파티클 속성을 사용한 L-시스템 트리

주 우 석†·박 현 민††·방 순 정†††

요 약

컴퓨터 그래픽에서 L-시스템은 주로 나무나 풀, 꽃 등 자연적인 모습을 생성하기 위해 사용된다. 이는 임의로 제시된 최초의 모양에 다중 축소복사 이론을 반복적으로 적용함으로써 가능하다. 본 논문은 일반적인 L-시스템 나무의 모습을 변형하여 더욱 현실적으로 묘사하는데 목적이 있다. 본 논문에서는, L-시스템에서 사용되는 단순한 반복함수를 적용하는 대신에, 반복함수를 적용하는 과정에서 나뭇가지 하나하나를 생명체로 간주하여 생명체의 속성을 부여한다. 가지의 수명, 생장 속도, 모양 변화, 외부환경에 따른 유인 등 생명체가 지닌 속성은 파티클 시스템을 구성하는 입자들이 지닌 기본 속성이기도 하다. 속성의 적용을 위해서 개별 속성 별로 필요한 매개변수를 바탕으로 한 가설로서의 모델을 제시하고 결과적인 다양한 나무의 모습을 제시하였다.

L-system Tree with Particle Attributes

Wouseok Jou†·Hyunmin Park††·Soonjung Bahng†††

ABSTRACT

In computer graphics, L-system is primarily used for the production of such natural shapes as flowers, trees, and grass. It is possible by iteratively applying the theory of multiple-reduction-copy-machine to an arbitrary initial shape. The purpose of this paper is to modify the shape of ordinary L-system trees so that more realistic trees can be generated. Instead of applying simple iterative function system of the L-system, we regard each branch of the trees as a living thing, and endow them with corresponding attributes. Such branch attributes as lifetime, growth speed, shape variation, attraction by environment are known to belong to the attributes of the particle system. We presented modeling methods as hypotheses for each of the attributes based on parameters, and shown the resulting diverse tree shapes.

키워드 : L-시스템(L-system), L-트리(L-trees), 파티클시스템(Particle system)

1. Introduction and Related Works

The MRCM (Multiple Reduction Copy Machine) [1, 2] is the basic method to create fractal patterns. Starting with an arbitrary image, one application of the machine produces a new image, a collage of contracted copies of the first image. Applying the machine to this new image and iteration process leads to a final image, the attractor of the machine. The theory states that the nature replicates itself by iterative feedback mechanism. The attractor of the machine is completely independent of the starting image, but the intermediate stages of the iteration very much depend on the initial image.

Lindenmayer L-system [2-5] was created for the description and creation of natural growth processes. It's a formal language for modeling the reproduction rule of the MRCM. For instance, if we denote a string of cells by a string of symbols, we can signify the cell reproduction rule as follows :

$$a \rightarrow aB,\ b \rightarrow aB,\ A \rightarrow bA,\ B \rightarrow bA. \qquad (1)$$

With initial cells *aB*, we have four cells *aBbA* by applying the reproduction rule. In the next cell division, we have eight cells, namely *aBbAaBbA*, and then sixteen cells *aBbAaBbA aBbAaBbA* and so forth [2].

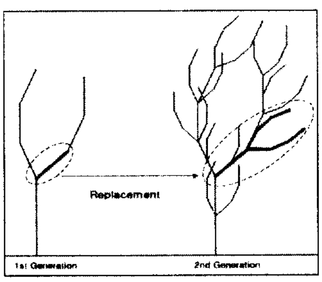
In L-system, we view a plant as a linear or branching structure composed of repeated units called modules. An L-system describes the development of this structure in terms of rewriting rules of productions, each of which replaces the predecessor module by successor modules. (Figure 1) shows how the replacement proceeds. The 2^{nd} generation tree is acquired by simple replacement or reproduction rule. The eight modules or branches composing the 1^{st} generation tree are simply replaced by reduced copy of its own 1^{st} generation shape to produce 2^{nd} generation tree.

In computer graphics, the particle system[3] is primarily used for modeling fuzzy objects such as clouds, smoke, wa-

† 정 회 원 : 명지대학교 컴퓨터공학부 교수
†† 종신회원 : 명지대학교 컴퓨터공학부 교수
††† 준 회 원 : 명지대학교 대학원 컴퓨터공학부
논문접수 : 2002년 7월 6일, 심사완료 : 2002년 10월 10일

ter and fire. The fuzzy objects do not have smooth, well-defined surfaces. Instead their surfaces are irregular, complex, and ill defined. The particle system models and approximates these objects by a collection of small primitive particles. Most importantly, the particle system regards the particles as living things. That is, the particles change from time to time as if they are "alive." New particles are born and old particles die. They can move into different direction with differing speed. They have different size and shape. Sometimes they are attracted and pulled by some other objects, and they avoid collision by detecting the presence of nearby particles.

