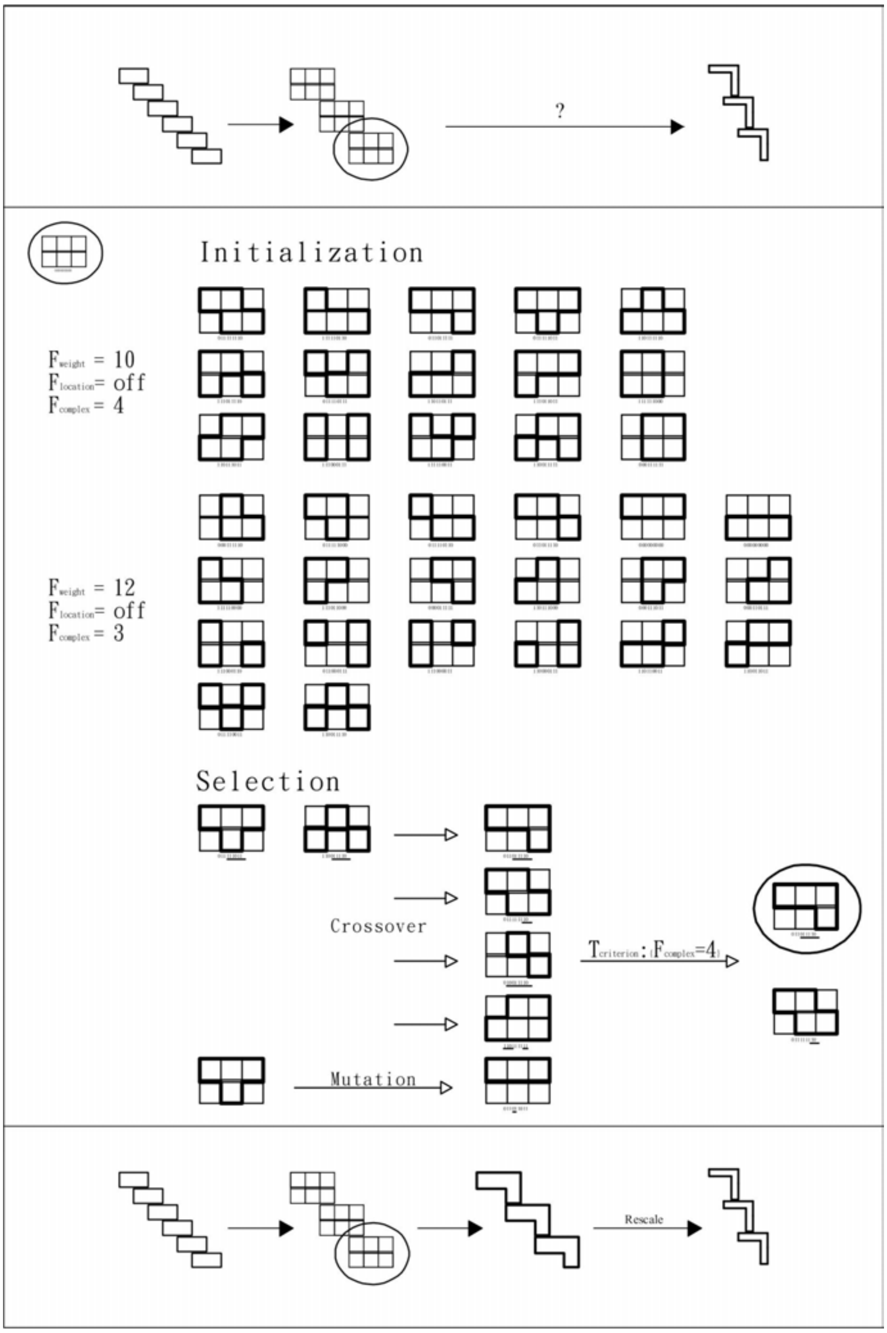


Figure 11. Demo.

In addition to symbolic and connectionist computation of shapes, this study proposes another computational approach, genetic algorithm, for solving the problem in symbolic or connectionist approach. Clearly, this approach shows a probability in generating shapes that are produced from the nature-like selection. Without applying rules and training, this program could help designers in propagating more possible and feasible shapes or subshapes in design. *Figure 11.* exercises a simple demo of this program. This is only a preliminary exercise. More extensive applications will be proposed, especially in three dimensional modeling environments. That is to say, no matter symbolic or connectionist approaches, there should go further into a practical application rather than theoretical explorations.

