THE EMERGENCE OF THE REPRESENTATION OF STYLE IN DESIGN

LAN DING AND JOHN S. GERO
Key Centre of Design Computing and Cognition
Department of Architectural and Design Science
University of Sydney NSW 2006 Australia
e-mail: {lan, john}@arch.usyd.edu.au

Abstract. This paper presents an evolutionary approach to the emergence of the representation of style in design. It explores issues involved in the interpretation of style, the emergence of the representation of style, and a computational process for emerging the representation of style. An evolutionary process model based on genetic engineering for style representation emergence is developed and demonstrated. Examples from its computational implementation are presented.

1. Introduction

Style in design is a recognisable phenomenon. It expresses certain characteristics of products and touches upon many areas such as structure, behaviour, function, society, culture, history, etc. Many studies have dealt with recognition and representation of style but have lacked any formal process for deriving and representing style. This paper develops a computational process to derive a computational representation of style.

REPRESENTATION OF STYLE

A style, in an object view, has been regarded as a set of products' common characteristics (Ackerman, 1967; Edwards, 1945). Common characteristics are related to form elements, relationships and qualities (Schapiro, 1961). A style, in a process view, has been considered to be one way of doing things, chosen from a number of alternative ways (Simon, 1975). The choice refers to the decision among possible alternatives (Meyer, 1979). Therefore, style can be represented in two ways: (1) products' characteristics or meanings; and (2) procedure features used in producing objects. The latter is regarded as an explicit representation of style. The derivation of an explicit representation of style is useful in design. It enables one to find the source producing style characteristics of products and to use them for reproducing designs with that style. This research focuses on the derivation of an explicit representation of style.

Recent studies of the explicit representation of style include Taiwanese traditional house style and Palladio style (Chiou and Krishnamurti, 1995; Stiny and Michell, 1978), which describe style in the form of rules of composition. These rules can be manipulated by the computer to generate designs with that style. However, these studies deal with style that is derived manually.

THE EMERGENCE OF THE REPRESENTATION OF STYLE

Emergence in design is the process of making properties explicit that were previously only implicit. Recent studies of emergence in design involve recognition and representation of emergent structure and behavior (Gero, 1992, Maher et al., 1993, Mitchell, 1989, Liu, 1996, Gero et al., 1995). Based on the notion of emergence in design, we propose the emergence of the representation of style, which is a computational process of finding an explicit representation of style that was previously implicit.

The study of style representation emergence emphases the interpretative aspect of style as well as the emergence process. A syntax-semantics model is developed to interpret style, where semantics is considered to be the implicit properties of style and syntax is regarded as the explicit representation of style. An evolutionary process is then modeled to be the emergence process, which finds the executions of syntax rules that produce a style through competition, genetic engineering and evolutionary combination.

The organisation of this paper is as follows. Sections 2 and 3 present an interpretation of style, which provides the basis for style representation emergence. An evolutionary approach to style representation emergence is introduced in Section 4. Section 5 demonstrates an evolutionary process model and Section 6 presents an example from a computational implementation of this model, resulting in a computer representation of an architectural style. The final section presents some conclusions. There is an Appendix to the paper that provides a list of definitions of technical terms commonly used in this paper.

2. Style Exemplified by Architectural Style

Our research provides an interpretation of style using a syntax-semantics model. Style in this paper is exemplified by architectural style.

We regard architectural style as the representation of common particular meanings called complex semantics within a group of designs. For instance, Gothic style refers to a set of common complex semantics: dynamic line, emphasis on spire, structural framework using stone to concentrate weights and stresses, etc. Common complex semantics is derivable from a set of lower level meanings, called simple semantics. Simple semantics is derivable from architectural forms. The same semantics could come from the same forms or from different forms. This results from the common decisions relating to form elements and relationships between form elements. The way style is produced is illustrated in Figure 1.

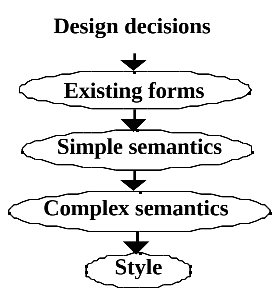


Figure 1. The production of style from design decisions.

We present Frank Lloyd Wright’s house style as an example. This style consists of a set of common complex semantics: dynamic configuration and linking with nature, Figure 2. These complex semantics are derivable from a set of simple semantics such as:

- parallel roofs
- natural materials
- horizontal axis
- etc

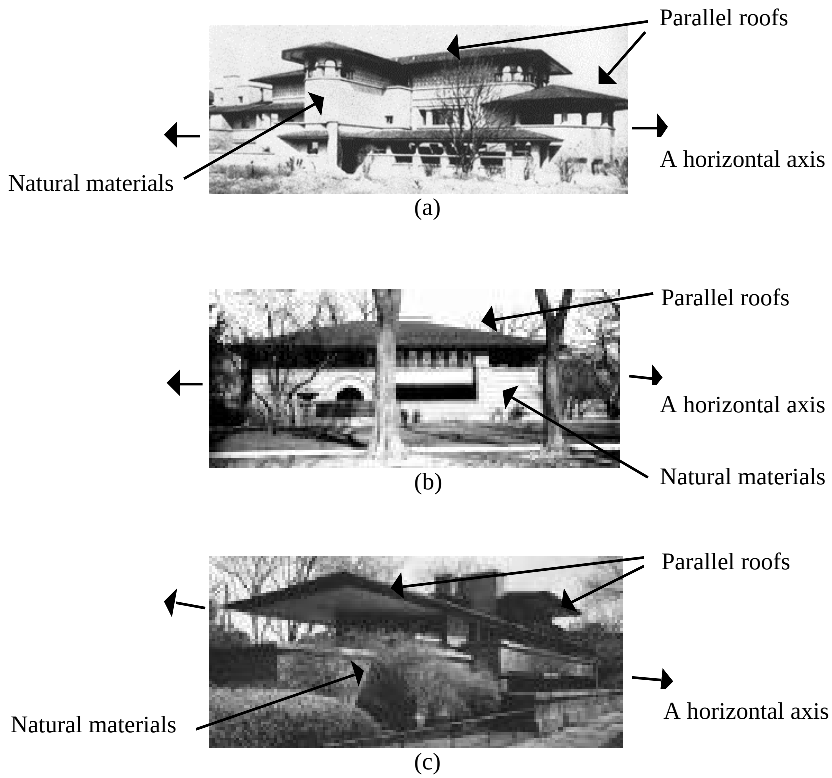


Figure 2. House style created by Frank Lloyd Wright.

(a) Husser House, 1899; (b) Heurtley House, 1902; (c) Robie House, 1909. (from Vincent Scully, Jr. (1985), *Frank Lloyd Wright,* George Braziller, New York).

These simple semantics are derived from architectural forms. They result from the common decisions relating to form elements and relationships between form elements, i.e.

- ‘parallel roofs’ (derived from the roofs) result from the decisions concerning particular lengths, widths and parallelarity,
- ‘natural materials’ (derived from the forms) result from the choice of materials and from the revelation of the nature of those materials,
- ‘a horizontal axis’ (derived from the forms) results from the geometrical layout with an emphasis on two axes.

These common design decisions were repeated in many of his other house designs. This results in the same style of these house designs, even though their forms are different.

The interpretation can be described using a syntax-semantics model. In a language model of design (Coyne and Gero, 1986), design parts are regarded as design vocabularies and design rules are considered to be design syntax. Syntax rules are applied to design vocabularies to produce design sentences. Executions of syntax rules produce a design as a context. Semantics is the interpretation of design as behavior.

According to the language model of design, we describe style space utilising hierarchical levels mapping onto syntax and semantics. Design vocabularies refer to form elements such as roofs and windows. Syntax rules refer to design rules. Design decisions are considered to be the decisions regarding form elements, syntax rules and executions of syntax rules. Simple semantics is derived from simple design forms that are produced by syntax rules. Complex semantics is derived from a set of simple semantics and design forms that are produced by the executions of syntax rules. Therefore, an architectural style is a set of common complex semantics and is determined by a particular set of syntax rules and executions of syntax rules.

Figure 3 contains examples of traditional Chinese architectural style. Its interpretation using a syntax-semantics model is illustrated in Figure 4.

(a) (b) (c)

Figure 3. An example of a traditional Chinese architectural style:
(a) Cideng Si Pagoda; (b) Great Enlightenment Temple; (c) Shaolin Si Pagodas.

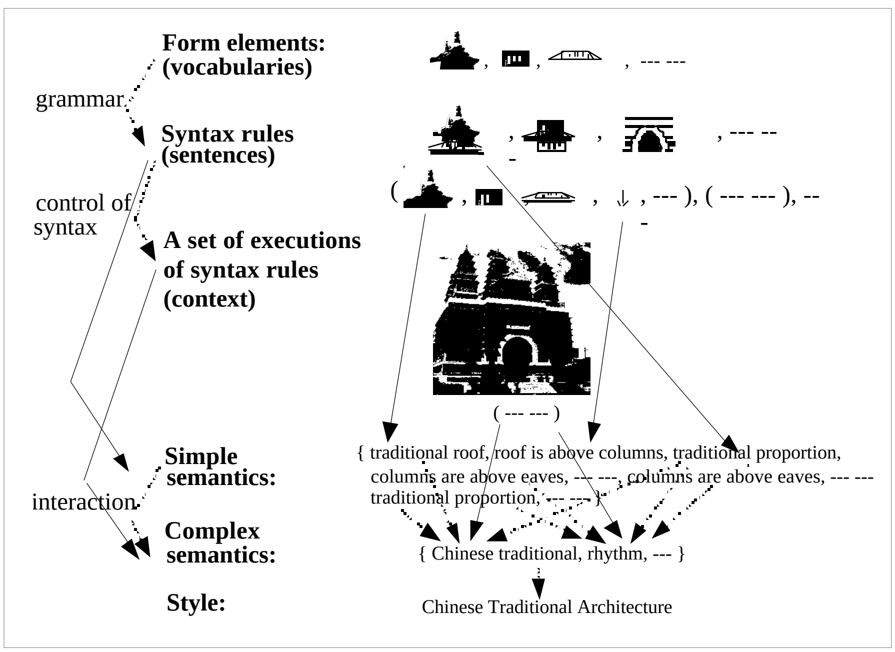


Figure 4. An example of interpreting architectural style using a syntax-semantics model.