In this paper, we will apply these "aliveness" characters of the particles to the branches composing the ordinary L-system tree. Since the branch as a living thing will also have those attributes of the particles, endowing them with the aliveness attributes can greatly increase the realism or naturalness of the tree. Although the particles are dynamically moving objects and the trees are static ones, those attributes can be applied at the conceptual level. In this paper, five of the particle attributes will be applied to the ordinary L-system tree ; namely lifetime, speed, shape, attractor, and collision avoidance.



(Figure 1) Application of the reproduction rule in L-system tree

Computer modeling of plant development started when Ulam applied cellular automata to simulate the development of branching patterns [6]. Lindenmayer proposed the formalism of L-systems as a general framework for plant modeling [4, 5]. Research on the modification of the L-system tree has been mainly concentrated on the interaction between the plant and its environment. The day length can be used in controlling the initiation of flowering [7], and the daily temperatures can be used to modulate the growth rate [8]. The spread of grass can be controlled at the presence of obstacles [9]. In sighted or exogenous mechanisms, parts of a plant influence the development of the other parts of the same or a different plant through the space in which they grow [10, 11]. Competition for space between segments of branching structures appears in [10, 12]. Competition for light between branches of trees appears in [13, 14]. Research on smooth animation of plant development can be found in [15, 16]. Greene proposed a model of branching structures suitable for accretive growth [15], and Prusinkiewicz proposed a differential method to fill the animation gap between the discrete L-system trees [16].

2. Application of the Particle Attributes

2.1 Lifetime : Modeling and Experimental Result

In ordinary L-system, the reproduction rule assigns one offspring for each branch. As was shown in (Figure 1), each of the eight branches of the 1st generation tree replicates its own offspring. As a result, the number of branches for each generation grows as a power of 2. For instance, 8 for the 1st generation, 64 for the 2nd generation, and so on.

In our approach, we regard the branch as a particle with its own lifetime. The lifetime in this sense means that under certain circumstances, some branches may quit further growth and stop reproduction. For instance, random environmental factors, such as mal-nutrition and draught, may affect the total number of surviving branches. In order to model this natural random phenomenon, we vary the total number of the branches that can be produced in i^{th} generation as follows :

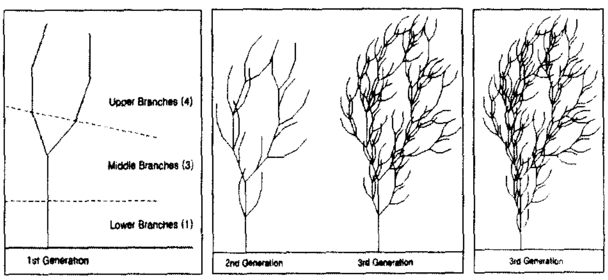
$$NB_i = MB_i + Round(Rand(\ ) \times VB_i) \qquad (2)$$

The parameter MB_i is the mean number of the branches, Rand( ) is the random number, and VB_i is the variance of the number of the branches. By these parameters, we can control and simulate the characteristics of the lifetime in nature.

Furthermore, spatial position of a given branch can greatly affect the growth. For instance, upper branches may absorb more sunlight, and thus can live longer. On the contrary, lower branches with shorter lifetime will stop growth more frequently. To accommodate this natural property, we subdivide the branches into three geometrical categories ; namely, upper, middle, and lower as shown in (Figure 2) (a). The number of surviving branches can be assigned for each category according to corresponding probability. We also invoke another random function choose exact surviving branch inside each category. By making the selection process to be random, balance between the number of left and right branches is automatically taken care of.

The 3rd generation tree in (Figure 2) (b) results from applying Equation (2) to the 2nd generation tree of the figure. The parameters used are MB of 34, VB of 32, Rand( ) of .5. In this case, the resulting number of surviving branches becomes 50 as shown in <Table 1>. Out of 64 branches in the normal 2nd generation, only 50 branches survive. By assigning arbitrary probability to each category, 30 of them are assigned to upper, 15 of them to middle, 5 of them to lower. In addition, because of the random function, the number of surviving branches in each class does not exactly match the probability.