5. Concluding remarks and future studies

By means of combining the genetic algorithm with a search model of shape emergence, this generative tool is going to provide a computational environment for designers to use during the early conceptual phase, particularly in propagating more emergent shapes. Moreover, considering some inseparable connections to creativity (Soufi, 1996;Ueda, 2001; Oxman, 2002; Knight, 2003), this computer aided design tool signifies a possibility to improve the creativity in terms of amplifying the quantity of emergent shapes. In the future, as a plug-in extension of any application, taking Alias|Wavefront Maya package, it will provide a more eligible way for designers to exploit it during the design process and get designers and design computation more closely. However, the limitation in this study is that these objects available for recognizing and manipulating are all rectangles. The reason why I choose this specific geometric figure as the preliminary study is, during the subshapes emerging, those figures are more easily to be recognized at the first glance for designers. Rectangles are just one of the most well-known geometric figures. Therefore, in the near future, the following study is focused on providing a CAD tool with no limitations in all kinds of geometry figures and manipulations. Ultimately, this tool could be in a widespread use during the design process.

Acknowledgements

I would like to thank Professor Aleppo Liu at Chiao Tung University for his insightful direction and ponderable comments. Also, I appreciate all the instructors here for their enthusiastic support. Finally, I am grateful to all the friends for their helpful discussions throughout this study.

References

Bentley, P. and Corne, D.: 2002, *Creative Evolutionary Systems*: Academic Press.

Coyne, R. D.: 1990, *Knowledge-based design systems*.

Gero, J. S.: 1999, Recent design science research: Constructive memory in design thinking, *Architectural Science Review* 42, 3-5.

Gero, J. S. and Yan, M.: 1993, Discovering Emergent Shapes Using a Data- Driven Symbolic Model, *CAAD Futures93*, Pittsburgh, PP3-17.

Gross, M. D.: 1996, The Electronic Cocktail Napkin-a computational environment for working with design diagrams, *Design Studies* 17(1), 53-69.

Knight, T.: 2003, Computing with ambiguity, *Environment and Planning B: Planning and Design*, 30(2), 165-180.

Knight, T.: 2003, Computing with emergence, *Environment and Planning B: Planning and Design*, 30(1), 125-155.

Liu, Y.-T.: 1991, Schematic-Designer: a knowledge-based CAD system for schematic design in architecture, *Design Studies*, 12(3), 151-167.

Liu, Y.-T.: 1993, A Connectionist approach to shape recognition and transformation, *CAAD Futures93*, Pittsburgh, PP18-36.

Liu, Y.-T.: 1995, Some Phenomena of seeing shapes in design, *Design Studies*, 16(3), 367-385.

Liu, Y.-T.: 1996a, Is designing one search or two? A model of design thinking involving symbolism and connectionism, *Design Studies*, 17(4), 435-449.

Liu, Y.-T.: 1996b. Restructuring shapes in terms of emergent subshapes: a computational and cognitive model, *Environment and Planning B: Planning and Design*, 23(3), 313-328.

Man, K.F., Tang, K.S. and Kwong, S. (eds): 1999, Genetic Algorithms: Concepts and Designs, Springer-Verlag, Berlin.

McCormack, J. P. and Cagan, J.: 2002, Supporting designers' hierarchies through parametric shape recognition, *Environment and Planning B: Planning and Design*, 29(6), 913-931.

Mitchell, W. 1990.: *The logic of Architecture*, Cambridge, MA: MIT Press.

Mitchell, W. 1992.: The uses of inconsistency in design, *in* Y.E. Kalay (ed), *Principles of computer-aided design: evaluating and predicting design performance*, Wiley, New York. Pp. 1-13.

Oxman, R.: 2002, The thinking eye: visual re-cognition in design emergence, *Design Studies*, 23(2), 135-164.

Schon, D. A. and Wiggins, G.: 1992, Kinds of seeing and their functions in designing, *Design Studies*, 13(2), 135-155.

Simon, H. A.: 1981, *The Sciences of the Artificial*, Cambridge, MA: MIT Press.

Soufi, B. and Edmonds, E.: 1996, The cognitive basis of emergence: implications

for design support, *Design Studies*, 17(4), 451-463.

Stiny, G.: 1990, What designers do that computer should, *in* M. McCullough, W.J. Mitchell, and P. Purcell (eds), *The electronic design studio*, MIT Press, Cambridge, Pp. 17-30.

Stiny, G. and Mitchell, W. J.: 1978, The Palladian grammar, *Environment and*

Planning B: Planning and Design, 5(1), 5-18.

Testa, P., U.-M. O'Reilly, et al.: 2001, Emergent Design: a crosscutting research program and design curriculum integrating architecture and artificial intelligence, *Environment and Planning B: Planning and Design*, 28(4), 481-498.

Ueda, K.: 2001, Synthesis and emergence - research overview, *Artificial Intelligence in Engineering*, 15(4), 321-327.