A 3D Modeling System for Creative Design

[1]Hiroaki NISHINO*   [2]Hideyuki TAKAGI   [3]Sung-Bae CHO   [1]Kouichi UTSUMIYA

[1]*Oita University, 700 Dannoharu, Oita 870-1192, Japan, * email: hn@csis.oita-u.ac.jp*
[2]*Kyushu Institute of Design, 4-9-1 Shiobaru Minami-ku, Fukuoka 815-8540, Japan*
[3]*Yonsei University, 134 Shinchon-dong, Sudaemoon-ku, Seoul 120-749, Korea*

Abstract

We propose a new approach to easily creating 3D geometric models. A technique called interactive evolutionary computation (IEC) is introduced to accelerate user's invention and inspiration of new shapes. The proposed IEC-based design system generates aesthetically pleasing shapes through the simulation of natural evolutionary processes. The user only is required to specify his/her subjective preference of each shape generated by the IEC. The system allows even beginners with little knowledge and experiences of the 3D modeling to acquire innovative shapes. Because the system is developed based on a 3D geometric modeler using implicit surface method, it also provides skilled experts with an advanced modeling interface. The experts can directly modify the internal parameters of the 3D models to make them more elaborate ones. Some experimental results are presented to show the potential of the proposed modeling method.

1. Introduction

As virtual reality (VR) and network communication technologies have dramatically advanced over the last decade, they have been put into practice in many application areas such as product design, manufacturing, education, entertainment, and art. Realistic computer graphics and real-time animation are now accessible technologies for everyone by using standard personal computers and the Internet. There are various software technologies available to share and exploit 3D data on the network such as VRML and Java.

The digital prototyping, as one of the most prospective applications, allows designers to seamlessly design, prototype, and evaluate new products in a networked virtual environment, which significantly increases efficiency and timeliness of the product development. Because the recent development activities give weight to products' sense and style rather than functions and performance, a design environment to readily sketch the products' images and 3D shapes stimulating customers'

KANSEI is desired. In spite of these increasing demands for omnifarious and appealing design, the creation of novel and unusual 3D geometries without following conventional modeling techniques still remains a challenging problem to attack.

In this paper, we propose a new digital design framework for easily yielding totally unprecedented 3D shapes. Figure 1 illustrates the concept of our proposed system. We introduce a technique, the so-called interactive evolutionary computation (IEC) [1], as a method for accelerating the designers' invention and inspiration of new geometric shapes. The proposed IEC-based design system generates fascinating shapes through the simulation of natural evolutionary processes such as crossover, mutation, and selection according to the designers' subjective preference. The designers can easily control the IEC-based shape generation via a simple interface even if they have little knowledge and experiences of the 3D modeling. They only need to subjectively specify the degree of "likes or dislikes" for the generated shapes by rating, and request the system to discover more preferred models based on the rates. This "rating-discovery" process is iterated until the system converges on satisfactory results.

We have implemented a 3D modeling system using the implicit surface method in our previous work [2]. It allows the users to intuitively create a complex 3D object by blending a number of primitive shapes. The system, however, expects them to understand its basic functions and acquire some levels of geometric modeling skills, especially preventing beginners from readily exploiting it. Therefore, we integrate the IEC interface into the implicit modeling system to realize the framework mentioned above. The implemented system can generally be used for wide range of applications including product design, artwork creation, and even for modeling education for beginners.

2. Related work

The IEC is an evolutionary computation (EC) technology that makes the application systems be dependent on human subjective evaluation. The IEC has
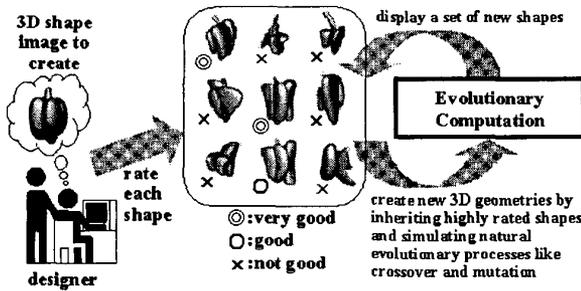
Figure 1: IEC-base 3D modeling.

been applied to several tasks in artistic, engineering, and entertainment fields for the last 10 years [1]. One of major application fields is the creative artistic design especially in the initial stage of the IEC research. Dawkins has demonstrated the power of computer-simulated evolution to create the "boimorphs", the 2D line drawings of complex structures found in the living organism [3][4]. Following his pioneering work, the two representative graphics applications were presented by Sims [5][6][7] and Unemi [8]. They established a concept, the so-called simulated breeding, to generate aesthetic graphics images based on artificial selection rather than natural selection. Although their approach was quite successful to create innovative results, the target shapes are limited to highly abstract artwork. Other artistic applications are: interactive EC-based drawing lines, such as morphological lines of insects, plant lines based on the L-system, and face drawing: interactive EC-based CG, such as for creating, 3D CG rendering of artificial life art [9], and animal and plants CG [10]: 3D CG lighting design support [11]: industrial designs. See the reference [1] for further detail survey.