3. Hierarchies and Correspondence of Style Space

The interpretation of style using the syntax-semantics model produces hierarchies of style space. The hierarchies involve semantics space and syntax space. Semantics space refers to behavior space while syntax space is the decision space. The hierarchies of semantics space involve simple semantics and complex semantics. An example of such a hierarchy of semantics space is presented in Figure 5.

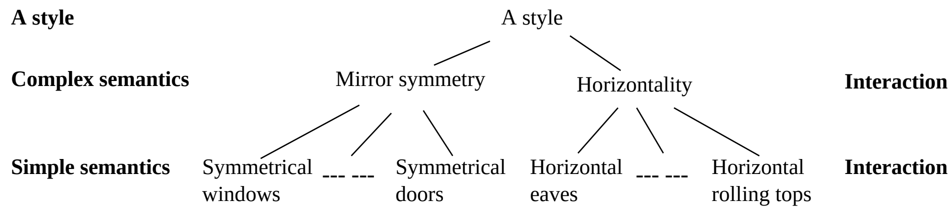
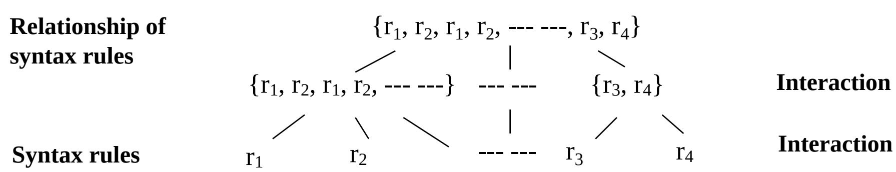


Figure 5. An example of a hierarchy of semantics space.

The hierarchies of syntax space involve syntax rules and control of syntax rules. The control of syntax rules refers to the relationships of syntax rules. An example of a hierarchy of syntax space is presented in Figure 6.



Where r_I denotes a syntax rule

r_I : rest a pyramidal roof on top of columns

r_2 : rest columns on top of podium

r_3 : place eaves at right of eaves

r_4 : place a rolling top at right of a rolling top

Figure 6. An example of a hierarchy of syntax space.

There are interactions of components in the hierarchies of semantics space or syntax space. The interaction among low level components results in a high level component. We represent the interaction as a relationship. A high level component can therefore be represented by a set of relationships of low level components.

There are hierarchical mappings between syntax space and semantics space. Syntax rules map onto simple semantics through the forms they develop, in form space, Figure 7.

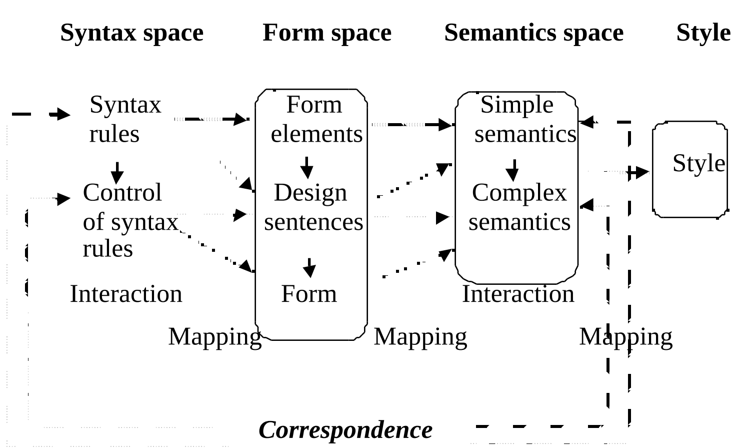


Figure 7. Hierarchical mappings between syntax space and semantics space.

The hierarchies of style space provide a framework for representing architectural style. Based on this, an evolutionary approach is developed for style representation emergence.

4. An Evolutionary Approach to Style Representation Emergence

We develop an evolutionary approach for style representation emergence, where both models of natural evolution and artificial intervention are considered. In this work the model of natural evolution refers to genetic algorithms while artificial intervention refers to genetic engineering.

4.1 GENETIC ALGORITHMS AND GENETIC ENGINEERING

Genetic algorithms (GAs) provide a computational model of simulating the Darwinian principle of 'survival of the fittest'. They have been widely used for solving design problems (Bentley, 1999). Genetic engineering (Gero, 1992) is an extension of the standard GAs model. It simulates the genetic engineering of natural organisms to find connections between genetic structure and highly successful performance. Genetic engineering has been successfully applied for design problems involving improving design state spaces and supporting design creativity (Gero and Schnier, 1995; Gero and Kazakov, 1996). Gero and Kazakov (2001) have shown that the genetic engineering techniques used here are computationally tractable and scale well.

A simple model of genetic engineering works as follows. The system seeds a candidate population and evaluates the individuals of the population against the fitness of the environment. It classifies the individuals into two subsets, those being good individuals with high fitness and bad individuals with low fitness. It then performs genetic engineering to locate the genetic structures linked to high fitness, which are in the good individuals but not in the bad individuals. These genetic structures linked to high fitness are turned into evolved genes. A new population is then produced utilising these evolved genes and initial genes through modified reproduction operations. The cycle is repeated until convergence is reached.

This research develops an evolutionary system applying genetic algorithms and genetic engineering for style representation emergence; the evolved genes and their relationships are the representation of the style. Its claimed advantages are:

(1) it can provide a rapid search in a large set of search spaces relating to a style;

(2) the mapping between syntax and semantics of style can be located through an evolutionary process based on genetic engineering, where a particular set of executions of syntax rules used in producing semantics of style can be derived to become the explicit representation of style; and

(3) the emergent representation of style can be learned and improved through evolution.

4.2 A FRAMEWORK

A framework for style representation emergence is presented, where a genetic description of style space and a hierarchical evolutionary process are proposed.

The genetic description includes translation of form elements and syntax rules into genetic codes resulting in genotypes, description of expressions transforming genotypes into phenotypes/designs, requirements of a style into the style environment and translation of the various components of style of these resulting designs into fitness functions. The mapping between semantics and syntax of style is transformed into the mapping between high fitnesses of the semantics and genetic structures.

This mapping can be derived through a hierarchical evolutionary process based on genetic engineering, Figure 8. We classify style environment as simple semantics, complex semantics and style. The process of emerging representation of style is performed in the genetic search space, where form elements and syntax rules are described as genes. The system randomly seeds a set of genes resulting in genotypes to generate design populations. Semantics in the populations is then examined against the environment. The system then locates the genetic structures (i.e. syntax rules and executions of syntax rules) that produce high fitnesses of the semantics suited to the style environment for the derivation of explicit representation of style.

This process proceeds hierarchically. Syntax rules mapping onto simple semantics are located, learned and evolved. They are then used to produce subsequent populations for complex semantics. The syntax rules and executions of syntax rules which map onto high fitnesses of complex semantics are then found, learned and evolved. These genetic structures that map onto simple and complex semantics are then used together with other genes to produce subsequent populations for a style. A set of syntax rules and executions of syntax rules mapping onto high fitnesses of that style is then derived.

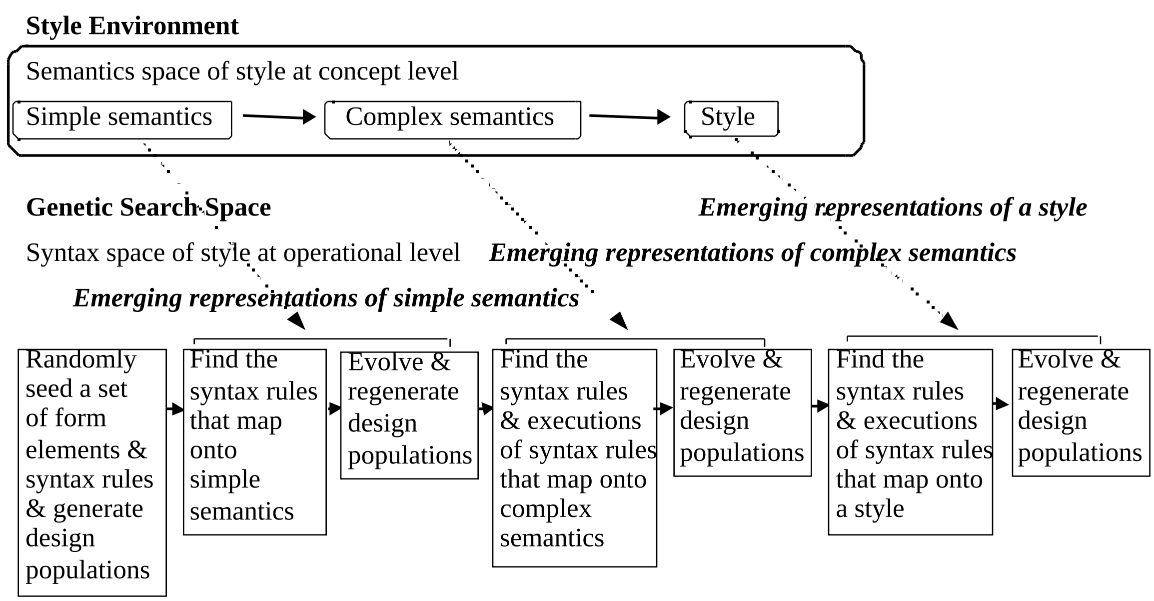


Figure 8. A framework for style representation emergence.

5. An Evolutionary Process Model of Emerging the Representation of Style

We develop an evolutionary process model in terms of this framework, where competition, genetic engineering and evolutionary combination are applied in the process of emerging the representation of a style, Figure 9.

The competition of natural organisms refers to the process of self-adaptation needed to survive in a certain environment. Here, we apply the competition process to search in syntax space and produce semantics suited to the style environment. It is used for the creation of alternative representations of style. The discovery process uses genetic engineering to locate the genetic structures (called evolved genes) which map onto high fitnesses of simple semantics, complex semantics and style. It is used for the discovery of the representations of style. After the evolved genes linked to high fitnesses of semantics have been found, an evolutionary combination is carried out. Those evolved genes are utilised together with the initial genes to produce a new combination for the improvement of the representations of style.

The cycle is repeated until a set of evolved genes that maps onto high fitness of a style is produced. The manipulations of competition, discovery and an evolutionary combination are demonstrated in the following sections.

