

# 가상해저공간에서 Artificial Fish의 사실적인 행동 생성을 위한 운동학적 접근

## Kinematic Access For Generation of Realistic Behavior of Artificial Fish in Virtual Marine World

김종한, 정승문, 신민우, 강임철  
동신대학교 디지털콘텐츠협동연구센터

Chong-Han Kim(askalan@naver.com), Seung-Moon Jung(jsm@dsu.ac.kr),  
Min-Woo Shin(mwshin@dsu.ac.kr), Im-Chul Kang(softkang@dsu.ac.kr)

### 요약

실시간으로 렌더링 되는 인터랙티브한 3D 가상공간에서 객체들은 일정한 조건을 고려한 이벤트 발생에 따라 상호작용이 가능하다. 그러나 이러한 상호작용의 행동 표현을 위해서는 이벤트의 조건과 행동 패턴들이 거의 일대일 대응되어야 한다. 이는 미리 규정한 패턴 수가 미흡할 때 사실적인 행동을 하지 못하며 다양한 행동패턴의 표현을 위해서는 그 만큼 고려되어야하는 이벤트의 조건이 늘어나게 되므로 전체 시스템의 복잡도를 증가시키는 문제점을 야기한다. 따라서 본 논문에서는 탄성 모멘텀을 이용한 물리학적 접근 방법을 적용하고 가변형 다중 센서를 이용하여 Artificial Fish의 움직임을 보다 사실적으로 표현할 수 있고 하나의 회피 이벤트에 대해 다양한 행동 패턴 생성이 가능한 새로운 방법을 제시한다.

■ 중심어 : | 가상현실 | Artificial Fish | 센서 시스템 | 탄성 모멘텀 |

### Abstract

The objects real time rendered in the 3D cyber space can interact with each others according to the events which are happened when satisfying some conditions. But to representing the behaviors with these interactions, too many event conditions are considered because each behavior pattern and event must be corresponded in a one-to-one ratio. It leads to problems which increase the system complexity. So, in this paper, we try to physical method based on elasticity force for representing more realistic behaviors of AI fish and apply to the deformable multi-detection sensor, so we suggest the new method which can create the various behavior patterns responding to one evasion event.

■ keyword : | Virtual Reality | Artificial Fish | Sensory System | Elasticity Momentum |

## I. 서론

디지털 크리처(Digital Creature)란 실세계의 동, 식물 등의 유기체들 혹은 가상의 생물들을 디지털로 복원한

캐릭터를 말하며 영상이나 게임 등과 같은 콘텐츠에 라이브러리 형태로 활용이 된다. 3D 가상 세계에서 디지털 크리처의 자연스러운 행동은 이것을 접하는 사람들에게 심미적 만족감을 준다. 따라서 자연스러운 디지털

접수번호 : #071025-001

접수일자 : 2007년 10월 25일

심사완료일 : 2008년 01월 14일

교신저자 : 김종한, e-mail : askalan@naver.com

크리처의 행동은 사용자가 요구하는 높은 그래픽 품질과 함께 적용되어야 하는 필수적인 요소이다.

Artificial Fish는 컴퓨터 애니메이션이나 컴퓨터 게임과 같은 인터랙티브 미디어에서 사용되는 자동적인 에이전트의 형태로 변환되는 디지털 크리처의 한 종류이다. 이것은 한정적 논리 규칙에 의해 생성되어진 데이터에 의해 행동이 제어되고 그 규칙에 의해 서로간의 동적인 의사소통 규칙들과 알고리즘들을 형성한다. 이를 통해 생성되는 자동적인 행동의 복잡성은 초기 Artificial Fish에 주는 인공적인 변수와 해저에서 작용하는 힘을 기반으로 환경변수와와의 연관관계를 표현한 정도에 따라 제약된다.

해양 생물의 표현을 위해 물리학에 근간한 모델링과 움직임의 패턴 등에 대한 많은 연구가 진행되어 왔다. 최초 Artificial Fish는 Tu[1]와 Terzopoulos[3]에 의해 창조되었고 동적 생체 근육 운동 모델과 실사수준의 텍스처 매핑, 다른 객체를 감지할 수 있는 센서, 선택 메커니즘에 기반을 둔 결정 트리 등을 통한 연구로 사실적이고 인공지능을 포함하는 Artificial Fish를 만들었다.