A common problem of these applications is their applicability to the experts who have sufficient knowledge, experiences and a clear design objective. They are especially useful solutions for the beginners because all these applications don't assume any detail understandings of graphics theory and the system's internal structures. However, it is difficult for these applications to incorporate the experts' fine-grained operations into the IEC process because the only permitted interaction is to evaluate the results generated by the system.

Providing a uniform solution for the users with different skill levels is a difficult problem. Our proposed approach, however, supports from novice to advanced level users by integrating the capability of the IEC-based fresh discoveries into the advanced geometric modeling technology. While the beginners can create their desired shapes by only rating the presented shapes from the system, the experts are permitted to get finer control to generate novel models that are qualitatively better than previous ones. In addition to the IEC-based modeling, the experts can directly modify
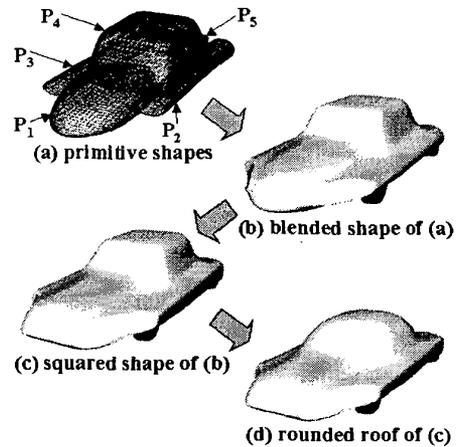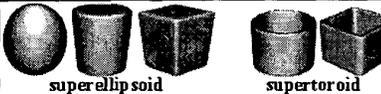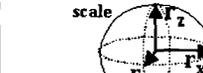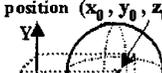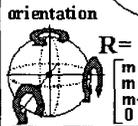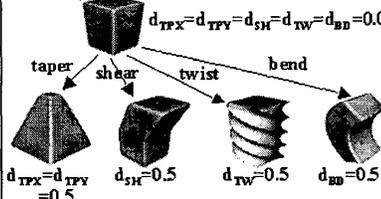

Figure 2: 3D model examples.

the internal parameters to make the generated model shapes more elaborate ones.

3. Implicit modeler functions

As shown in figure 2, our 3D modeler employs the implicit surface method [12] making easy and rapid creation of rough sketches by smoothly blending primitive shapes possible. The figure illustrates the auto body design examples, including (a) five primitive shapes P_i (i=1,..,5) used, (b) a blended shape made of the primitives in (a), (c) a squared body of (b), and (d) a rounded roof of (c). Although the traditional CAD tools can only support the detailed design stages for the experts, our modeler is adequate for the use in the preliminary stages such as 3D sketchy description and conception. It allows the users to easily rough out their ideas and imagination in 3D by iterating the primitive blending and the shape deformation.
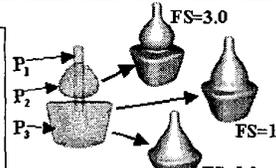
Each primitive shape has six kinds of functional parameters as described in figure 3(a). A primitive shape is represented by two kinds of superquadric functions [13] that are the superellipsoid to define solid objects and the supretoroid for hollow and cavity shapes. Other parameters define scale, position and orientation in a global coordinate system, geometric pattern, and deformation strength of the primitive shape. Changing some of the parameter values as those illustrated in the figure dynamically modify the primitive shape [14].

Further complex shape can be represented by blending multiple primitive shapes. Figure 3(b) exemplifies a bottle shape made by blending three primitives P_1, P_2 and P_3. Additional parameters as described in the figure control the global deformation and the blending operation. Figure 3(b) indicates the effect of the field strength (FS) parameter. Table 1 lists the actual parameter values to render the bottle shape as illustrated in figure 3(b).