Comparing our tree shape with that of the regular L-system tree shown in (Figure 2) (c), we see that much of the lower and middle branches stopped further growth. It can be explicitly noticed near the bottom of the trees. Overall, resulting tree appears to be sparser than that of the regular L-system tree. Furthermore, sparsity is shown near inside of the tree where the middle and lower branches are concentrated. Since the growth of the inner branches tends to be slowed down, it looks natural.



(Figure 2) (a) classification ; (b) Tree by applying Equation (2) ; (c) Ordinary L-system tree

<Table 1> Branch data of (Figure 2) (b)

	Upper Branch	Middle Branch	Lower Branch
Total Branches	32	24	8
Assigned Probability	.6	.3	.1
Surviving Branches	30	15	5
Expired Branch	2	9	3

2.2 Speed : Modeling and Experimental Result

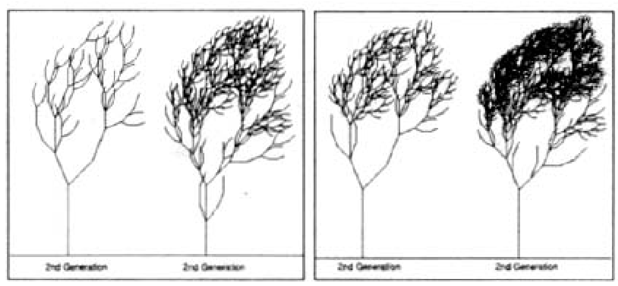
The particles can move with different speed and acceleration. By the same token, we can make the branches to grow with different speed. The growth speed in this sense corresponds to the number of reproduction for a given period. That is, we let some branches to reproduce more times than others. The variation can be thought of as a violation of the ordinary L-system tree in which every branch reproduce just once. At the expense of the violation, more diverse of the tree can be generated.

The variation in the growth speed may depend on the branch position and season. For instance, the summer provides enough sunlight for the branches to reproduce many times. In view of the branch position, the upper branches are usually exposed to outer sunlight, and they have more chance to repeat reproduction. Considering these factors, the number of reproduction at the i^{th} generation can be expressed as follows :

$$NR_i = PF + SM_i + Round(Rand() \times VR_i) \quad (3)$$

where SM_i is the seasonal mean number of reproduction, Rand( ) is the random number, VR_i is the variance, and PF is the positional factor of the branch. We can arbitrarily assign PF of 1 for the middle and lower branches, and 2 for the upper branches. The parameter can be adjusted for the proper control of acceleration and deceleration of the growth.

(Figure 3) (a) shows the result of controlling growth speed by Equation (3). Although both trees in (Figure 3) (a) are the 2nd generation trees, they show great difference in the number of reproduction. The top portion of the right tree has repeated three reproductions, while the bottom portion has repeated just once. <Table 2> shows the parameters used for the generation of (Figure 3) (a). By varying the VR in Equation (3), the difference can be amplified. (Figure 3) (b) shows the tree shapes when the variance was set to 2., while the remaining parameters remain the same.



(Figure 3) (a) Difference in the growth speed ; (b) Amplification of the difference

<Table 2> Data of the sample running in (Figure 3). BR means the number of branches

	SM	Upper Branch			Middle Branch			Lower Branch		
		PF	BR	NR	PF	BR	NR	PF	BR	NR
(Fig. 3) (a)	0	2	21	2	1	11	1	1	7	1
			17	3		8	2			
(Fig. 3) (b)	0	2	21	3	1	11	2	1	7	2
			17	4		8	3			

2.3 Shape : Modeling and Experimental Result

Particles can change its radius or mean size. Likewise, the tree can change its branch length. The branches in this sense include trunks and stems. In nature, the growth rate is retarded as the tree grows up. Therefore, we assume that it is non-linear. More specifically, we assume that the tree grows exponentially in early times of its growth, and logarithmically near its full maturity. If we let the tree reaches maturity at m^{th} generation, the growth can be modeled as follows :

a) Up to Maturity : $BL_i = (GR)^i \pm Round(Rand(\ ) \times VR_i)$

b) After Maturity : $BL_i = \log_{GR} i - \log_{GR} m + (GR)^m \pm Round(Rand(\ ) \times VR_i)$ (4)

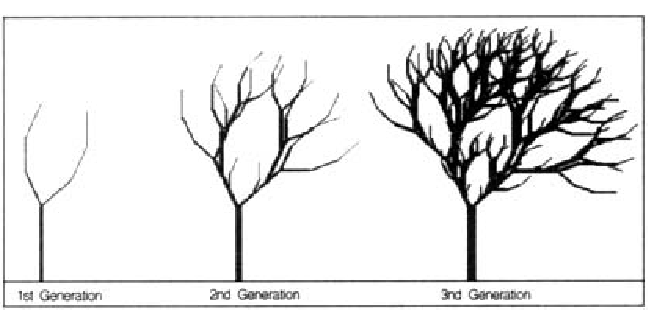
where BL_i means the branch length at i^{th} generation, VR_i means the variance for the scaling factor, and GR means the growth rate. Here, $(GR)^m$ is the total growth up until the m^{th} generation.