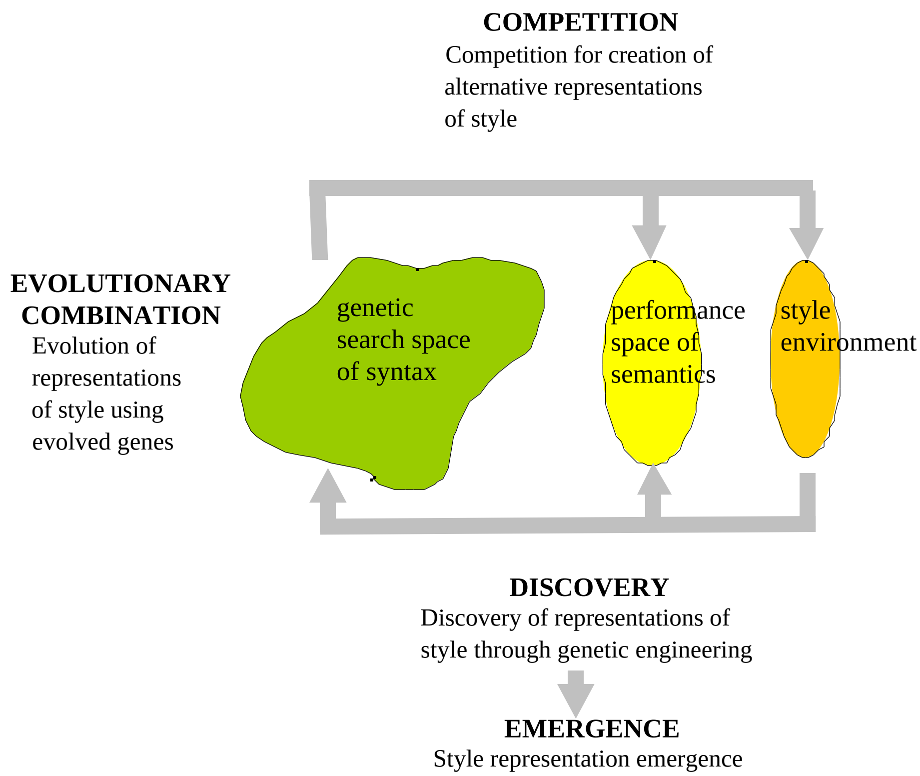


Figure 9. An evolutionary process model of emerging style representation.

5.1 COMPETITION

The competition process includes a search process and a detection process. The search process carries out genetic search for syntaxes. The detection process is concerned with measuring whether the semantics produced by the syntaxes is suited to the style environment.

1. Search process

The search process produces alternative syntaxes by employing a genetic mechanism that contains standard crossover, mutation and selection. The crossover operation performs the task of swapping syntax rules in the parent population, which results in a new combination of syntax rules of different individuals in the form of gene strings. A crossover point is selected at a random location. The mutation operation is used for randomly modifying syntax rules, which allows the production of possibly new syntax rules. The selection operation is concerning with selecting parents for the production of a subsequent population. We use the 'tournament selection' method.

The search process in syntax space is represented as:

$$G_m = \upsilon_g(G_{m-1}, \sigma)$$
$$\sigma = \{\sigma_1, \sigma_2, \sigma_3\}$$

Where G_{m-1} denotes a parent population (genotypes) and G_m denotes a subsequent population (genotypes) produced through the genetic mechanism σ, υ_g is a transformation. The genetic mechanism σ includes selection (σ_1), crossover (σ_2) and mutation (σ_3). The crossover and mutation operations are performed based on a probability rate.

2. Detection process

A population of phenotypes or design forms is produced from the corresponding genotypes. The detection process is then performed to examine semantics of the population (phenotypes) against their suitability to the style environment. The process is as follows.

(1) Producing phenotypes (i.e. design forms) from the genotypes (i.e. a set of executions of syntax rules), namely,

$P_m = \upsilon_p(G_m, K_p)$ - producing phenotypes

where P presents phenotypes, G presents genotypes, m denotes generation of population, K_p presents interpretation knowledge (i.e. mapping rules from G to P) and υ_p is a transformation.

(2) Detecting semantics in phenotypes and measuring them against style environment, and providing feedback regarding the fitness in style environment for each individual, namely,

$F = \upsilon_f(S_{ev})$ - defined fitness of style environment
$S_{im} = \upsilon_{im}(P, K_{im})$ - existing semantics in phenotypes
$\xi = \upsilon_{me}(S_{im}, F)$ - evaluating semantics against fitness

where F is the defined fitness function of a style environment, S_{ev} presents style environment, S_{im} denotes existing semantics in phenotypes, P denotes phenotypes, K_{im} is interpretation knowledge, ξ is an evaluation operator, which examines the semantics in phenotypes against the fitness of a style environment and produces fitness values of individuals and υ_{im}, υ_f and υ_{me} denote transformations.

The fitness of the style environment (F) is constructed as hierarchical fitnesses, which include a set of simple fitnesses (F^0) for simple semantics, a set of complex fitnesses (F^1) for complex semantics, and the fitness of style (F^2) for a set of multiple complex semantics, as represented below:

$F = \{F^0, F^1, F^2\}$

The evaluation process (ξ) proceeds hierarchically, namely,

$\xi = \{\xi^0, \xi^1, \xi^2\}$
$\xi^0 = \{\xi^0_i\}$
$\xi^1 = \{\xi^1_j\}$

Where ξ is the evaluation operator, including ξ^0, ξ^1 and ξ^2 which are respectively the evaluations of simple semantics, complex semantics and a style.

The different simple or complex semantics are respectively evaluated, namely,

$\xi^0_i = \upsilon_{me}(I^i_{im}, f^0_i), f^0_i \quad F^0$
$\xi^1_j = \upsilon_{me}(C^j_{im}, f^1_j), f^1_j \quad F^1$
$\xi^2 = \upsilon_{me}(S_{im}, F^2)$

where ξ^0 denotes the evaluation of different simple semantics, I^i_{im} denotes a simple semantics, i denotes its number, ξ^1_j denotes the evaluation of different complex semantics,

C^{j}_{im} denotes a complex semantics, j denotes its number, and ξ^{2} denotes the evaluation of a style (i.e. a set of multiple complex semantics).

Based on the fitness, a design population is classified, and the discovery process is then carried out.

5.2 DISCOVERY

The discovery process uses genetic engineering to locate the commonality of the genetic structures which produce high fitnesses of semantics suited to the style environment. These genetic structures are then produced as evolved genes.

This process is as follows. The individuals (phenotypes) in the population are classified as good or bad according to their fitness values, namely,

$$P = P^{+} \quad P^{-}$$
$$P^{+} = \upsilon_{c}(P, \xi)$$

where P denotes the phenotypes, P^{+} is the subset of those phenotypes which are highly suited to the style environment, P^{-} is the subset of those phenotypes which are lowly suited to the style environment, ξ is the evaluation operator and υ_{c} is a transformation.

The subsets of genotypes corresponding to the subsets of phenotypes are then obtained, namely,

$$G = G^{+} \quad G^{-}$$
$$G^{+} = \upsilon_{t}(P^{+})$$
$$G^{-} = \upsilon_{t}(P^{-})$$

where G^{+} is the subset of genotypes corresponding to P^{+}, G^{-} is the subset of genotypes corresponding to P^{-} and υ_{t} is a transformation.

The system then searches for gene structures which are common in G^{+} but not in G^{-}, and produces evolved genes that capture those gene structures.

$$\omega = \upsilon_{d}(G^{+}, G^{-})$$
$$U \quad G^{+}$$
$$U \quad G^{-}$$

where ω denotes a discovery operator, υ_{d} is a transformation and U is a set of evolved genes.

These evolved genes are found hierarchically. Firstly, the system sets the simple semantics as the environment and searches for the gene structures that map onto high fitnesses of simple semantics. The gene structures found are put in the pool of evolved genes, and then used to seed complex gene structures. It then shifts the environment to the complex semantics and searches for the gene structures that map onto high fitnesses of complex semantics. The gene structures found are put in the pool of evolved genes, and then used to seed more complex gene structures. Finally, it shifts the environment to a style and searches for the gene structures that map onto high fitnesses of that style. Thus, the system derives a set of the evolved genes that map onto hierarchical semantics of a style, namely,

$$U = \{U^{1}, U^{2}, U^{3}\}$$
$$U^{1} \leftrightarrow \{I\}$$
$$U^{2} \leftrightarrow \{C\}$$
$$U^{3} \leftrightarrow S$$

where U^1 is a set of the evolved genes which map onto a set of simple semantics, U^2 is a set of the evolved genes which map onto a set of complex semantics and U^3 is a set of the evolved genes which map onto a style. 'I' denotes simple semantics, C denotes complex semantics and S denotes a style. The symbol $\leftrightarrow$ means mapping. Examples of gene structures are presented in Figure 10.

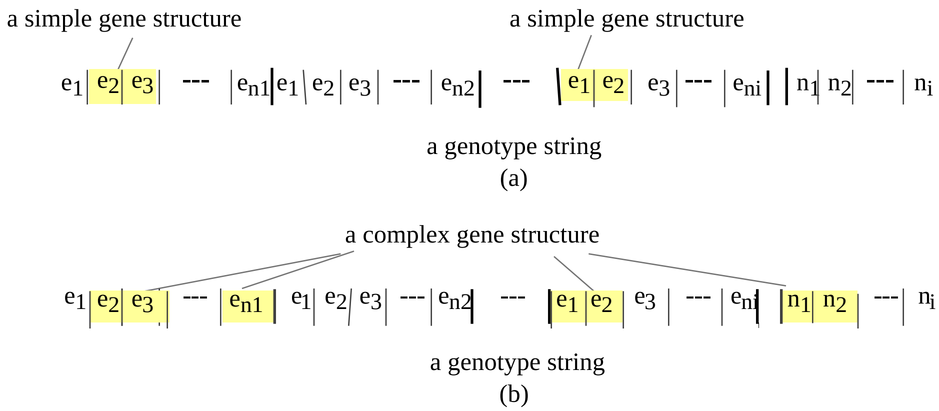


Figure 10. Examples of gene structures: (a) simple gene structures where the genes are contiguous, and (b) a complex gene structure where the genes are non-contiguous.