Parameter	Roles and functions
type *primitive type in int*	super ellipsoid supertoroid
$(r_x, r_y, r_z, r_a{}^*)$ *scale in float*	scale position (x_0, y_0, z_0)
(x_0, y_0, z_0) *position in float*	orientation
R (4x4 rotation matrix in float) *orientation*	$R= \begin{bmatrix} m & m & m & 0 \\ m & m & m & 0 \\ m & m & m & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ global coordinate system
(e_1, e_2) *shape in float*	(e_1, e_2) = (2.0, 2.0) (2.0, 1.0) (1.0, 1.0) (0.1, 1.0) (0.1, 0.1)
d_{TPX}, d_{TPY} : tapering d_{SH}: shearing d_{TW}: twisting d_{BD}: bending *deformation in float*	$d_{TPX}=d_{TPY}=d_{SH}=d_{TW}=d_{BD}=0.0$ taper shear twist bend $d_{TPX}=d_{TPY}=0.5$ $d_{SH}=0.5$ $d_{TW}=0.5$ $d_{BD}=0.5$

$^*r_a$: torus radius for supertoroid primitive

(a) Functional parameters to define a primitive shape.

Parameter	Roles and functions
FS: *field strength in int*	FS=3.0
d_{TPX}, d_{TPY}: tapering d_{SH}: shearing; d_{TW}: twisting d_{BD}: bending *deformation in float (Same role with primitive deformation)*	P₁ P₂ P₃ FS=1.5 FS=1.0

(b) Functional parameters to define blending.

Figure 3: Functional parameters to define 3D

Table 1: Actual parameter values of the bottle shown in figure 3(b).

Parameters	P_1	P_2	P_3	B
type	0	0	0	-
r_x	0.3	1.0	1.25	-
r_y	0.3	1.0	1.25	-
r_z	1.0	1.35	1.4	-
r_a	0.0	0.0	0.0	-
x_0	0.0	0.0	0.0	-
y_0	0.0	0.0	0.0	-
z_0	2.2	0.75	-2.12	-
R	unit	unit	unit	-
e_1	0.1	1.0	0.15	-
e_2	1.0	1.0	0.65	-
d_{TPX}	0.0	-0.35	0.15	0.0
d_{TPY}	0.0	-0.35	0.15	0.0
d_{SH}	0.0	0.0	0.0	0.0
d_{TW}	0.0	0.0	0.0	0.0
d_{BD}	0.0	0.0	0.0	0.0
FS	-	-	-	3.0

P_i : primitive parameters, B : blending parameters, **unit**: unit rotation matrix

4. IEC-based 3D modeling

4.1. IEC-based modeling principle

The implicit modeler as mentioned in the previous section still demands the users' understanding of which parameter to adjust to perform a desired modification. Additionally, generic understanding of the geometric modeling also helps them to produce the quality shapes effectively. These requirements, however, enforce the users to become familiar with the modeling operations among others, significantly preventing them from concentrating on the actual design tasks. Our solution to this problem is to support a simple yet powerful modeling option by incorporating the IEC technology, allowing even beginners to intuitively explore aesthetically pleasing forms and shapes.

The proposed IEC-based 3D modeling uses two separate spaces: the search space and the solution space as shown in figure 4 [15]. We adopt genetic algorithms (GAs) as one of the EC technologies in the system. While the search space is a space of coded solutions to the problem (3D model generation), the solution space contains actual solutions (3D models). A coded solution, or *GTYPE* (*genotype*), must be mapped onto an actual solution, or *PTYPE* (*phenotype*). IEC maintains a population of *individuals* where each individual consists of a PTYPE and its corresponding GTYPE. A collection of individuals created in the evolutionary process forms a *generation*.

A 3D model is encoded for the simulated evolution as illustrated in figure 5. It shows the PTYPE and the GTYPE of the bottle shape mentioned in the previous section. The PTYPE consists of three primitive (P_1, P_2 and P_3) and one blending (**B**) parameters, and the corresponding GTYPE composes a bit string of all parameters held in the PTYPE. Each parameter is encoded as an 8-bit string, the so-called *gene*. For example, the parameter e_1 ($0.1<=e_1<=2.1$) is encoded as: **P(e_1) = 0.1 + G(e_1) / 128**, where P(e_1) is the value of the e_1 parameter; G(e_1) is its 8-bit gene. A collection of genes in one GTYPE is referred as a *chromosome*. The bit length of the chromosome is given as **p x n + b**, where **p** is the number of primitive parameters; **n** is the number of primitives; **b** is the number of blending parameters. Therefore, the length of the chromosome of the bottle shape in figure 5 is 60 bytes (18x3+6).