Tu는 이벤트에 따라 모션 메커니즘의 조합을 통해 행동을 표현하였고 Artificial Fish에 Intention 속성을 주어 Fear, Hunger, Libido에 따른 모션들을 조합하여 행동으로 표현하였다. 각 속성이 주어지면, 물고기가 사실적으로 움직이도록 근육 시스템을 이용하여 상황에 맞는 행동을 연출하게 된다. 그러나 환경의 요인에 의해 결정되어 지는 변수가 고려되어지는 부분이 취약하여 동일한 이벤트에 대해 같은 동작을 반복하게 되는 한계를 가지고 있다. 포식자와 피식자의 거리 또는 물고기의 무게와 접근 속도 등과 같은 변수들이 고려되지 않고 감지 센서와 물고기의 상태에 따라 행동하게 된다. 이것은 동일한 조건에는 동일한 행동을 유발하는 결과를 낳는다.

본 논문에서는 더욱 사실적인 행동을 표현하기 위해 환경 변수를 고려하여 물리학 기반의 행동 생성 기법을 제안한다. 이 방법은 Artificial Fish가 동일한 입력 조건과 상태 속성에서도 주어지는 환경 변수에 의해 다른 행동을 유발한다. 이를 위해 초기 추진력과 방향을 정의하고 환경 변수로 물리력을 적용하여 피식자와 포식

자의 방향 벡터 및 회피 방향을 결정하고, 탄성력을 통해 둘 사이의 접근거리 및 회피 속도를 결정한다.

## II. 기존 Artificial Fish 기본 요소

### 1. 모델링

초기 Artificial Fish를 디자인 한 Tu는 탄성력을 지닌 Spring-mass 모델에 기반을 둔 Fish를 제안하였다. 이 물리력에 기반을 둔 Fish 모델은 23개의 포인트 매스와 91개의 스프링으로 이루어져 있고 이 스프링들을 "Muscles"라고 칭하였다. 물고기는 이 스프링의 유효 길이(rest length)를 변화시킴으로써 움직임을 생성한다. [그림 1]은 Tu가 제안한 동적 물고기 모델이다.

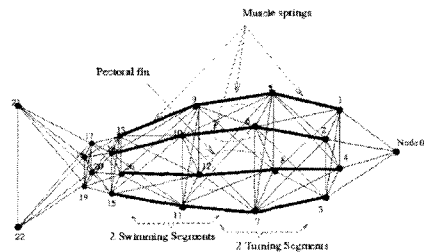


그림 1. Dynamic Fish Model

### 2. 센서 시스템

물고기들은 많은 탐지 능력을 가지고 있다. 대표적으로 청각과 같은 기계적 감지력, 시각적 감지력, 맛이나 냄새와 같은 화학적 감지력 3가지로 구분된다. 마찬가지로 Artificial Fish가 해저 가상공간 내에서 다른 동적, 정적 객체들을 탐지하기 위해서는 센서 시스템이 필요하다. 센서 시스템이란 Artificial Fish의 일정거리에서 반응하는 객체들을 탐지하여 그 대상에 따라 이벤트를 주기위한 시스템이다. Reynolds는 물고기의 머리 쪽에서 발사되는 광선을 이용한 라이트레이싱 방법을 통해 다른 물고기나 장애물들을 감지한다. 이 방법은 물고기의 머리 쪽에 있는 객체만 감지하는 원초적인 방법이다. [그림 2-1]은 Reynolds가 제안한 라이트레이싱 센서 시스템이다. 이를 보완하여 Tu는 위의3가지 대표적

감지력 중 비전을 이용하여 300도의 시야각을 가지는 센서시스템을 제안하였다. 그림 2-2는 Tu가 제시한 시각 센서 모델이다.  $V_v$ 는 뷰 볼륨의 반경을 나타내고 확장된 감지 센서의 반경은  $O_v$ 로 나타내었다.  $V_v$ 와  $O_v$ 는 각 물고기  $i$ 의 애니메이션 내에서 크기 파라미터  $S_i^p$ 에 영향을 받는다. 즉  $V_v$ 는  $S_i^p$ 에 비례한다.