The set of evolved genes contains evolved genes of different complexity. The higher complex evolved genes consist of lower complexity evolved genes and initial genes, namely,

$$g^l \quad U$$
$$g^l = B(g^0, g^{l-1}, g^{l-2}, ---, g^{l-b})$$

where g^l denotes an evolved gene, l denotes complexity, g^0 denotes the initial gene and B denotes a gene structure. Evolved genes become more and more complex as the evolution continues.

An example of the complexity of evolved genes is presented in Figure 11. Here the evolved gene g_{13} consists of lower complexity evolved genes g_{11} and g_{12}. A tree structure is used to represent the derivation path of the evolved gene. Complexity corresponds to structural level.

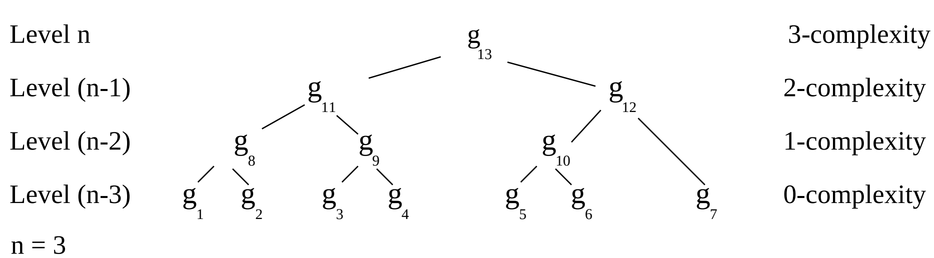


Figure 11. An example of the complexity of an evolved gene.

5.3 EVOLUTIONARY COMBINATION

The system modifies the genotype space to produce subsequent populations through an evolutionary combination, namely, the genetic structures of genotype space are improved through continuous combination of evolved genes and initial genes. This enables the system to derive the representations which are increasingly suited to the style environment.

The variability of genotype space with time is represented as follows.

$$G_{t1} = \{H(g^0)\}$$
$$G_{t2} = \{H(g^0, U^1)\}$$
$$G_{t3} = \{H(g^0, U^1, U^2)\}$$
$$G_{t4} = \{H(g^0, U^1, U^2, U^3)\}$$

where G_t denotes genotype space (ie. a population of genotypes), t denotes time and $t4 > t3 > t2 > t1$. $H(g^0)$ is a gene string (ie. a genotype), g^0 denotes the initial genes, U^1 denotes the evolved genes linked to high fitness of simple semantics, U^2 denotes the evolved genes linked to high fitness of complex semantics and U^3 denotes the evolved genes linked to high fitness of a style.

6. Application Example

We demonstrate the application of this model by presenting the emergence of the representation of a traditional Chinese architectural facade style.

6.1 REPRESENTATION OF STYLE SPACE

1. Syntax

We code the form elements of traditional architectural facade as initial genes and describe them using alphabets. The form elements include roofs, walls, doors, columns, windows, podiums and so on. An example of the coding of form elements is presented in Figure 12.

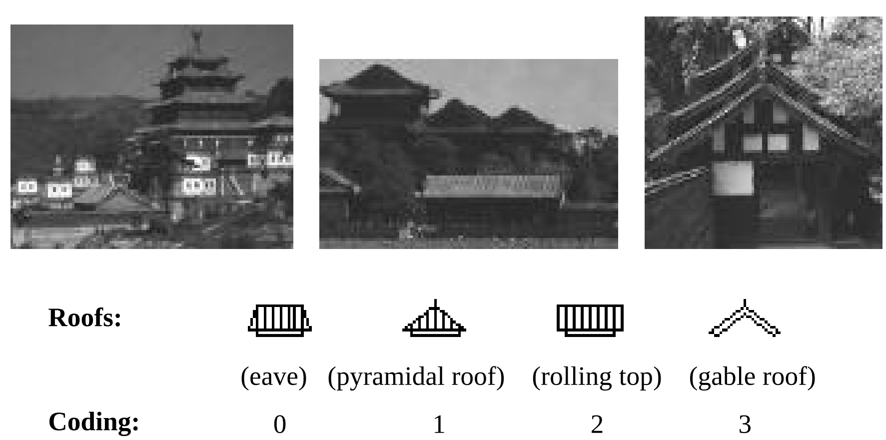


Figure 12. Examples of coding form elements as initial genes.

We construct a gene schema to represent syntax space. The gene schema (S) consists of a plan gene schema (S_{plan}) and a rule gene schema (S_{rule}) that code the hierarchies of executions of syntax rules for a design, ie:

$$S = \{S_{rule}; S_{plan}\}$$

The plan gene schema maps onto the planning of the whole facade. A facade is divided into M blocks. Each block involves n_i form elements and every block is labelled using the number of the form elements in the block. A plan gene schema is a sequence of these:

$S_{plan} = [n_1, n_2, n_3, \quad , n_j]$

where n_i denotes a block of the facade and corresponds to the number of form elements in the block, i (1, j). The phenotype produced by this is shown in Figure 13(a).

The rule gene schema maps onto the placement of form elements in each block of the facade. Form elements are placed from bottom to top. A rule gene schema is a sequence of placements of form elements. It is represented as:

$S_{rule} = [n_1E_1, n_2E_2, n_3E_3, \quad , n_jE_j]$
$E_i = [e_1, e_2, e_3, \quad , e_{ni}]$

where E_i denotes a set of the form elements placed in block n_i, n_i is labelled using the number of form elements in the block and e_{ni} is a form element. The phenotype produced by it is illustrated in Figure 13(b).

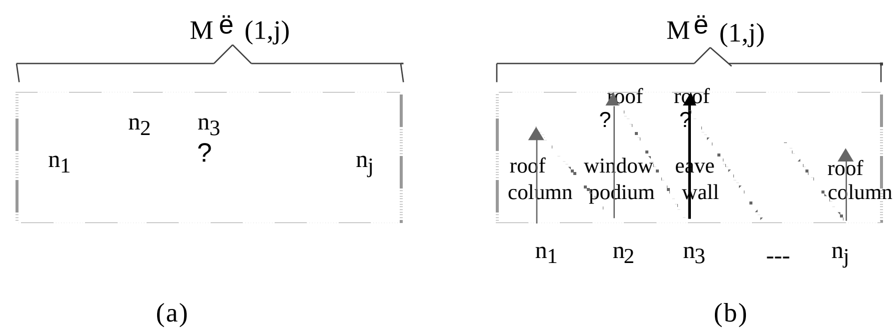


Figure 13. (a) Phenotype produced by the plan gene schema; (b) phenotype produced by the rule gene schema.

Based on the gene schema, the system randomly seeds a set of syntax rules, ie. genotypes. It seeds the plan genes first, then seeds the rule genes under the plan to produce an individual. The template of the genotype of an individual is illustrated in Figure 14.

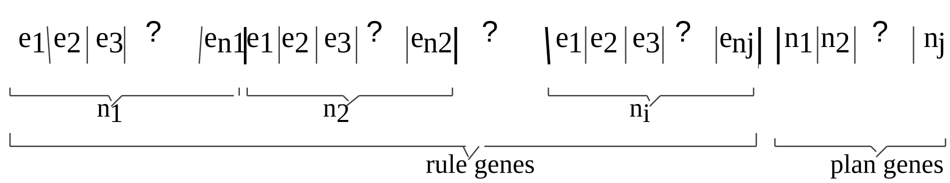


Figure 14. The template of a genotype of an individual in terms of the gene schema.

The genotype strings of a population have variable lengths. They vary dynamically whenever a new population is produced.

2. Form

The system executes the syntax rules in a genotype to produce a phenotype as a form, which is a structural description of facade design. A phenotype is represented using a grid as below.

$$P = \{[E^0_{x,y}, (E^1_{x,y}, E^1_{x+a,y+b}, \quad)]\}$$

where P denotes a phenotype and $\{[E^0_{x,y}, (E^1_{x,y}, E^1_{x+a,y+b}, \quad)]\}$ is a set of the position descriptions of form elements. $E^0_{x,y}$ denotes a form element, whose central point is at the grid point (x, y). $(E^1_{x,y}, E^1_{x+a,y+b}, \quad) \equiv 1$ is a set of the grid points covered by this element, $E^1_{x,y} \quad (0, 1)$. The phenotypes are produced from corresponding genotypes using mapping rules.

We introduce an example of decoding genotype into phenotype. Figure 15 presents a genotype consisting of a plan gene string and a rule gene string. The system translates this genotype into a symbolic description as in Figure 16. It then executes the plans and rules to produce a description relating to the positions and attributes of form elements, ie. a phenotype, Figure 17. A resulting phenotype is represented in Figure 18.

11 | 0 \ 1 | 11 | 2 | 11 | 2 | 11 0 1 11 | 2 11 | 2 11 | 0 || 1 | 3 | 22 | 3 | 2 | 2 | 3

rule gene string plan gene string

an individual genotype string

Figure 15. A genotype consisting of plan gene string and rule gene string.

$$\{[(\bar{\ }e_{column\_1})(\bar{\ }e_{eave})(\bar{\ }e_{pyramidal})][(\searrow e_{column\_1})(\bar{\ }e_{rolling\_top})][(\searrow e_{column\_1})(\bar{\ }e_{rolling\_top})]$$
$$[(\searrow e_{column\_1})(\bar{\ }e_{eave})(\bar{\ }e_{pyramidal})][(\searrow e_{column\_1})(\bar{\ }e_{rolling\_top})][(\searrow e_{column\_1})(\bar{\ }e_{rolling\_top})$$
$$[(\searrow e_{column\_1})(\bar{\ }e_{eave})(\bar{\ }e_{pyramidal})]; [(B(3))(B(2))(B(2))B(3))(B(2))(B(2))(B(3))\}$$

Where, (δe_i) are symbolic descriptions of rule genes, $\delta \quad (\downarrow, \uparrow, \searrow)$ denotes the action applied to form element, $e_i \quad$ (column_1, eave, pyramidal roof, rolling_top) denotes a form element and B(j) is the symbolic description of a plan gene, j (2,3).

