JOURNAL OF KOREA MULTIMEDIA SOCIETY, VOL. 10, NO. 6, JUNE 2007(pp. 770-778)

Production of Contents Embodiment for Cyber Underwater Using Environment Fish Schooling Behavior Simulator

Jong Chan Kim†, Seung-Il Cho††, Kim Chee-yong †††, Eung Kon Kim††††

ABSTRACT

Fish schooling or group moving in cyber underwater is a part of beautiful and familiar ecosystem. It is not so easy to present the behavior of fish crowd naturally as a computer animation. Thanks to development of computer graphics in entertainment industry, the numbers of digital films and animations is increased and the scenes of numerous crowd are shown to us. Though there are many studies on the techniques to process the behavior of crowd effectively and the developments of crowd behavioral systems, there is not enough study on the development for an efficient crowd behavioral simulator. In this paper, we smartly present the types of fish behavior in cyber underwater and make up for the weak points of time and cost. We develop the fish schooling behavior simulator for the contents of cyber underwater, automating fish behavioral types realistically and efficiently.

Keywords: Cyber Underwater Environment, Flocking Algorithm, Fish Schooling Behavior Simulator,

1. INTRODUCTION

As computer graphics are developing and digital films and animations are growing, we can meet cyber characters easily. We should develop the techniques of 3D modeling and animation to produce the cyber underwater using these cyber characters[1]. Crowd behavior scenes in cyber underwater are used in game and entertainment industries.

Characters of computer animation take an important role of putting life into cyber space. As the number of these characters are growing, the immersion of cyber space are increasing at the same rate. We can reduce the time of computation and save the resources because we don't need the real time computation in moving the crowd characters in digital films and animations. But the larger the scale of crowd is, the lower the performance of simulation for generating frames is. The term of 'crowd' means a set of someone gathering in a place or moving somewhere. The simulation of these crowd behaviors for applying to animation, cyber space, or games is crowd simulation. Although there are crowd in a same place and in a same purpose, the individual of the crowd has each different thought and behavior[2,3]. When we produce the animation of these characters by hand, we need much time and cost. The study of automation for animation should be needed to generate crowd behavior scenes realistically and effectively[4,5].

※ Corresponding Author : Eung Kon Kim, Address :(540-742) 315, Maegok, Suncheon, Jeonnam, Korea. TEL : +82-61-755-7999, FAX : +82-61-750-3620, E-mail : kek@sunchon.ac.kr

Receipt date : Mar. 5, 2007, Approval date : Mar. 23, 2007

† Department of Computer Engineering, Sunchon National University Jeonnam, Korea
(E-mail : seaghost@sunchon.ac.kr)
†† Department of Computer Engineering, Sunchon National University Jeonnam, Korea
(E-mail : blindcho@hanmail.net)
††† Department of Visual Information Engineering, Dong-Eui University Busan, Korea
(E-mail : kimchee@deu.ac.kr)
†††† Department of Computer Engineering, Sunchon National University Jeonnam, Korea

※ This research was partially supported by the MIC (Ministry of Information and Communication), Korea, under the ITRC(Information Technology Research Center) support program supervised by the IITA(Institute of Information Technology Advancement) (IITA-2006-C1090-0602-0001)

Though the processing a scene of crowd and the behavior system of crowd, related to the processing techniques of crowd behavior in VR contents, have been implemented so far, the research for developing the natural crowd behavior simulator can not be still satisfying. Therefore, we should develop the fish behavior simulation to take the role of each behavior and react to each other to present the objects in cyber underwater[6,7].

In this paper, we design a realistic and efficient Fish Schooling Behavior Simulator for the contents production of cyber underwater environment, which shows each type of fish behavior in cyber underwater smartly, and which generates the animating the behavior automatically, reducing the time and cost.

2. FLOCKING ALGORITHM AND A* ROUTE SEARCHING ALGORITHM

2.1 Flocking Algorithm

Flocking algorithm is the implementing the group movements of living things by computer program. Flocking algorithm is introduced by Craig Reynolds in 1987. Reynolds, called the father of flocking, showed the group actions of boids using four rules in Fig. 1 [8]. Reynolds called these rules steering behavior, the rules are following

Separation steering behavior gives a character the ability to maintain a certain separation distance from others nearby. This can be used to prevent characters from crowding together. To compute steering for separation, first a search is made to find other characters within the specified neighborhood.