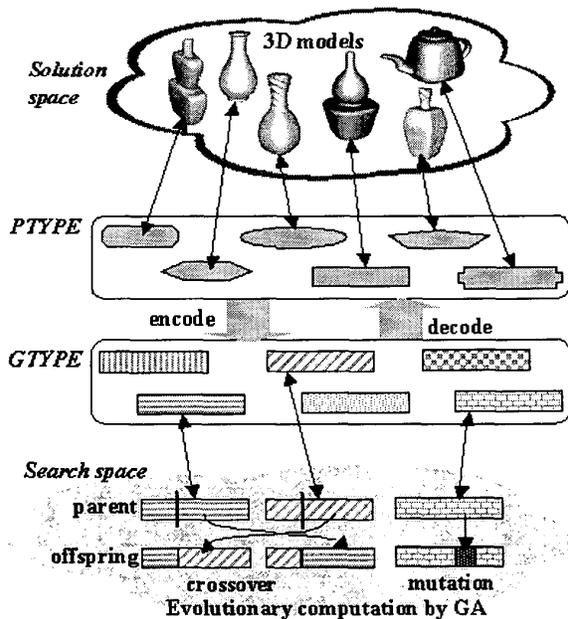Figure 4: IEC-based modeling principle.
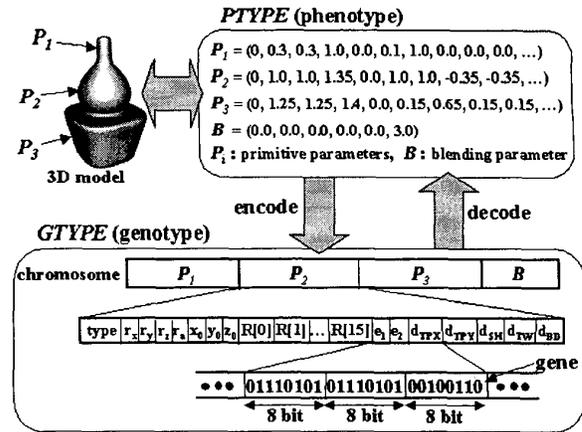

Figure 5: GA coding of 3D shape.

```
set execution options;
randomly generate an initial set of GTYPEs;
do {
        decode GTYPEs to PTYPEs for rendering 3D
        models;
        rate fitness value for each model (PTYPE) by
        user to select elite individuals;
        generate a new set of GTYPEs by executing GA
        operations such as crossover and mutation;
} while (there is no acceptable solution)
```

Figure6: IEC algorithm for 3D model generation.

4.2. IEC algorithm

Figure 5 summarises the IEC-based modeling algorithm. First of all, some execution options are defined such as a set of functional parameters (PTYPEs) to activate for the modeling and GA operator controls like crossover and mutation rates. Next, an initial set of GTYPEs is randomly created and decoded to PTYPEs for rendering corresponding 3D models. A user rates the *fitness value* for each rendered model according to his/her subjective preference. Then, a new set of GTYPEs (offsprings) is reproduced from existing GTYPEs (parents) by applying crossover and mutation as shown in figure 4. The crossover operator creates offsprings by exchanging genes of two parents. While crossover operation is frequently executed, mutation is occasionally applied to a chromosome with a low probability; e.g. 80 to 95% crossover rate and 1% mutation rate are used in our preliminary experiments as described in section 5. Highly rated GTYPEs have high probability to survive, being selected as parents to generate new offsprings. Top 20% of such highly rated individuals survive in the new generation if the crossover rate is 80%. This "evaluation and reproduction" process is iterated until acceptable model shapes are produced.

Although ordinary genetic algorithms (GAs) utilize an explicit evaluation function to automatically measure the fitness of the PTYPE, they don't permit the users to be involved in the evolutionary design. Whereas the IEC depends on the human opinions to judge the aesthetic quality of the generated models, it enables the users to loosely control and direct the results even if they have no sufficient knowledge and experiences for the 3D modeling task.

4.3. System Interface

As mentioned in the previous sections, the IEC-based modeling depends on the users' direction and evaluation to get successful results from the simulated evolution. Therefore, the interface to allow the users to easily find their preferred shapes and properly set the fitness values is a crucial element to realize. Additionally, a mechanism for providing the experts with finer control on the modeling process is a very important requirement in our framework design. Consequently, we implemented the interface consisting of two different windows as shown in figures 6 and 7. The 3D models generated through the IEC process are simultaneously displayed in a list format as depicted in figure 6. The users can observe up to twenty generated models at the same time, comparing one another, and assessing whether the shape is good or bad. The 3D modeler window as shown in figure 7 is provided to define an initial shape used for the IEC modeling and elaborate the models generated in the IEC modeling window. The users are free to invoke a specific shape in
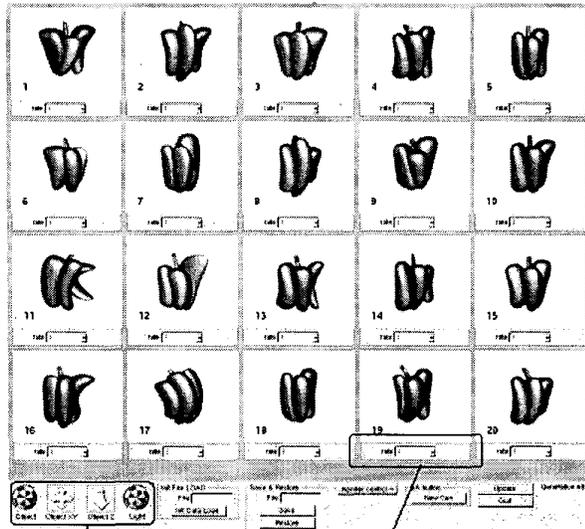
listbox control to set fitness value

rate [1]   cursor

When the user click on a listbox, a
list of fitness values drops down,
from which the user selects one.