Figure 16. Translation of a genotype into symbolic description (corresponding to Figure 15).

$$\{[E^0_{0,0} = e_{column\_1}, E^1_{0,0} = 1], [E^0_{0,1} = e_{eave}, E^1_{0,1} = 1], [E^0_{0,2} = e_{pyramidal}, E^1_{0,2} = 1],$$
$$[E^0_{1,0} = e_{column\_1}, E^1_{1,0} = 1], [E^0_{1,1} = e_{rolling\_top}, E^1_{1,1} = 1], [E^0_{2,0} = e_{column\_1}, E^1_{2,0} = 1],$$
$$[E^0_{2,1} = e_{rolling\_top}, E^1_{2,1} = 1], [E^0_{3,0} = e_{column\_1}, E^1_{3,0} = 1], [E^0_{3,1} = e_{eave}, E^1_{3,1} = 1],$$
$$[E^0_{3,2} = e_{pyramidal}, E^1_{3,2} = 1], [E^0_{4,0} = e_{column\_1}, E^1_{4,0} = 1], [E^0_{4,1} = e_{rolling\_top}, E^1_{4,1} = 1],$$
$$[E^0_{5,0} = e_{column\_1}, E^1_{5,0} = 1], [E^0_{5,1} = e_{rolling\_top}, E^1_{5,1} = 1], [E^0_{6,0} = e_{column\_1}, E^1_{6,0} = 1],$$
$$[E^0_{6,1} = e_{eave}, E^1_{6,1} = 1], [E^0_{6,2} = e_{pyramidal}, E^1_{6,2} = 1]\}$$

Figure 17. Translation of symbolic descriptions of a genotype into the descriptions relating to the positions and attributes of form elements, ie. a phenotype (corresponding to Figures 15 and 16).

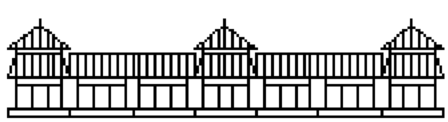


Figure 18. A resulting phenotype corresponding to Figure 17.

3. Semantics

A set of concepts of traditional facade style is specified in the system, which encompass simple semantics, complex semantics and style. They are modelled into hierarchical fitnesses used to examine the populations (phenotypes) and guide convergence of style of the populations.

Simple semantics of traditional facade is specified using phenotypic properties. For example, simple semantics ‘vertical eaves’ is specified as below:

$$\text{Vertical eaves} \Leftrightarrow \{(e_{eave1}, e_{eave2});\ e_{eave1} = e_{eave2},\ x_{eave1} = x_{eave2}\}$$

where, e_{eave} denotes a form element ‘eave’, x_{eave} denotes its position in x axis direction of a grid.

The fitness value of simple semantics is either ‘1’ or ‘0’, based on whether it exists or not in the phenotype.

Complex semantics of traditional facade is specified using a set of simple semantics. For example, complex semantics ‘horizontal roofs along an axis' is given as:

$$\text{Horizontal\_axis (roofs)} \Leftrightarrow \{[\text{Horizon}(e_{roof1}, e_{roof2}), \text{Horizon}(e_{roof3}, e_{roof4}), \text{Horizon}(e_{roof5}, e_{roof6}), \quad];\ y_{roof1} = y_{roof2} = y_{roof3} = y_{roof4} = y_{roof5} = y_{roof6} = \quad \}$$

where, e_{roof} denotes a form element ‘roof’, [Horizon (e_{roof1}, e_{roof2}), Horizon (e_{roof3}, e_{roof4}), Horizon (e_{roof5}, e_{roof6}), ---] represents a set of simple semantics regarding horizontal relationships, y_{roof} denotes the position of a roof element in y axis direction of a grid. Complex semantics ‘mirror symmetry’ is given as:

$$\text{Mirror\_symmetry} = \{[\text{Repetition}(e_1, e_2), \text{Repetition}(e_3, e_4), \quad , \text{Repetition}(e_i, e_j)];\ e_1 = \tau_0(e_2, X_0), e_3 = \tau_1(e_4, X_1), \quad , e_i = \tau_n(e_j, X_n), X_0 \equiv X_1 \equiv \quad \equiv X_n\}$$

where [Repetition (e_1, e_2), Repetition (e_3, e_4), ---, Repetition (e_i, e_j)] represents a set of simple semantics regarding repetition relationships, X_n denotes a symmetry axis between the elements e_i and e_j, τ_n is a transformation and $X_0 \equiv X_1 \equiv$ --- $\equiv X_n$ represents that these Repetition relationships have a common symmetry axis,

The fitness of complex semantics is defined to be the best fit to all the relationship properties that constitute this complex semantics.

A set of style concepts of traditional facade is also specified. A style is the set of common complex semantics. Therefore, a style concept is specified as:

Style: {horizontal roofs along an axis, mirror symmetry of facade}

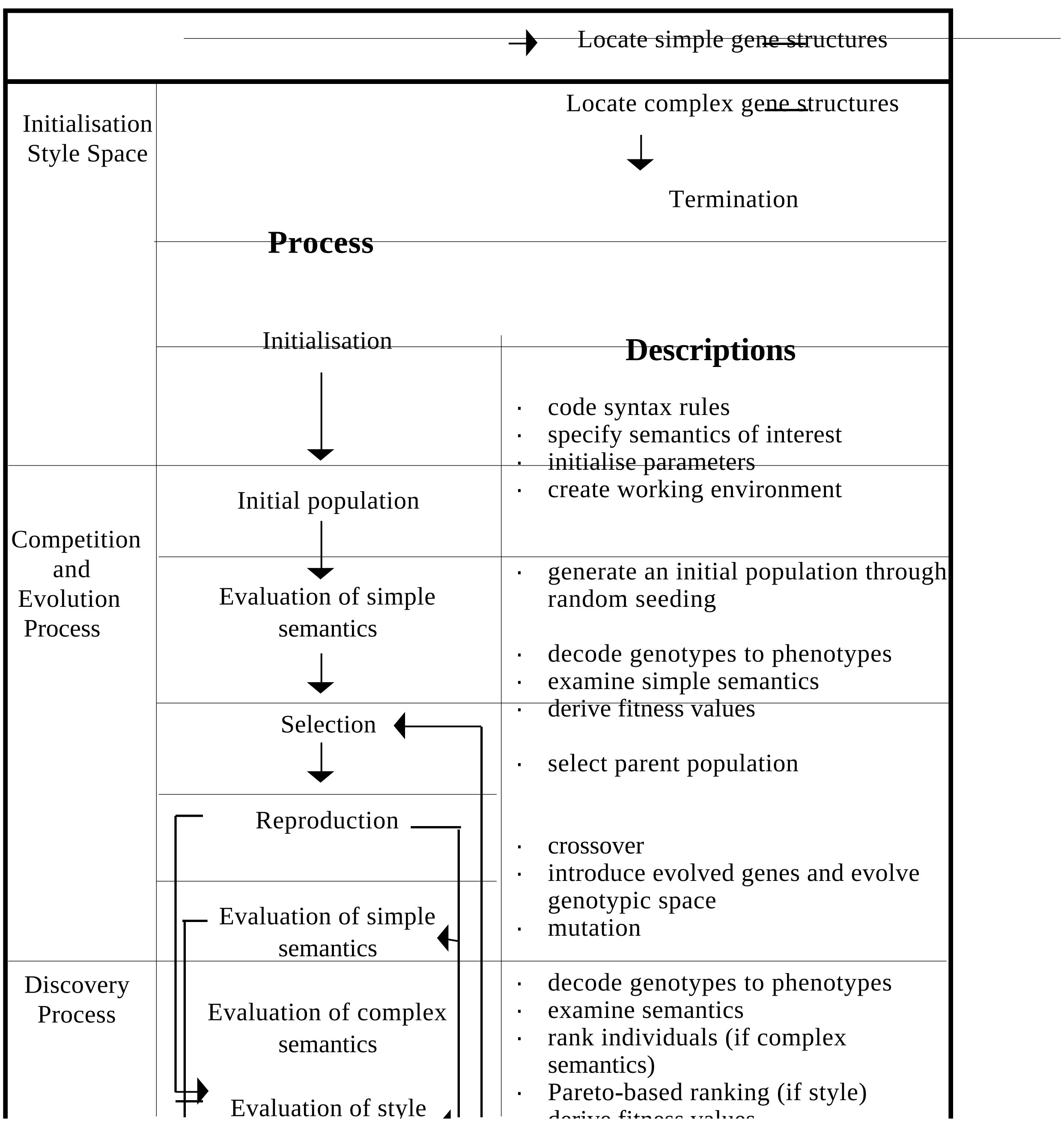
Fitness of a style is the fitness of a set of the multiple complex semantics. Fitness evaluation of a style is performed using a Pareto-based method.

6.2 THE BASIC STRUCTURE AND SEQUENCE OF EVOLUTIONARY PROCESS

The basic structure and sequence of the evolutionary process for style representation emergence is as follows and is summarised in Table 1.

1. Initialise style space. This includes coding of syntax rules, specification of semantics of style of interest, initialisation of parameters, and creation of working environment.
2. Randomly seed a set of syntax rules to produce in initial population (genotypes).

3. Decode the genotypes into phenotypes for evaluation of the fitness of simple semantics.
4 Classify phenotypes into subset P^+ and P^- according to their fitness values. The subsets of genotypes G^+ and G^- corresponding to them are then located.
5. Compare the gene pairs in genotypes and search for the sequences that are common in G^+ but not in G^- in order to produce evolved genes. These evolved genes are put in the pool of evolved genes for the production of subsequent populations.



	· locate gene structures and produce evolved genes
· set out sub-population	· store · report · redescribe genotypes

Table 1. Process sequence.

6. Shift to another simple semantics and repeat the above process until the evolved genes mapping onto all of the simple semantics are obtained.
7. Combine the initial genes and evolved genes to produce the subsequent population, evaluate this population against the fitness of complex semantics and classify it.
8. Compare the interactions between the evolved genes and initial genes in the genotypes and search for their sequences that are common in G^+ but not in G^-, and evolve them.
9. Shift to another complex semantics and repeat the above process until the evolved genes mapping onto all of the complex semantics are obtained.
10. Combine the initial genes and evolved genes to produce the subsequent population, evaluate this population against the fitness of a style (ie. fitness of a set of multiple complex semantics), and classify it.
11. Compare the complex interactions between the evolved genes and initial genes in genotypes and search for their sequences which are common in G^+ but not in G^-, and evolve them.
12. A set of evolved genes and their relationships representing the specific style is obtained.

The evolutionary process is terminated when a fixed number of generations are completed or when the convergence of semantics of resulting designs is not improved over a fixed number of generations.