Alignment steering behavior gives an character the ability to align itself with (that is, head in the same direction and/or speed as) other nearby characters. Steering for alignment can be computed by finding all characters in the local neighborhood, averaging together the velocity of the nearby characters. This steering will tend to turn our character so it is aligned with its neighbors.

Fig. 1. Separation, cohesion, alignment rules.

Cohesion steering behavior gives an character the ability to cohere with other nearby characters. Steering for cohesion can be computed by finding all characters in the local neighborhood, computing the "average position" of the nearby characters.

Boids should not collide with obstacles. Boids keep in some distance constantly and when they meet any obstacle, they escape to different direction from one another. This rule can be applied not only obstacles but also agents of natural enemy. The rule can move to direction of the definition in advance. Boids can not realize the direction and move against other enemies or with other boids occasionally.

We can easily implemented the group movement of units because of these flocking rules and we can use the rules as the instrument of realistic game environment for users[9,10].

The characteristics of flocking algorithm is the thing which does not have any information of each boid, that is, stateless algorithm. As each boid exams the situation without any information in flocking algorithm, the algorithm need much less memory than other crowd behavior algorithm. Flocking is emergent behavior essentially[11,12]. Though boids can not see where they go, they go into group, avoid obstacles, or parallel to others. Many games have already used this flocking. For example, Unreal and Half-Life applied this flocking to the movement of monsters or safe animals (fish or birds). As we use flocking, we can acquire more realistic results than simple scripting in RTS or RPG games[13].

2.2 A* Route Searching Algorithm

One of the most important things in games is the route searching to go to destination from certain place. For example, when we point the direction of mouse, we can arrive the destination in strategic simulation. The most useful method is A* route searching algorithm.

This method is the one to restrict efficiently for searching range using estimated cost. This has a strong point of applying various heuristic weight to character of map. When the way to goal is cut or exploded, the way will change and we can not search the route. A* algorithm is the part of many studies in Artificial Intelligent problems.

Some characteristics of A* route search algorithm are object directed algorithm, obstacle avoidance, destination arrival, every route search. Object directed algorithm is not to find the only way, but to find efficient route in searching and comparing neighbors. Obstacle avoidance consists of escape from special ground. Destination arrival use heuristic method. After every route can be examined, we should conclude the route.

In A* route search algorithm, there are three stages of start point, present position, destination goal like Fig. 2.

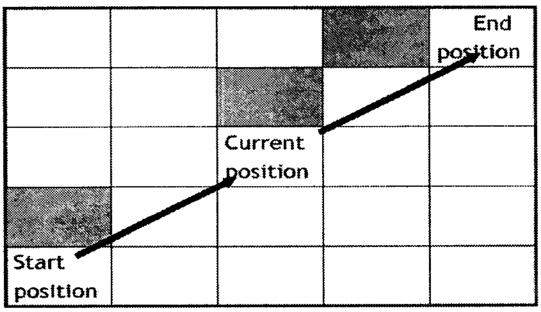


Fig. 2. State Expression of A* Route Searching Algorithm.

Present cost is the cost between start point and present position, goal cost is the cost between present position and object, and final cost is the sum of present cost and goal cost. The final cost is from the start point to destination goal and it can be distance or other things according to setting of application program. When there are many routes from start point to destination, we have to select the route with the least cost comparing with them. The present cost is the one which is produced with nodes in close list and the goal cost is the one which will be produced with nodes in open list.

When we search the route in large area, we may have a long distance between start point and destination point. In this situation, the space we should search for will increase explosively, so that the memories will take large space and the search efficiency will decrease. Therefore, we generate the midway between start point and destination point and we can search the ways by each stages.

2.3 Leader-Follower Model

Flocking Algorithm based on Crowd Simulation has the features of swimming in air and in water. And every object in cyberspace has no destination swim in cyberspace , drifts about in the water and wander about from place to place keeping in certain distances.

When crowd which has special purpose moves in cyberspace, there will be usually a leader. The character which has leader property is decided and destination and halfway are added. The model similar to real crowd is called Leader-Follower Model. We import Crowd Motion Planning in this simulator[14].

3. SCHOOLING FISHES

According to objects, the schooling form is various. In the case of bird, the number of crowd unit is small and the density of objects is not so high, and in fish, the number of crowd unit is large and the density of objects is quite high.

Fish schooling is shown simply in following way like Fig. 3 Only leader fish knows the destination and moves to target point. The rest of fish realize the position which leader shows and try to keep the position.

When the leader fish move to target point, there is an elliptical orbit around it. The leader fish endow other fish the elliptical orbit with the proper destination according to the number of fish unit.