1
2
3
4
5

rotation and translation control to modify a
viewpoint in space

Rotation control
allows the user
to directly
manipulate the
viewpoint angle
by clicking a checkerboard-
textured sphere and spinning
it by mouse movements.

Object   Object XY

Translation control
allows the user to
modify the
viewpoint position
by clicking an arrow
symbol and moving
the mouse to desired
direction.

Figure 6: Look and feel of IEC modeling window

the IEC window to the modeler window, performing the exquisite deformations by setting the definite model parameters. This bidirectionally accessible interface between the two windows enables the users to synergistically utilize the both functions, namely the IEC-based innovative shape discovery and the implicit modeling for the elaborative design.

A mechanism to freely observe the models from various angles is a mandatory function for the 3D shape evaluation. IEC window provides the rotation and translation controls to modify the viewpoints of all twenty models at a time, allowing the users to easily examine their shapes from different angles. The fitness value can be set on a listbox control attached to each model's

A list of slider
controls to
modify model
parameters.

rotation and translation controls to modify
model orientation and position in space.
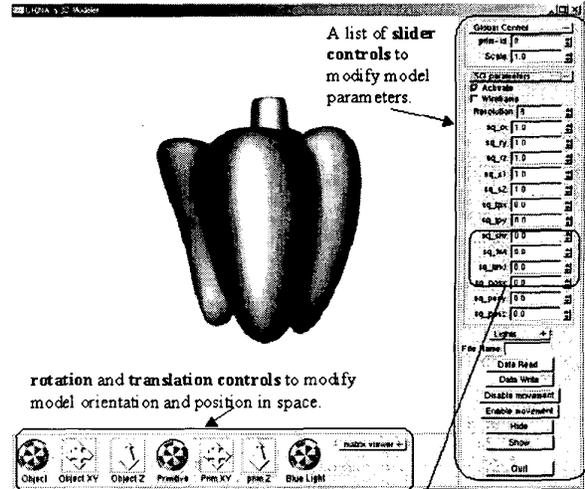
slider control to modify a model parameter

parameter → rx [0.50]   cursor   move
name   value
                      slider icon

The user selects the slider icon by mouse and move it with
holding the button. The value is modified in proportion to the
movements.

Figure 7: Look and feel of 3D modeler window

subwindow as shown in figure 6. Additionally, the functional parameters of a specific model are modified by using the slider controls in the modeler window as described in figure 7. All these operations can be handled only by a mouse device to make the users be concentrating on the modeling tasks.
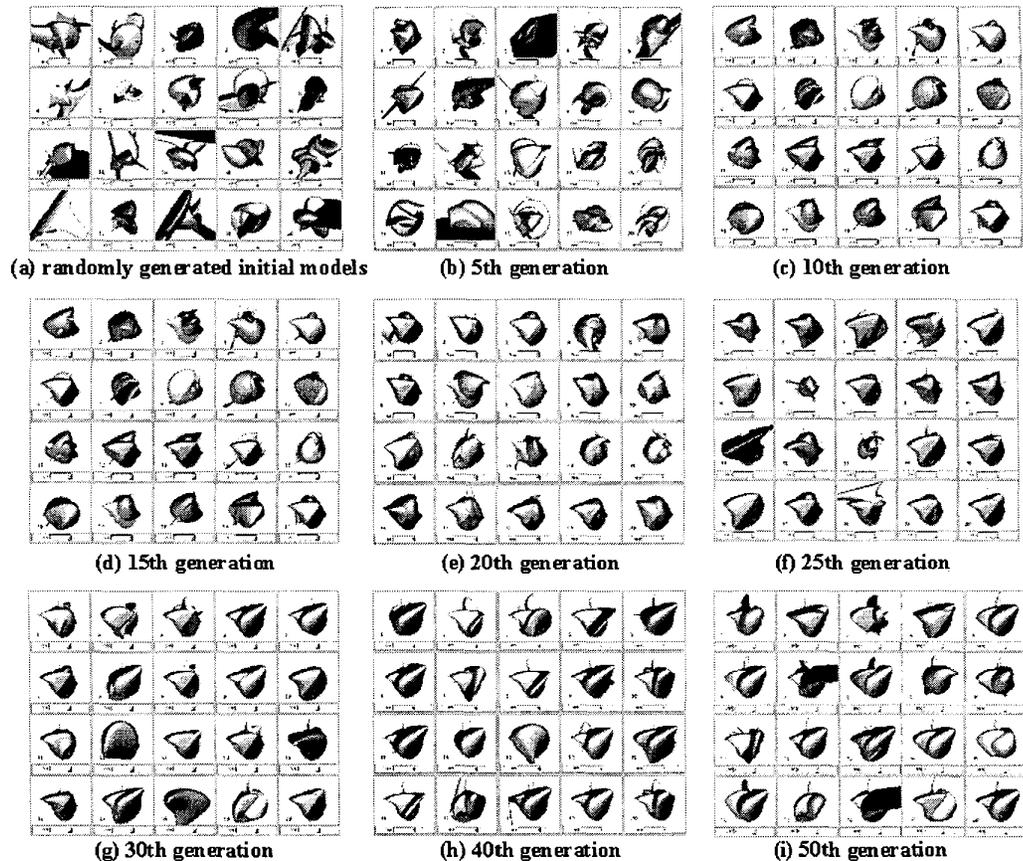
Although the better operational environment should simultaneously make both windows visible on a wide screen or dual monitors, they can be swapped back-and-forth on a single small monitor system. Because the wide screen projection system provides superior operational environment, it is a promising technology to become a common output device in the near future.