6.3 THE EMERGENCE OF THE REPRESENTATION OF TRADITIONAL FACADE STYLE

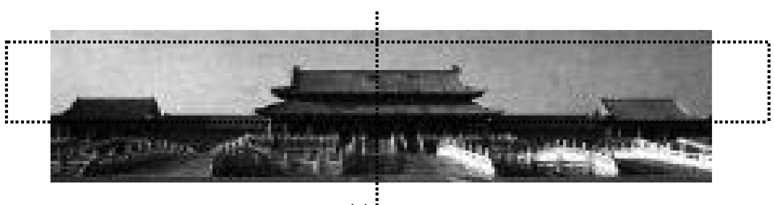
We present some results from an implemented system. The population size and the maximum number of generations are respectively set to 100. The simple semantics required includes: eave is above column, repeated eaves, pyramidal roof is above eave and so on, Table 2. The complex semantics required includes: horizontal roofs along an axis, mirror symmetry of entire facade. A style is the set of common complex semantics: {horizontal roofs along an axis, mirror symmetry of entire facade}. Figure 19 presents an example of a traditional facade style consisting of the common complex semantics ‘horizontal roofs along an axis’ and ‘mirror symmetry of entire facade’.

1. Emerging representations of simple semantics

The form elements of traditional architectural facade are coded as initial rule genes and the blocks of facade are coded as initial plan genes. In the system, there are 23 initial rule genes (described as 0 – 22) and 10 initial plan genes (described as (0-9). Rule gene numbers above 22 and plan gene numbers above 9 are evolved genes.

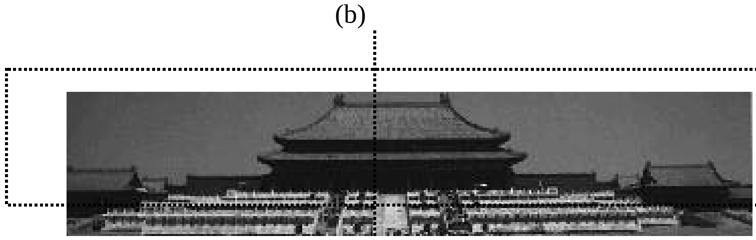
Simple Semantics	
Pyramidal roof is above eave	Two horizontal roofs
Pyramidal roof is above columns	Vertical eaves
Eave is above columns	Vertical columns
Eave is above window	Repeated windows
Gable roof is above wall	Repeated walls
Rolling top is above wall	Repeated eaves
Rolling top is above window	Repeated gable roofs
Rolling top is above columns	Repeated columns
Rolling top is above eave	Pyramidal roofs touching each other at a corner
Podium is on the ground	Eaves touching each other at a Window corner
Columns rest on top of podium is above podium	Two similar pyramidal roofs
Two similar rolling tops	Parallel pyramidal roof and rolling top
Eave is above wall	

Table 2. A set of simple semantics.



(a)

(b)



(c)

Figure 19. An example of a traditional facade style consisting of the common complex semantics ‘horizontal roofs along an axis’ and ‘mirror symmetry’.

The system initially derives different simple semantics and evaluates them in the context of finding representations of simple semantics of traditional Chinese architectural facade. Examples of these evolved genes (syntax rules) are shown in Figures 20.

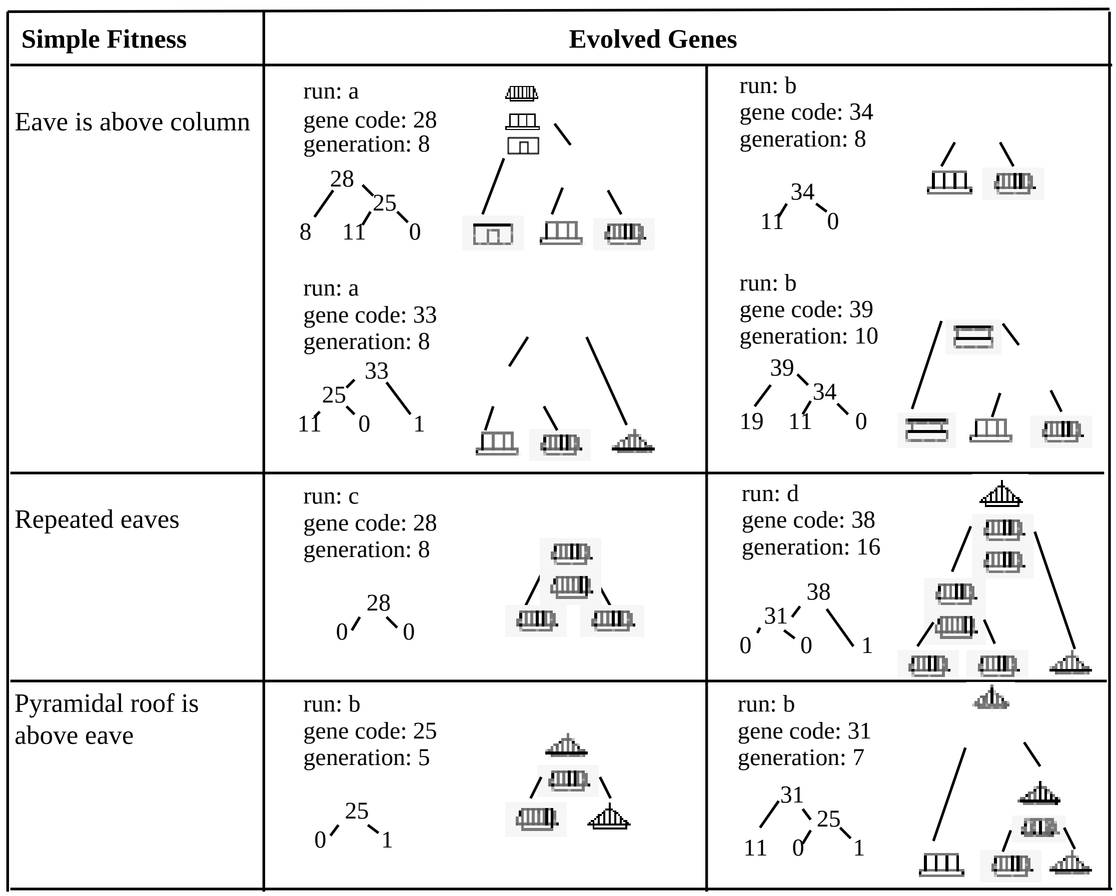


Figure 20. Examples of the evolved genes mapping onto some simple semantics of traditional architectural facade derived by the system; a, b, c and d denote different runs.

We see that a number of identical evolved genes for the same simple semantics are derived from different runs. For example, for the same simple semantics 'eave is above column', the evolved gene 25 derived from Run a and the evolved gene 34 derived from Run b are identical. This reflects the regularities of syntax decisions regarding a simple semantics.

2. *Emerging representations of complex semantics*

The system then derives different complex semantics and evaluates them in the context of finding the representations of complex semantics of traditional architectural facade. Examples of the individuals with high fitnesses of the complex semantics ‘horizontal roofs along an axis’ are shown in Figures 21. Some evolved genes mapping onto this complex semantics are represented in Figure 22.

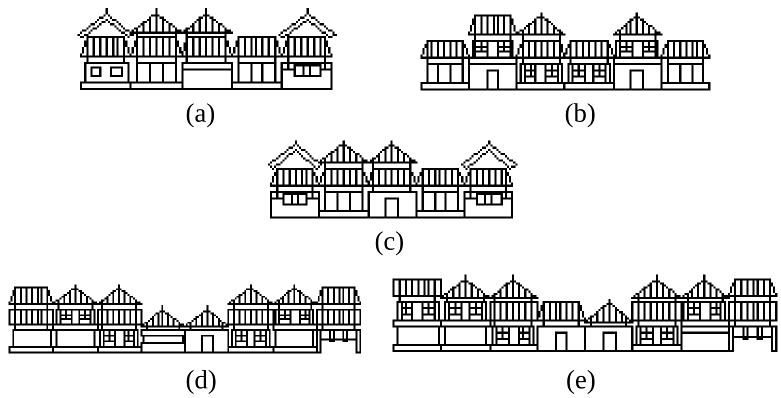
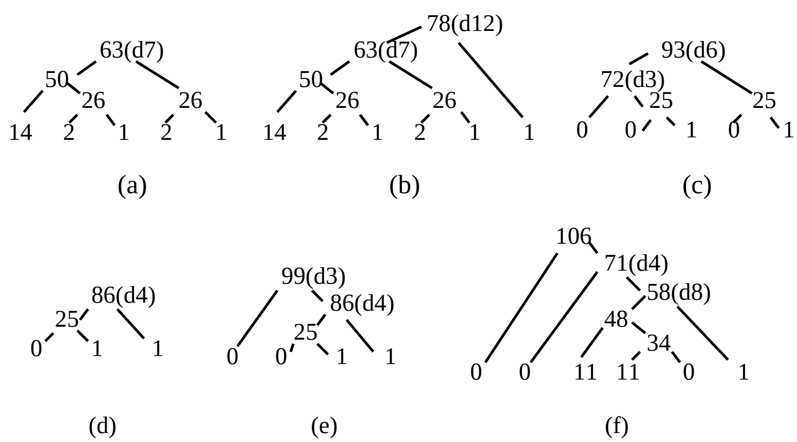


Figure 21. Examples of some individuals with high fitness of the complex semantics ‘horizontal roofs along an axis’ produced by the system.



Where dj denotes the position interval j of two genes which compose a higher level of evolved gene in the genotype.

Figure 22. Some evolved rules genes mapping onto the complex semantics ‘horizontal roofs along an axis’ derived from different runs.

In Figure 22, the codes 25, 26 and 34 are the evolved genes for simple semantics of relationships of roofs. We see that most evolved genes for this complex semantics are a combination of those evolved genes mapping onto simple semantics of roof placement. For example, the evolved gene 93 consists of the evolved rule genes of placing pyramidal roofs and eaves which map onto simple semantics. We also see that the evolved genes of low complexity can interact with other genes to form the evolved genes of high complexity. For example, two evolved genes 26 interact with each other to form the evolved gene 63, and the evolved gene 63 interacts with the gene of the roof again to form the evolved gene 78; so does the evolved gene 99.