The width of the tree also increases as the tree grows. In this case, however, the incremental width is dependent on the branch positions. Since the growth is cumulative, old trunks must be wider than the new born branches. Incorporating this consideration into our model, we have

$$BW_i = i + PF + Round(Rand(\ ) \times VR_i) \quad (5)$$

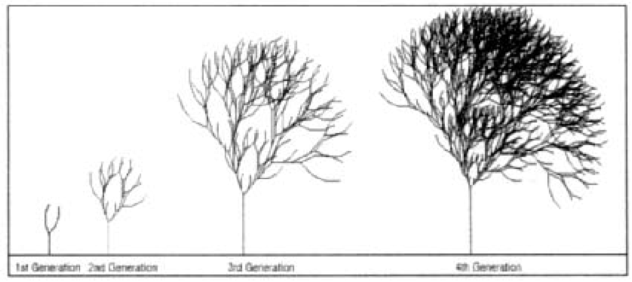
Where BW_i is the branch width of the branches which lived for the total duration of i generations, and PF is the positional factor.

(Figure 4) shows the growth pattern according to Equation (4). Random values are all set to zero in producing the figure. In the figure, the growth rate GR is set to 1.5. By Equation (4), the 2^{nd} generation branches have the length of 2.25, and those of the 3^{rd} generation have the length of 3.375. Since the maturity was set to the 3^{rd} generation, the 4^{th} generation tree has the branch length of 4.078. Consequently, the marginal growth is slowed down during the transition from the 3^{rd} to the 4^{th} generation.



(Figure 4) Simulation of natural growth pattern by Equation (4)

(Figure 5) shows the width increase during the first three generations. The width is controlled by Equation (5), where the positional factor was set to 2 for the upper, 1 for the middle, and 0 for the lower branches. These width factors are applied to all of the branches composing the tree.



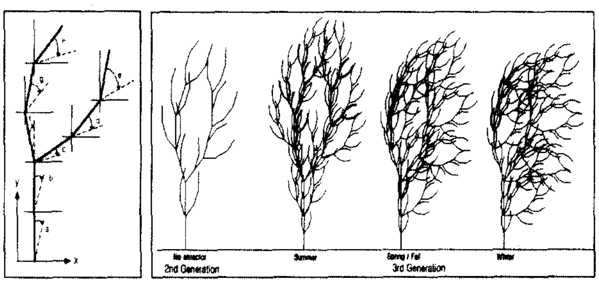
(Figure 5) Growth in the branch width by Equation (5)

2.4 Attractor : Modeling and Experimental Result

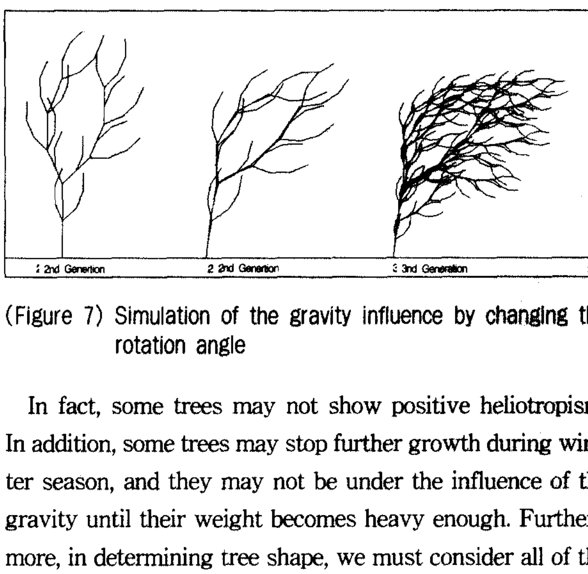
The particles are usually attracted by nearby target in the particle system. The concept can be expanded to our system by accommodating the direction of sunlight, and the source of gravity and wind.