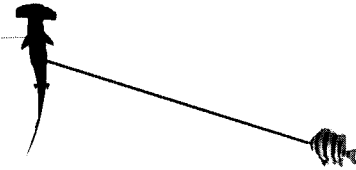


그림 2-1. Reynolds가 제안한 Ray-Tracing방법

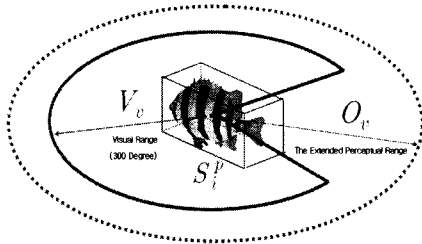


그림 2-2. Tu가 제안한 Sensory System

### 3. Artificial Fish 모션 생성과 행동 패턴 알고리즘

Grey는 물속의 중심축으로 물고기 꼬리의 수직 속도를 통해 물고기의 속도를 나타낼 수 있는 공식을 제공하였다. 식1은 물고기의 꼬리 속도에 따른 진행 속도를 나타낸 것이다.

$$(V_{fish}/V_{tail}) = (2\pi a^2/\lambda^2)[1/(1 + [4\pi a^2/\lambda^2])] \quad (1)$$

식 1에서  $a$ 는 꼬리의 진폭  $\lambda$ 는 파동의 길이  $V_{fish}$ 와  $V_{tail}$ 은 각각 물고기의 진행 속도와 꼬리의 전파 속도를 나타낸다. Artificial Fish는 포식자와 피식자, 그리고 양쪽에도 포함되지 않는 중립자(pacifist)로 나뉜다. 포식자는 먹이를 사냥하고 피식자는 포식자의 추적을 회피하는 행동을 한다. 포식자와 피식자는 센서에 반응하여 행동이 이루어진다. 감지 센서에 어떠한 입력값도 없는 경우 각 객체들은 Wander행동을 취한다. 포식자

의 경우 센서 반경 내에 피식자가 탐지될 경우 Pursuit 행동을 한다. 상대적으로 센서의 반경이 작은 피식자의 경우 일정 거리 내에 포식자가 나타날 경우 Flee 행동을 한다. [표 1]은 각 행동에 대한 설명이다.

표 1. Artificial Fish의 행동

Wander	먹이를 찾거나 목적없이 가상공간을 랜덤하게 헤엄치는 행위
Pursuit	피식자를 향해 속도를 증가시켜 추적
Flee	포식자를 피해 도망치는 행위

### 4. 기존 연구의 문제점 및 개선책

초기 Artificial Fish는 포식자와 피식자 관계에서 피식자는 포식자를 감지했을 때, 센서감지, 각 객체에 맞게 정의된 행동 선택, 회피행동의 순서로 설계되었다. 이는 각 객체마다 동일한 이벤트(센서의 감지)에 의해 동일한 행동을 유발하여 단순로운 행동을 표현할 수밖에 없는 한계를 지니고 있다. 이를 개선하기 위한 기존의 연구로는 피식자의 센서가 포식자를 감지했을 때 피식자와 포식자의 상태인 속도, 무게 등과 환경 상태 유속, 마찰력, 부력 등을 고려하여 행동 패턴을 하게 유발하는 방법이 있다. 그러나 이것 또한 행동을 유발하기 위해서는 각 입력값에 맞는 출력값을 미리 생성해야만 함으로 시스템의 설계 및 구현 시 복잡도가 증가하는 단점이 있다. [그림 3]은 모든 입력 파라미터를 고려한 상황에서 포식자가 피식자를 탐지했을 때의 알고리즘과 탄성 모멘텀을 적용했을 때의 알고리즘을 비교한 것이다.