The system is written by C, C++, and OpenGL, running on Windows PCs and SGI Unix workstations. The GUI interfaces mentioned in this section are implemented by using GLUI toolkit [16].

5. Preliminary Experiments

5.1. Experiment 1

We conducted some preliminary experiments to test the effectiveness of the proposed system. To verify the system's ability to support creative 3D modeling, a green pepper is used as the target shape in the experiments because of the following reasons:

(a) randomly generated initial models (b) 5th generation (c) 10th generation

(d) 15th generation (e) 20th generation (f) 25th generation

(g) 30th generation (h) 40th generation (i) 50th generation

Figure 8: Experiment 1: Creation of green pepper shape from randomly generated initial models.

(1) everyone easily imagines its shape without depending on his/her knowledge and experience,

(2) it is an inartificial shape which is difficult to design by using the traditional CAD tools, and

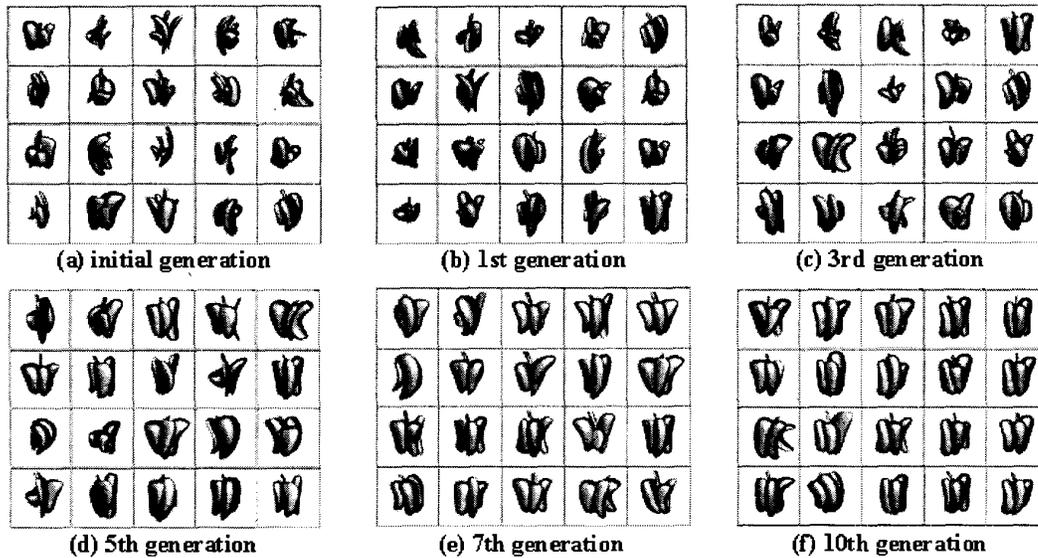(3) its shape has enough variations to encourage the subjects' flexible thinking and imagination.