The fish which was endowed with the information of the destination by leader has the properties to keep the position, that is, the fish has the features of alignment and cohesion. The left side of Fig. 4 shows simply the moving fish unit and the right side shows the moving direction vector of each unit. When the leader fish moves, others also follow to leader. If the leader fish moves to center, the target points of others change A1 to A1′, A2 to A2′, When this process is applied repeatedly, the simple crowd behavior of fish will appear.

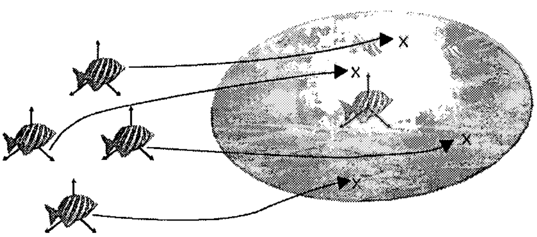


Fig. 3. Fish Schooling.

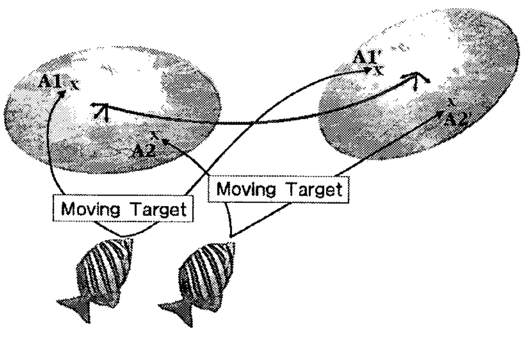


Fig. 4. Movement of Schooling Fish.

4. INTERACTION OF FISH SCHOOLING

To construct the cyber underwater, not only schooling fish but also many kinds of underwater plants are needed. We design the type of fish, the type of habitation, activeness, the step of food chain, crowd type, breeding type, avoidance, prey acquisition, and so on.

The type of fish is divided into stick and float and is decided by the movement of object property. The type of habitation is divided into temperature, depth, place. According to proper temperature of water, the movement of fish is growing. According to proper depth of water, the activeness and the rate of breeding of fish are growing. After comparing the defined place with cyber space, fish move to proper place.

Activeness is shown to percentage according to various attributes of fish. If the percentage is high, the speed of fish is fast and the areas of movement is wide. In the stage of food chain, fish decide whether it can eat or not and it appears the response of acting to predator and prey. The type of breeding is shown to percentage according to various attributes of fish. If there is a proper situation, the quantity of plankton increases to add the objects and the situation must be shown in the display.

There is an interaction of schooling fish with each other. The type of hunting is active hunting - chase and passive hunting - temptation. The predator are active to acquire the prey and the prey tries to escape from the predator. According to the kind of prey, fish can swim or float to eat plankton and can act active to eat some prey.

5. FISH SCHOOLING BEHAVIOR TYPES AND CROWD BEHAVIOR MODULE

Behavior rules of crowd may define behavior types for the fish around each object as follows.

- Wander : Fish are moving freely without any rules.

- Separation : Fish are moving at some distances.
- Alignment : Fish are moving in keeping a certain form.
- Cohesion : Fish are moving to one point.
- Hunt : Predators are hunting prey.
- Flee : Preys are running away from predators.
- Collision Avoidance : Fish are avoiding collision in a group or to obstacles.
- Path Following: Fish are moving to a certain direction.
- Leader Following: Fish are following the fish of leader.
- Seek: Fish are wandering to find prey.
- Pursuit: Fish are searching for prey.

Fig. 5 is shown the Hierarchy of Fish schooling behavior. After the type of action is selected, the behavior path is defined and then fish schooling is animated.

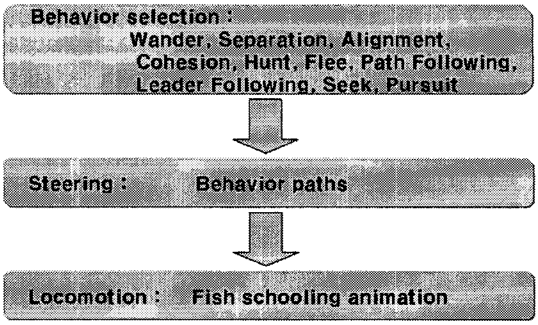


Fig. 5. The hierarchy of Fish schooling behavior.