As for the sunlight, the branches are attracted toward the sun that changes its altitude seasonally. In the northern hemisphere, for instance, the summer altitude of the sun is (90 - latitude + 23.5) degrees. To accommodate the seasonal angle variation, we rotate the direction of branches. In doing so, we restrict the rotation to the middle and upper branches only. Upon test, rotation of lower branches distorts general tree shape to the extent of anomaly. The observation that the outer branches are more prone to bend toward sunlight rationalizes our approach. In addition, we first rotate all the 2^{nd} generation branches, and then reproduce the 3^{rd} generation based on those rotated branches. Every branch except the lower branches experiences the same amount of rotation. The rotation angle can be controlled in accordance with and the location and selected season.

The trees are also under the influence of wind and gravity. The wind may exert forces toward a certain direction. Also the gravitational force may exert downward forces. Since their net effect is simply to bend the branches, these forces can be treated the same way. For the simulation of the bending, we control the tree shape by assigning different rotation angles for each branch as in (Figure 6) (a). Here, the 1^{st} generation branches of (Figure 1) are classified according to their relative position, and labeled from 'a' to 'z'. In this way, we can assign different rotation angle for each branch. (Figure 6) (b) shows the growth pattern due to the seasonal change of the solar position. With the northern latitude of 38 degrees, the solar position becomes 76, 53, 29 degree for the summer, spring, and winter respectively. As can be seen, the branches stretch out toward those directions for each season. (Figure 7) shows the bending due to the gravitational force. Here, the rotation angle of 5, 5, 6, 7, 8, 6, 7, and 8 degrees were used for each branch 'a' to 'z' in (Figure 6). In addition, the bending was applied cumulatively. For instance, the bended 3rd generation tree is produced by the 2nd generation to which we have already applied gravitational bending.Assigning different rotation angle for each branch can control the amount of the bending.



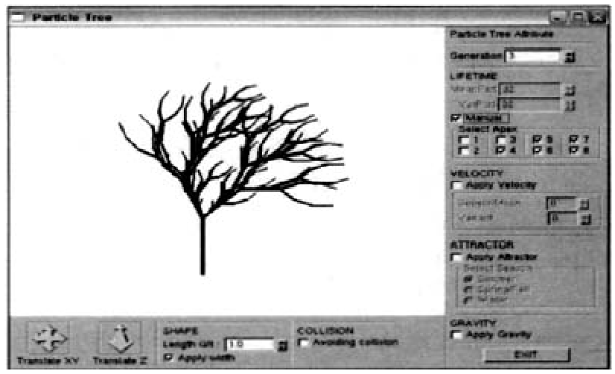
(Figure 6) (a) Bending angles for each branch ; (b) Attraction by the solar position



(Figure 7) Simulation of the gravity influence by changing the rotation angle

In fact, some trees may not show positive heliotropism. In addition, some trees may stop further growth during winter season, and they may not be under the influence of the gravity until their weight becomes heavy enough. Furthermore, in determining tree shape, we must consider all of the above-mentioned factors simultaneously.

In order to accommodate this consideration, our experimental shape generator uses a combinational input interface shown in (Figure 8). Here, such factors as the lifetime, growth speed, shape and attractors can be applied simultaneously. Furthermore, the parameters associated with each factors can be adjusted. If a tree does not show positive heliotropism, corresponding weight parameter can be set to zero. Actually, producing exact natural tree shape may involve more complex factors other than those presented in this paper. Our approach in this paper is to propose some important factors at the level of hypothesis, and to observe the resulting tree shape. Although naturalness of such tree shape may be a matter of subjectivity, the diversity of the tree shapes can be exploited to such computer graphic areas, which require production of artificial graphic images that mimics natural shapes.



(Figure 8) Input interface for the generation of the tree shapes

3. Concluding Remarks

Although the particle system was originally devised to simulate fuzzy objects such as cloud or mountain, the particles are modeled as if they were living things. The branches composing a tree make no difference from the particles in that they are also living things. We have applied the general attributes of the particle system into ordinary L-system trees. Five of the attributes - namely lifetime, speed, shape, and attractor - were used to control and produce more diversified tree shapes. The branches may stop further growth. Some of them grow faster than others. Their length and width vary with years. They stretch toward the sun. These natural traits are applied during the iterative reproduction process of the L-system. The four control attributes can be applied independently or can be integrated and applied simultaneously.

References

[1] Barnsley, M., *Fractals Everywhere*, Academic Press, Boston, 1992.

[2] Peitgen, H. O., Jurgens, H., and Saupe, D., *Chaos and Fractals*, Springer-verlag, New York, 1992.