Figure 23 shows some individuals with high fitnesses of the complex semantics ‘mirror symmetry of entire facade’. Some evolved plan and rule genes that map onto this complex semantics are represented in Figure 24. We see that most evolved plan genes that map onto this complex semantics have the same or reversed execution sequence.

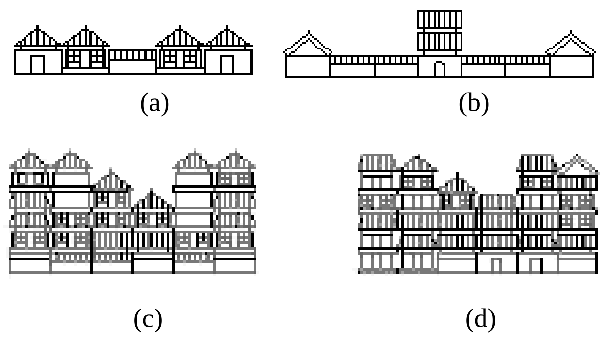


Figure 23. Examples of some individuals with high fitness of the complex semantics ‘mirror symmetry of entire facade’ produced by the system.

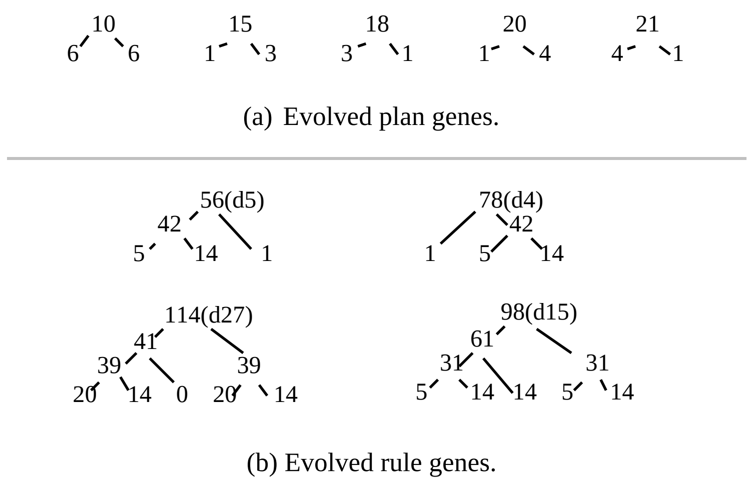


Figure 24. Some evolved plan and rule genes mapping onto the complex semantics ‘mirror symmetry of entire facade’ derived from different runs.

The system then uses these evolved genes to improve the representations of complex semantics for a predefined number of generations. The pool of evolved genes is updated.

3. Emerging representations of a style

Finally, the system emerges the representations of style of the populations in the context of finding the representations of a traditional Chinese architectural facade style. Figure 25 shows examples of individuals with high fitnesses of this style. Figure 26 presents some evolved genes for this style derived from various runs.

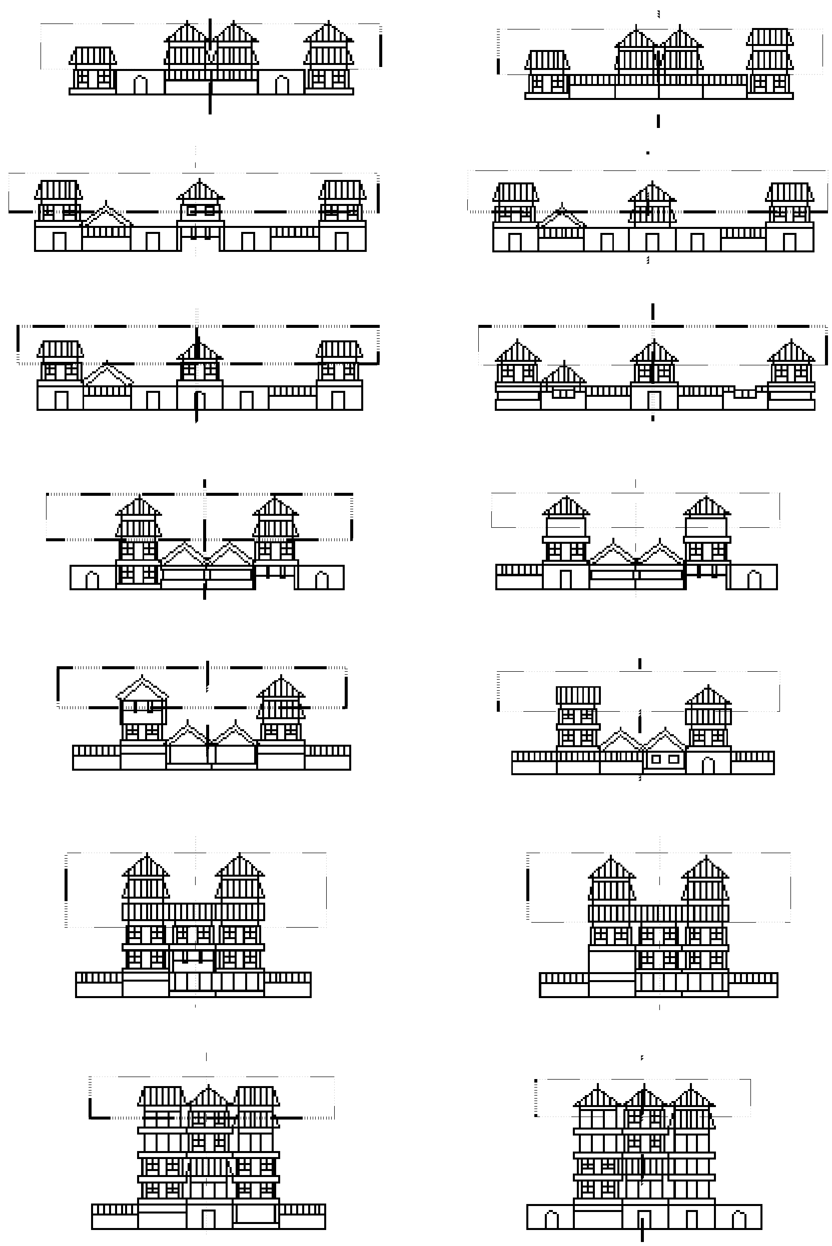


Figure 25. Examples of some individuals with high fitness of a style ‘horizontal roofs along an axis, symmetry of facade’ produced by the system.

We see that the evolved genes of low complexity that map onto simple and complex semantics are combined to form evolved genes of higher complexity that map onto a style. For example, in Figure 26, the evolved gene 126 consists of the evolved gene 99 which maps onto complex semantics and the evolved gene 25 which maps onto simple semantics and other genes.

We also see that most evolved genes mapping onto a style have similar structural relationships such as repetitions of evolved genes, reversed execution sequence of evolved genes and so on. For example, in Figure 26, the evolved genes 111, 116 and 371 have a similar structural relationship which is the repetitions of evolved genes.

4. Examination of emergent representations of style

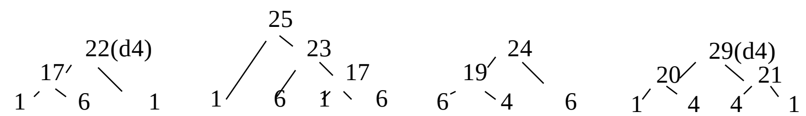
We examine: (1) whether semantics and style of the populations produced using these evolved genes are highly suited to the style environment required; and (2) whether there are new evolved genes emerging for the style required. Results from 50 typical runs are presented.

The system uses the set of evolved genes that has been derived for a style as well as the initial genes to produce design populations. The best fitness of populations is reached at the first generation, Figure 27. This means that the populations produced using these evovled genes are highly suited to the environment of simple semantics.

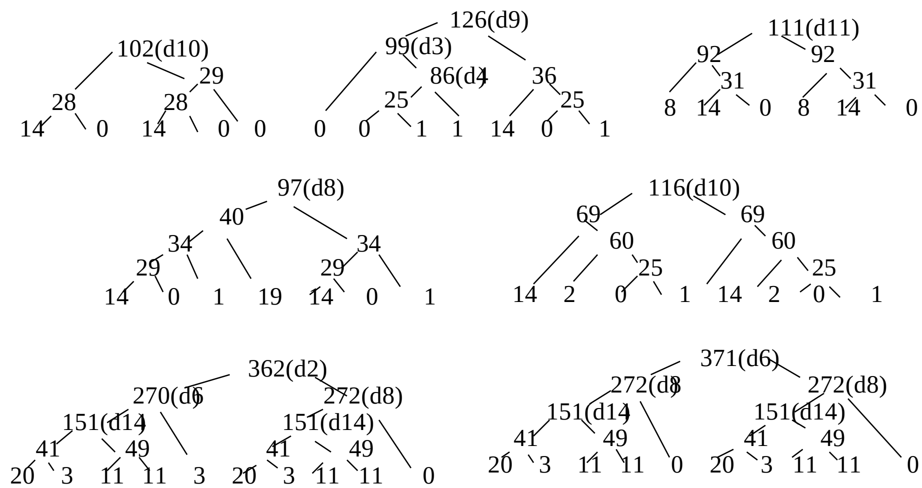
We use the set of evolved genes to produce populations and examine the fitness of complex semantics. Figure 28 shows the average of the best fitnesses of the populations versus generation number. We see that fitnesses of populations arrive at the higher fitness value quickly and then remain steady. This means that the populations produced using these evolved genes are highly suited to the environment of complex semantics.

We use these evolved genes to produce populations and examines fitnesses of a style. The average of the best fitnesses of the populations versus generation number is shown in Figure 29. We see that the fitness of populations arrives at higher values very early and then increases only slightly. This means that the populations produced using these evolved genes are highly suited to the style environment.

We run the emergence process again to see if further evolved genes are produced. Results from 30 typical runs for 3 different styles are shown in Figure 30. We see that very few new evolved genes are derived. This means that the existing set of evovled genes for a style has covered the representations of the style well. This forms the basis of testing examples to determine whether they exhibit a particular style.



(a) Evolved plan genes.



(b) Evolved rule genes.

Figure 26. Some evolved plan and rule genes mapping onto the style ‘horizontal roofs along an axis, symmetry of facade’ derived from different runs.