6. SCHOOLING BEHAVIOR FRAMEWORK AND ALGORITHM

6.1 Schooling behavior modeler

To design schooling behavior modeler, after the information of underwater environment and biology is considered, the behavior is selected and the path is decided, and the animation is producted through camera control and rendering process. The elements of underwater are light, floating, wave and top side of sea. There are coral reef, turban shell, starfish, shellfish, and so on in underwater.

In the next stage, there is a mental state and it is the part of type of fish and express of fish state. For example, there are hunger, fear, wandering, satisfaction, and sex desire. A certain behavior is selected and decided behavior path in schooling behavior selection. Position, speed, view point, acceleration are selected in camera control. Fish is showed 3D effect really in rendering part. Finally, underwater environment are shown naturally in animation stage. Fig.6 showed Framework.

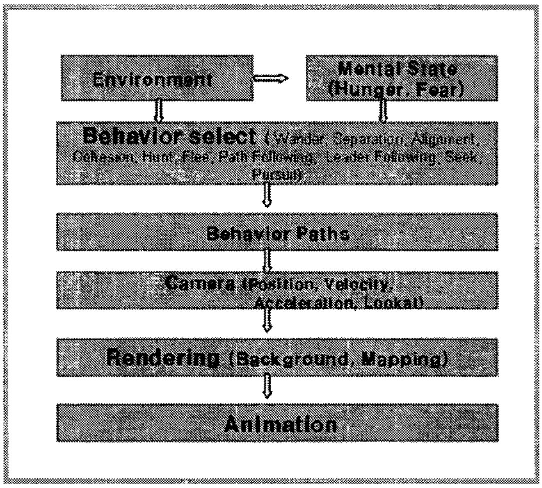


Fig. 6. Schooling behavior framework.

6.2 Schooling Behavior Algorithm

In this algorithm, fish which is defined generates in the first stage. In the next stage, the value of position is calculated, camera mode is set, and 3D object is generated. The behavior classes which the objects move, the priority order and action and exist are defined are declared. After the classes of the objectives are the number of units are increased, checked the boundary. If the boundary condition is 0, the fish turn, and if the boundary condition is 1, the fish start to the next stage. After attribution of the fish is reset and the vector of alignment, separation, cohesion are set, the schooling simulation is executed. Fig. 7 shows fish schooling behavior algorithm.

```
  Fish (ProximityDatabase& pd) //fish initialization
  Vec3 handleBoundary (void) // fish position
void update (const float
currentTime, const float elapsedTime)
{applySteeringForce
(steerToFlock ()+handleBoundary(),
elapsedTime); }
Vec3 steerToFlock(void)
proximityToken->findNeighbors
(position(),maxRadius,neighbors, FISH);
  // fish class behavior types
  CFishSchooler::FishBehavior
  F_BEHAVIOR_WANDER:
  cohesionW * 2 + alignmentW / 2.2 + Vec3::forward
  * 1.5;
  F_BEHAVIOR_SEPARATION:
  separationW * 2;
  F_BEHAVIOR_ALIGNMENT:
  alignmentW + Vec3::forward * 2;
  F_BEHAVIOR_COHESION:
  cohesionW * 2 + alignmentW / 2.2;
  F_BEHAVIOR_LEADER:
  cohesionW * 3 + alignmentW / 2.2 + Vec3::forward
  * 2;
```

```
// Fish target select
void applySteeringForce (const Vec3& force, const float
elapsedTime)
{ setSpeed (newVelocity.length());
setPosition (position() + (newVelocity * elapsedTime));
}
float separationWeight = alignmentWeight
= cohesionWeight = 3.0f;
// apply weights to components
Vec3 separationW
 = separation*separationWeight;
Vec3 alignmentW
 = alignment* alignmentWeight;
Vec3 cohesionW
 = cohesion * cohesionWeight;
// Fish redraw display
void redraw (const float currentTime, const float
elapsedTime)
{for (i = fish.begin(); i != fish.end(); i++)
 (**i).draw() ;
for(i = shark.begin(); i != shark.end(); i++)
(**i).draw() ;
}
```

Fig. 7. Schooling behavior algorithm.

Fig. 8 shows fish schooling behavior flowchart. Environment setting is initialized in Initialize stage. OpenGL API is initialized in InitializeGraphics stage. Fish schooling behavior model is generated in Display stage. 3D Schooling fish is generated and schooling behavior is executed in Draw. Finally, schooling behavior is executed dividing into fish and shark in event part.