[3] Reeves, W. T., "Particle Systems-A Technique for Modeling a Class of Fuzzy Objects," Computer Graphics, Vol.17, No.3, pp.359-376, 1983.

[4] Lindenmayer, A., "Mathematical Models for Cellular Interaction in Development, Parts I and II. Journal of Theoretical Biology," Vol.18, pp.280-315, 1968.

[5] Lindenmayer, A., "Developmental Systems without Cellular Interactions, Their Language and Grammars, Journal of Theoretical Biology," Vol.30, pp.455-484, 1971.

[6] Ulam, S., "On Some Mathematical Properties Connected with Patterns of Growth of Figures, Proc. of Symposia on Applied Mathematics, American Mathematical Society," Vol. 14, pp.215-224, 1962.

[7] Fruters, Dl, and Lindenmayer, A., "A Model for the Growth and Flowering of Aster Novae-angliae on the Basis of Table(1,0) L-systems," Eds., Lecture Notes in Computer Science 15., Springer-Verlag, Berlin, pp.24-52, 1974.

[8] Hanan, J., "Virtual Plants-Integrating Architectural and Physiological Plant Model," Proc of ModSim 1995, Vol.1, pp. 44-50, 1995.

[9] Arvo, J., and Kirk, D., "Modeling Plants with Environment-sensitive Automata," Proc., of Ausgraph'88, pp.27-33, 1988.

[10] Bell, A., D., "The simulation of Branching Patterns in Modular Organisms," Philos. Trans. Royal Society London, Series B-313, pp.143-169, 1986.

[11] Prusinkiewicz, P., "Visual Models of Morphogenesis, Artificial Life 1," pp.61-74, 1994.

[12] Cohen, D., "Computer Simulation of Biological Pattern Generation Processes," Nature 216, pp.246-284, 1967.

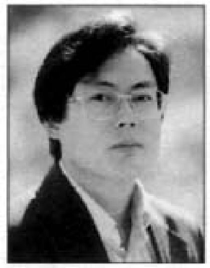
[13] Chiba, N. et al., "A Growth Model Having the Abilities of Growth-regulations for Simulating Visual Nature of Botanical Trees," Computers and Graphics, Vol.18, No.4, pp.469-479, 1994.

[14] Takenaka, A., "A Simulation Model of Tree Architecture Development Based on Growth Response to Local Light Environment," J. of Plant Research, Vol.107, pp.321-330, 1994.

[15] Greene, N., "Voxel Space Automata : Modeling with Stochastic Growth Processed in Voxel Space," Proc. of Computer Graphics, ACM SIGGRAPH 1989, pp.175-184, 1989.

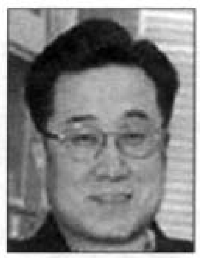
[16] Prusinkiewicz, P., Hammel, M. S., and Mjolsness, E., "Animation of Plant Development," Proc. of Computer Graphics, ACM SIGGRAPH 1993, pp.351-360, 1993.

[17] Reynolds, C. W., "Flocks, Herds, and Schools : A Distributed Behavioral Model, Proc. of Computer Graphics," ACM SIGGRAPH 1987, pp.25-34, 1987.



주 우 석

e-mail : red@mju.ac.kr
1976년~1983년 서울대학교 공과대학 전자공학과
1983년~1985년 한국 IBM, 데이콤 정보통신 연구소
1985년~1987년 Univ. of Florida 컴퓨터공학 석사
1987년~1991년 Univ. of Florida 컴퓨터공학 박사
1992년~현재 명지대학교 공과대학 컴퓨터학부 교수
관심분야 : 컴퓨터 그래픽스, 멀티미디어, 인터넷 원격교육 소프트웨어



박 현 민

e-mail : hpark@mju.ac.kr
1985년 서울대학교 공과대학 전자공학과
1988년 North Carolina State University Electrical & Computer 공학석사
1995년 North Carolina State University Electrical & Computer 공학박사
1966년~현재 명지대학교 공과대학 컴퓨터학부 부교수
관심분야 : 멀티미디어, ATM 교환기, 네트워크 트래픽 관리 및 보안

방 순 정

e-mail : soonjung@mju.ac.kr
1997년~2001년 명지대학교 공과대학 컴퓨터공학과
2001년~현재 명지대학교 공과대학 컴퓨터학부 석사과정
관심분야 : 컴퓨터그래픽스, 멀티미디어, 인터넷응용