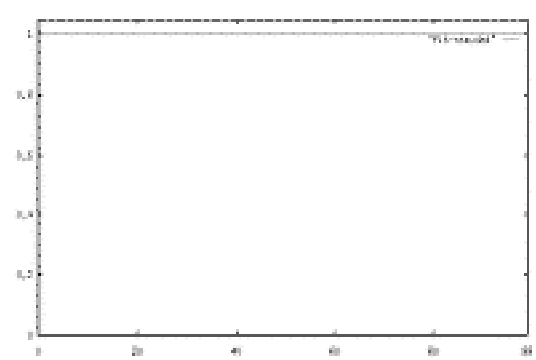
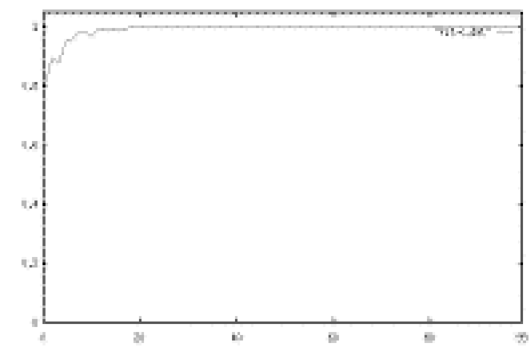


Figure 27. The average of the best fitness of populations versus generation number for simple semantics.

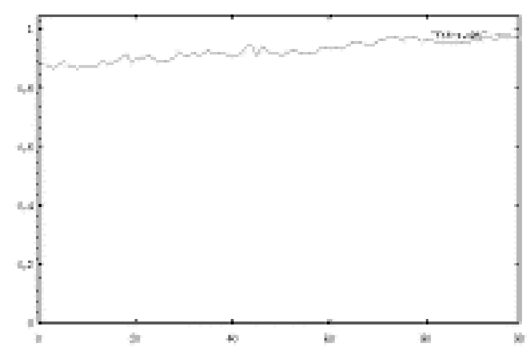
Best fitness (complex semantics)



Generations

Figure 28. The average of the best fitness of populations versus generation number for complex semantics.

Best fitness (a style)



Generations

Figure 29. The average of the best fitness of populations versus generation number for a style.

Style	Average Number of Existing Evolved Genes for a Style	Average Percentage of New Evolved Genes
Style 1 (10 runs)	193	0.5 %
Style 2 (10 runs)	173.3	1.0 %
Style 3 (10 runs)	164.6	0.7 %

Figure 30. The average percentage of new evolved genes derived for 3 different styles.

7. Conclusions

This paper presents an evolutionary process model for style representation emergence in design as well as an application of an implementation to traditional Chinese architectural facades. The mapping between the meanings and the explicit representation of style is interpreted as semantics and syntax, which can be derived through this hierarchical evolutionary process. The development demonstrates that it is possible to make a style representation explicit automatically.

Once a style is represented other individuals can be tested against it to determine whether they fit into that style. In addition to being able to be applied to any corpus of work to determine its style representation, the approach lends itself to the determination of style shift. Style shift is the change of style over time. New designs with a similar style can be generated by reusing the representation of that style. Further, with the explicit representation of style it becomes possible to ascertain how close two styles are by measuring the distance between their representations.

This approach provides design researchers with an aide to automatically represent, recognize and learn design styles and it may be used to explore styles through perturbations of the representation.

Acknowledgments
This work is supported by an Australian Postgraduate Award and by an Australian Research Council grant (No. A89700249).

References

Ackerman, J.S.: 1967, A theory of style, *in* Beardsley, M.C. and Schueller H.M. (eds), *Aesthetic Inquiry: Essays on Art Criticism and the Philosophy of Art,* Dickenson, Belmont, CA, pp. 54-66.

Bentley, P. (ed.): 1999, *Evolutionary Design by Computer,* Morgan Kaufmann, San Francisco.

Chan, C.: 1992, *Imperial China,* Chronicle Books, San Francisco.

Chan, C.S.: 1995, A cognitive theory of style, *Environment and Planning B: Planning and Design* **22**: 461-474.

Chinese Academy of Architecture: 1986, *Classical Chinese Architecture,* Joint, HongKong.

Chiou, S-C and Krishnamurti, R: 1995, The grammar of Taiwanese traditional vernacular dwellings, *Environment and Plan B: Planning and Design* **22**: 689-720.

Chiou, S-C and Krishnamurti, R: 1996, Example Taiwanese traditional house, *Environment and Plan B: Planning and Design* **23**: 191-261.

Coyne, R. D. and Gero, J. S.: 1986, Semantics and the organization of knowledge in design, *Design Computing* **1**: 68-88.

Ding, L. and Gero, J.S.: 1998, Emerging representations of Chinese traditional architectural style using genetic engineering, *in* Huang, X., Yang, S. and Wu, H. (eds), *The International Conference on Artificial Intelligence for Engineering,* HUST Press, Wuhan, China, pp. 493-498.

Edwards, A.T.: 1945, *Style and Composition in Architecture,* John Tiranti, London.

Gero, J.S.: 1992, Creativity, emergence and evolution in design, *in* J. S. Gero and F. Sudweeks (eds), *Preprints of Computational Models of Creative Design,* Department of Architectural and Design Science, University of Sydney, Sydney, pp. 1-28.

Gero, J.S., Damski, J. C. and Jun, H. J.: 1995, Emergence in CAAD systems, *in* M. and R. Teh (eds), *The Global Design Studio,* National University of Singapore, Singapore, pp. 423-438.
Gero, J.S. and Ding, L.: 1997, Learning emergent style using an evolutionary approach, *in* Verma, B. and Yao, X. (eds), *ICCIMA'97*, Griffith University, Gold Coast, Queensland, Australia, pp.171-175.
Gero, J.S. and Ding, L.: 1997, Exploring style emergence in architectural design, *in* Liu, Y-T., Tsou, J-H. and Hou, J-H (eds), *CAADRIA'97*, Hu's Publisher, Taipei, Taiwan, pp.287-296.
Gero, J.S. and Kazakov, V.A.: 1996, Evolving building blocks for design using genetic engineering: a formal approach, *in* Gero, J. S. (ed.), *Advances in Formal Design Methods for CAD*, Chapman and Hall, London, pp.31-50.
Gero, J. S. and Kazakov, V.A. (2001) A genetic engineering extension to genetic algorithms, *Evolutionary Systems* (to appear)
Gero, J.S. and Schnier, T.: 1995, Evolving representations of design cases and their use in creative design, in Gero, J.S., Maher, M.L. and Sudweeks, F. (eds), *Preprints of the Third International Round-Table Conference on Computational Model of Creative Design*, Department of Architectural and Design Science, University of Sydney, Sydney, pp. 343-368.
Goldberg, D.E.: 1989, *Genetic Algorithms in Search, Optimization and Machine Learning.* Addison-Wesley, Reading.
Hoffmann, D.: 1986, *Frank Lloyd Wright: Architecture and Nature,* Dover, New York.
Holland, J.: 1975, *Adaptation in Natural and Artificial Systems,* University of Michigan Press, Ann Arbor, Michigan.
Knight, W.T.: 1995, Constructive symmetry, *Environment and Planning B: Planning and Design* **22**: 419-450.
Liu, L.G.: 1986, *Chinese Architecture*, Rizzoli, New York.
Liu, Y.T.: 1996, Restructing shapes in terms of emergent subshapes: a computational and cognitive model, *Environment and Planning B: Planning and Design* **23**: 313-328.
Lip, E.: 1995, *Feng Shui*, Academy Editions, Great Britain.
Maher, M.L., Gero, J.S. and Saad, M: 1993, Synchronous support and emergence in collaborative CAAD, in U. Flemming and S. Van Wyk (eds), *CAAD Futures '93,* North-Holland, Amsterdam, pp. 455-470.
Mitchell, W.: 1989, Creativity, emergence, and evolution in design, in J. S. Gero (ed.), *Preprints of Modelling Creativity and Knowledge-Based Creative Design,* Design Computing Unit, University of Sydney, Australia, pp.263-285.
Schapiro, M.: 1961, Style, *in* Philipson, M. and Gudel, P. J. (eds), *Aesthetics Today,* New American Library, New York, pp.137-171.
Simon, H.A.: 1975, Style in design, *in* C. Eastman (ed.), *Spatial Synthesis in Computer-Aided Building Design,* Applied Science, London, pp.287-309.
Stiny, G. and Mitchell, W.J.: 1978, The Palladian grammar, *Environment and Planning B: Planning and Design* **5**: 5-18.
Vincent Scully, Jr.: 1985, *Frank Lloyd Wright,* George Braziller, Inc., New York.

Appendix : Definitions

Architectural semantics – is the representation of architectural meaning. One class of architectural semantics is derived from architectural forms and treated as the description of the forms' relationships. The quantitative and qualitative relationships of forms are considered in this paper.

Architectural style – is defined as a set of common complex semantics (architectural semantics) in a group of designs. It results from a set of common syntaxes and their controls. We formulate an architectural style using the hierarchical syntaxes that determine the style:

Complex semantics – consists of a set of simple semantics and is determined by syntax and its control.

Control of syntax – represents the relationships between different syntax rules or executions of syntax rules. It can be described by the execution order of syntax rules.

Design syntax – is composed of design rules that are used to transform design states. We represent a syntax rule using a set of form element and operators applied to the element.

Explicit representation – is a computational representation consisting of the procedures used in producing objects, for example, a set of the executions of syntax rules. In this paper, it is represented using a genetic description, namely, a set of evolved genes.

Form – is a structural representation of design (see phenotype). It is generated by a set of syntaxes.

Gene – is smallest unit of a genotype.

Genetic codes – are numerals or alphabet letters used for coding in a genotype.

Genetic description – is a description employing genetic terms. The genetic operations used in evolutionary systems can than be carried out on this genetic description.

Genetic structure – A set of genes with a certain order or relationship.

Genotype – the genetic constitution of a design, rather than its physical appearance (see phenotype).

Phenotype – the observable properties of a design, its form.

Semantics – Semantics is the representation of the meaning of the design product. The meaning of the design product can be classified as a perception grounded meaning or as a social-relations grounded meaning. This paper uses the perception grounded meaning.

Simple semantics – Simple semantics is the interpretation of syntax.

This is a copy of the paper: Ding, L. and Gero, J. S. (2001) The emergence of the representation of style in design, *Environment and Planning B: Planning and Design* (to appear)