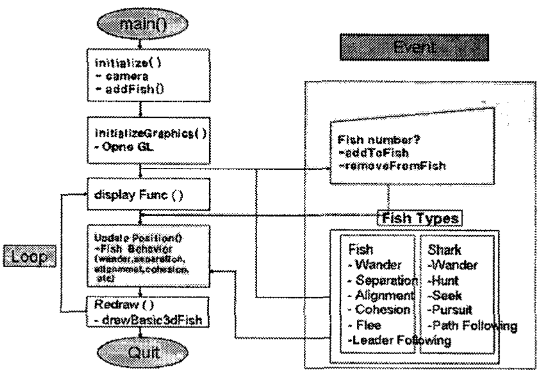


Fig. 8. Fish Schooling behavior flowchart.

7. IMPLEMENTATION AND USER INTERFACE

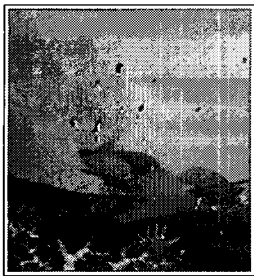
We developed using Windows XP, Visual C++ 6.0 and OpenGL API and showed 3D modeling each fish with 30 meshes. Fig. 9 shows the fish schooling behavior simulator Interface.

Fig. 9. Fish schooling behavior simulator interface.

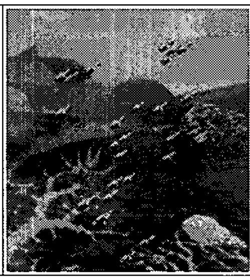
Fig. 10 is the display of separation behavior and alignment behavior. Fig. 11 is the display of cohesion behavior and wander behavior. Fig. 12 is the display of path follower behavior and a seek behavior. Fig. 13 is the display of leader follower behavior.

8. CONCLUSION AND FUTURE WORKS

We developed the fish schooling behavior simulator which generates behavior type and event in cyber underwater realistically and effectively. It made the reality and the performance superior,

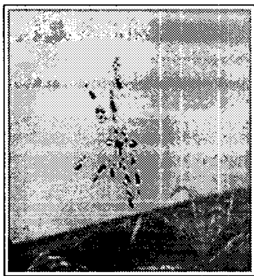


```
Vec3 steerToFlock(void)
proximityToken->
findNeighbors(position(),
maxRadius,neighbors, FISH);
F_BEHAVIOR_SEPARATION:
 return separationW * 2;
```



```
Vec3 steerToFlock(void)
proximityToken->
findNeighbors(position(),
maxRadius,neighbors, FISH);
F_BEHAVIOR_ALIGNMENT
:return alignmentW +
Vec3::forward * 2;
```

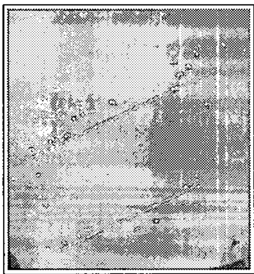
Fig. 10. Separation behavior and alignment behavior display.



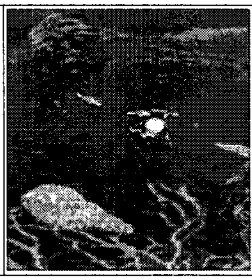
```
Vec3 steerToFlock(void)
proximityToken->
findNeighbors(position(),
maxRadius,neighbors, FISH);
F_BEHAVIOR_COHESION:
return cohesionW * 2 +
alignmentW / 2.2;
```

```
Vec3 steerToFlock(void)
proximityToken->
findNeighbors(position(),
maxRadius,neighbors, FISH);
F_BEHAVIOR_WANDER:
return cohesionW * 2 +
alignmentW/ 2.2 + Vec3::forward
* 1.5;
```

Fig. 11. Cohesion behavior and wander behavior display.

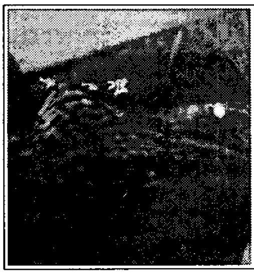


```
Vec3 steerToFlock(void)
proximityToken->
findNeighbors(position(),
maxRadius,neighbors, FISH);
F_BEHAVIOR_PATH:
d=path->getTotalPathLength()
* frandom01();
r = path->radius;
```



```
Vec3 steerToFlock(void)
proximityToken->
findNeighbors(position(),
maxRadius,neighbors, FISH);
F_BEHAVIOR_SEEK:
steer = Vec3(frandom01()*10,
frandom01()*10,
frandom01()*10);
```

Fig. 12. Path follower behavior and Seek behavior display.



```
Vec3 steerToFlock(void)
proximityToken->
findNeighbors(position(), maxRadius,neighbors, FISH);
F_BEHAVIOR_LEADER:
if(this != leader)
return cohesionW * 2 + alignmentW / 2.2 + Vec3::forward * 1.5;
return separationW +
Vec3::forward * 1.5;
```

Fig. 13. Leader follower behavior display.

expressed the interaction efficiently, and generated fish schooling behavior scene. It could control many cyber character and could simulate in real time and in PC environment.