Figure 8 shows a result of the experiment performed under the follwoing conditions:

(a) five primitive shapes, expecting the four primitives organize a body and the remaining one for a stalk of a green pepper, are used,

(b) scale (r_x, r_y, r_z), position (x_0, y_0, z_0), shape (e_1, e_2), tapering (d_{TPX}, d_{TPY}), and bending (d_{BND}) parameters are activated for the IEC modeling (apparently inappropriate parameters to make the green pepper shape like twisting and shearing are excluded, and orientation parameter is fixed to a unit rotation matrix),

(c) the population of individuals is 20,

(d) the crossover rate is 95%,

(e) the mutation rate is 1%, and

(f) the fitness value is a rate on a scale of 1 to 5 (the worst to the best correspond to 1 to 5).

While 19 offsprings (95%) out of 20 are generated by the crossover operation in every generation, the only 1 individual (5%) is treated as a highly rated elite parent to survive.

As perceived from figure 8, many random shapes are reproduced during the first dozen generations. A few green pepper-like shapes gradually appear through the 15th to the 25th generations. The positive selection of these likely shapes seems to happen during this period. The stalk of the green pepper also occurs in this period. Finally, the IEC modeling reaches a point of convergence up until the 30th to the 50th generations. These generations always include some likely shapes and only small changes are observed during the iteration.

We employed a few beginners as subjects for this experiment, and all of them completed the modeling within about 30 to 50 minutes. All subjects finally get the target shape in the 30th to the 50th generations. The average evaluation time is about one generation a minute. The number of individuals in a generation should not be less than twenty to get the successful results. It always is possible to find at least one likely shape to rate a highest score out of the twenty models. If less population applies

484

| (a) initial generation | (b) 1st generation | (c) 3rd generation |
| (d) 5th generation | (e) 7th generation | (f) 10th generation |

Figure 9: Experiment 2: Creation of green pepper shape from a roughly modeled initial shape.
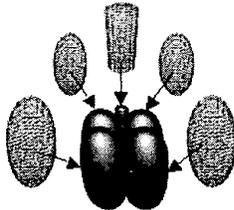
Figure 10: An initial shape used for experiment 2.

to the modeling, it may be impossible to find any likely ones and make the iteration toward the target shape.

Enormous amount of training time and efforts is required for the beginners to create the shape like the green pepper by using the traditional CAD tools. Therefore, the proposed IEC-based modeling is thought to be a useful method to support creative modeling for the beginners.

5.2. Experiment 2

The beginners-oriented modeling method as explained in section 5.1 always is not a suitable approach to the medium- to expert-level users who can create the 3D shape without the IEC interface. Because they tend to have existing 3D models designed for their own goals and targets, it seems to be desirable to start the new shape creation from such existing models rather than from scratch. Therefore, we add an optional function to load the pre-existing 3D models as the initial set of GTYPEs instead of the randomly created ones in the IEC modeling.

Figure 9 shows a result of the same experiment with the previous section, except that the existing rough shape as shown in figure 10 is used to create an initial set of models. As easily noticed from figure 9, a few initial

generations already include some very likely shapes. Then the continuous allocation of highest scores to these likely shapes make the 7th to the 10th generations contains enough variations of natural green pepper shapes. The crossover rate is changed to 80% to preserve more elite individuals on each generation for fine tuning.
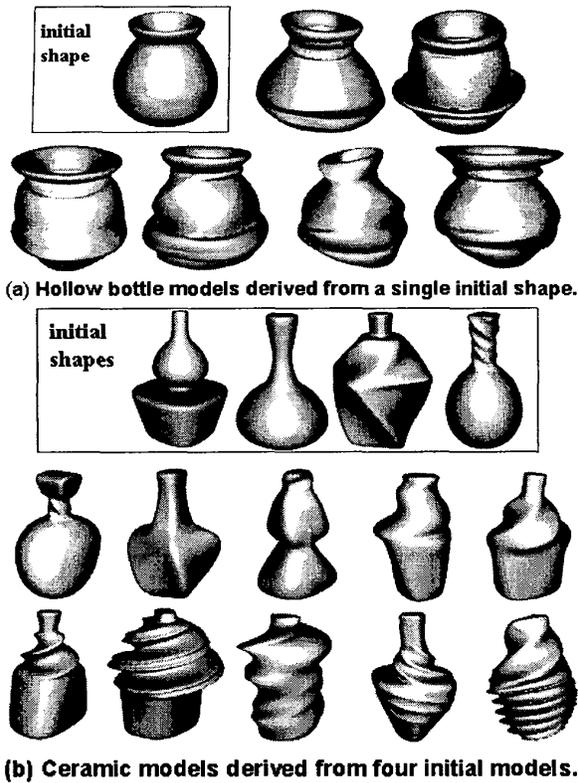
Figure 11 illustrates another examples to create ceramic models from existing shapes. Figure 11(a) shows some variations of hollow bottle models derived from a single initial shape and figure 11(b) presents the example solid models spawned from the four initial shapes. All these models are created in a few generations up to the 5th. Although the system yields interesting variations even by using a single shape, the result proves that the more initial shapes are specified, the more novel and surprising forms appear.