In future works, the environment around fish such as the movement of seaweed and the current will have to display realistically in 3D. The production of ocean cultural VR contents in fluid simulation and underwater animal will have to be developed.

REFERENCES

[1] Kingsley Stephens, "Modelling Fish Behaviour," 2003.

[2] C. W. Reynolds, "Flocks, Herds, and Schools: A Distributed Behavioral model," *ACM SIGGRAPH 87*, Vol. 21(4), pp.25-34, July 1987.

[3] C. W. Reynolds, "Interaction with groups of autonomous characters," *GDC* 2000, pp. 449-460, 2000.

[4] D. Mcforlan, "Animal Behaviour," *Pitmanm*, 1985.

[5] W. F. Gates, *Animation of Fish Swimming*, TR-2001-19, Department of Computer Science, University of British Columbia, 2002.

[6] X. Tu and D. Terzopoulos. "Perceptual Modeling for Behavioral Animation of Fishes," In *Proceeding. Second Pacific Conference on Computer Graphics*, pp.185-200, 1994.

[7] D. Terzopoulos, B. Blumberg, P. Prusinkiewicz, C. W. Reynolds, K. Sims, and D. Thalmann, "Artificial Life for Graphics, Animation, Multimedia, and Virtual Reality," *ACM SIGGRAPH 95* Course Notes, 1995.

[8] C. W. Reynolds, "Steering behaviors for autonomous characters," In *Proceedings of Game Developers Conference 99*, pp. 763-782, 1999.

[9] D.Terzopoulos, X. Tu, and R.Grzeszczuk, "Artificial Fishes: Autonomous Locomotion, Perception, Behavior, and Learning in a Simulated Physical World," *Artificial Life*, Vol. 1(4), pp. 327-350, 1994.

[10] S. Musse and D. Thalmann, "Hierarchical Model for Real Time Simulation of Virtual Human Crowds," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 7(2), pp. 152-164, 2001.

[11] S. Musse, C. Babski, T. Capin, and D, Thalmann, "Crowd modeling in collaborative virtual environment," *ACM VRST 98*. November, pp. 115-123, 1998.

[12] X. Tu and D. Terzopoulos, "Artificial fishes: physics, locomotion, perception, behavior," *ACM SIGGRAPH 94*, July 1994.

[13] J. Funge, "Cognitive modeling for games and animation," *Communications of the ACM*, Vol. 43(7), pp. 40-48, 2000.

[14] Jong Chan Kim, Seung Il Cho, and Eung Kek Kim, "Fish Schooling Behavior Modeler for Constructing Virtual Underwater World," *In Proceedings of The 33rd KISS Fall Conference*, pp. 158-161, 2006.

Jong-Chan Kim

He received his B.S. degree from Sunchon National University in 2000,

His M.S degree from department of computer science, Sunchon National University in 2002,

His Ph.D. degree from department of computer science, Sunchon National University in 2007.

His current research interests are image processing, computer graphics, games and HCI.

Seung-Il Cho

He reveived his B.S degree from Chosun University in 1992,
His M.S degree from department of computer education, Sunchon National University in 2001,
His Ph.D degree from department of computer science, Sunchon National University in 2007.
His current research interests are computer graphics, multimedia and image processing.

Kim Chee-yong

He received the Ph.D. degree in Computational Physics from Inje University,
Republic of Korea, in 2000.
He worked at Inje University, Busan College from 2000~2003 and at Dongseo University from 2003-2006. In 2006, he joined dept. of Visual Information Engineering the Dong-Eui University, where he is an assistance professor.
His research interests are in 3D Animation, Multimedia Design and VR · Web 3D Contents Design.

Eung-Kon Kim

He received B.E. degree Chosun University in 1980,
His M.E. degree from department of computer engineering, Hanyang University in 1986,
His Ph.D degree form department of computer engineering, Chosun University in 1994,
In 2007, He joined department of computer science, the Sunchon National University, where he is a professor.
His current research interests are image processing, computer graphics, multimedia and HCI.