Eliminating artificial and geometrical properties of the models to look real and natural is very difficult issue for everyone in the 3D modeling. Figures 9 and 11 reveal a fact that the proposed method is effective for the skilled users to offer stimulative new ideas by suggesting natural and unexpected shapes.

While the GUI to control the IEC modeling is easy to use, it also is possible to further improve the operationality by adding other intuitive interaction methods such as voice and gesture [17].

6. Conclusions and Future Work

We proposed a new 3D modeling system based on the IEC technology. The system allows the users to generate entirely new shapes by starting from little or nothing

(a) Hollow bottle models derived from a single initial shape.

(b) Ceramic models derived from four initial models.

Figure 11: Ceramic model creation from existing shapes.

through the interaction with the simulated evolution. The system's potential for creative design has been shown by our preliminary experiments.

Conducting more comprehensive modeling experiments by different target shapes and subjects with various skill levels is the second stage of our research. Because the system efficiency highly depends on the users' performance to perceptually select and evaluate their preferred models, the subjective tests and their statistical analysis also are very important studies. Further enhancements such as the integration of the VR interaction methods [17] and the incorporation of the distributed data sharing capabilities [18] are planned to make the system be usable for various application domains on the network.

Acknowledgement

References

[1] Takagi, H.: Interactive Evolutionary Computation – Cooperation of computational intelligence and human KANSEI -, Proc. of IIZUKA'98, World Scientific, pp.41-50, 1998.

[2] Nishino, H., Utsumiya, K., Sakamoto, A., Yoshida, K., and Korida, K.: A Method for Sharing Interactive Deformations in Collaborative 3D Modeling, Proc. of ACM VRST'99, pp.116-123, 1999.

[3] Dawkins, R.: The Blind Watchmaker, Longman, Essex, 1986.

[4] Dawkins, R.: The Evolution of Evolvability, Artificial Life (Ed. by Langton, C. G.), Addison-Wesley, pp.201-220, 1989

[5] Sims, K.: Artificial Evolution for Computer Graphics, Proc. of SIGGRAPH'91, pp.319-328, 1991.

[6] Sims, K.: Interactive Evolution of Dynamical Systems, First European Conf. on Artificial Life (Eds. by F.J. Varela and P.Bourgine), Paris, France, MIT Press, 1991.

[7] Sims, K.: Interactive Evolution of Equations for Procedural Models, The Visual Computer, Vol. 9, No.8, pp.466-476, 1993.

[8] Unemi, T.: A Design of Multi-Field User Interface for Simulated Breeding, Proc. 3rd Asian Fussy System Symposium, pp.489-494, 1998.

[9] Todd, S. and Latham, W.: Artificial Life or Surreal art?, 1st European Conf. on Artificial Life, MIT Press, pp.504-513, 1992.

[10] Graf, J. and Banzhaf, W.: Interactive Evolution for Simulated Natural Evolution, Artificial Evolution. European Conf. (AE'95). Selected Papers. (Eds. by Alliot,J. M. et al.), Springer-Verlag, Berlin, Germany, pp.259-272, 1995.

[11] Aoki, K. and Takagi, H.: 3-D CG Lighting with an Interactive GA, Proc. of KES'97, pp.296-301, 1997.

[12] Bloomenthal, J. et al.: Introduction to Implicit Surfaces, Morgan Kaufmann Publishers, 1997.

[13] Barr, A.H.: Superquadrics and Angle Preserving Transformations, IEEE Computer Graphics and Applications, Vol.1, No.1, pp.11-23, 1981.

[14] Nishino, H., Fushimi, M., Utsumiya, K., and Korida, K.: A Virtual Environment for Modeling 3D Objects Through Spatial Interaction, Proc. of IEEE SMC'99, Vol.6, pp.81-86, 1999.

[15] Bentley, P. et al.: Evolutionary Design by Computers, Morgan Kaufmann Publishers, 1999.

[16] Paul Rademacher: GLUI User Interface Library, http://www.cs.unc.edu/~rademach/glui/.

[17] Nishino, H., Korida, K., and Utsumiya, K.: Deformable 3D Shape Representation Using Bimanual Gestures, Trans. of IPSJ, Vol.40, No.2, pp.698-701, 1999. (in Japanese)

[18] Nishino, H., Mori, Y., Utsumiya, K., Yoshida, K, and Korida, K.: A Collaborative Design Framework in a Distributed Virtual Environment, Proc. PDPTA'99, Vol.1, pp.348-354, 1999.