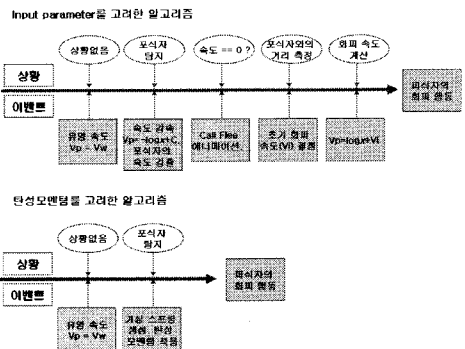


그림 3. 입력변수를 고려한 알고리즘과 탄성 모멘텀을 생성한 알고리즘의 비교

그러나 기존 알고리즘에서 고려해야 될 입력파라미터 값을 탄성 모멘텀을 이용한 물리식을 이용한 통합하여 복잡도를 감소시킬 수 있다. 또한 기존의 센서 시스템에서는 하나의 센서로 포식자와 동적객체와 장애물과 같은 정적 객체를 탐지하여 피식자가 회피하는 공간이 한정적이었다. 가변형 다중 센서를 통해 실제 물고기가 느끼는 배고픔 등의 정보를 통해 센서의 크기가 변화하도록 설계하고 다른 물고기 객체와 바위와 같은 장애물을 각각 달리 탐지할 수 있는 센서를 두어 피식자가 포식자 탐지 시 바위에 접근하여 회피할 수 있게 하는 등의 사실적인 행동을 유발시킬 수 있다. 그림 4는 기존의 단순 행동 유발 알고리즘과 모든 입력변수를 고려한 행동 유발 알고리즘과 본 논문에서 제시한 알고리즘을 비교한 것이다. 초기 탐지 후 회피에서 단순한 행동패턴이 문제점으로 지적되었고 탐지 후 입력 파라미터 분석 및 입력값 따른 출력값 생성에서는 각 물고기마다 접근속도, 접근거리, 초기 회피속도 등을 미리 정의해 놓고 입력값의 조합에 맞게 불러오는 방법을 취한다. 이는 설계의 복잡성 및 구현의 어려움을 불러온다. 본 논문에서 제시한 탄성 모멘텀을 이용한 방법은 이러한 입력값, 출력값 사이의 상관관계를 단순화하여 출력값을 바로 수치적으로 이용하여 위의 접근속도, 접근거리, 초기 회피속도 등의 바로 결정한다.

### III. Artificial Fish의 센서 시스템의 개선 및 자동행동 유발 시스템 설계

사실적인 가상 해저 공간의 구축을 위해 피식자와 포식자는 가변적인 센서를 가지게 된다. 객체의 크기, 배고픔의 정도에 따라 센서의 크기를 변화시킴으로써 더욱 사실적인 행동을 유발할 수 있다. 또한 3D 가상 공간내의 동적 객체에 작용하는 힘은 물고기의 추진력, 부력, 중력, 물의 점성에 의한 마찰력 등이 있다. 이는 피식자가 포식자를 발견했을 때 자연스러운 움직임을 표현하기 위한 중요한 파라미터들이 된다. 빠른 스피드로 접근하는 두 객체는 회피 직전의 접근거리가 더 짧아지고, 천천히 움직이는 객체의 경우 센서의 크기에 해당하는 거리에서 바로 회피 행동을 취하게 한다. 본 논문에서는 이러한 기본적인 힘에 자연스러운 행동을 위한 탄성 모멘텀을 생성시킨다. 탄성 모멘텀은 피식자와 포식자 사이의 가상의 스프링으로 생성된다. 동적 객체의 추진력에 의한 관성에 의해 스프링은 수축하다가 임계점을 벗어나면 반발하게 된다. 이는 빠른 속도로 다가가는 객체나 느린 속도로 가다가는 객체에 대해서도 다른 탄성력을 주고 물고기의 무게나 크기에 의해 회피 속도를 하게 생성한다. 또한 탄성 계수의 조절을 통해 사실적인 물고기의 행동이 가능하게 된다.

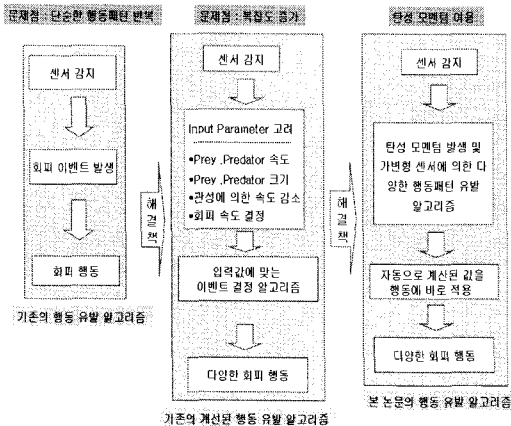


그림 4. 기존의 행동 유발 알고리즘과 개선된 알고리즘의 비교

#### 1. 가변형 센서와 회피방향 결정 시스템

##### 1.1 가변적 다중 센서

현실세계에서 물고기는 자신의 상태에 따라 먹이를 탐지하는 능력을 조절한다. 이를 위해 본 논문에서는 센서를 가변적으로 설계하고 초기 추적 속도 및 회피속도를 다르게 설정하였다.

시간이 지날수록 배고픔을 더 느끼게 되는 포식자는 탐지 센서의 크기를 확장 시키고 높은 초기 추적 속도 값을 갖게 된다. 또한 피식자는 포식자를 탐지하는 센서와 바위나 해초 등과 같은 일반 장애물을 탐지하는 센서를 따로 두었다. 그림 5는 Artificial Fish의 탐지 센서를 나타낸 것이다.  $L_f$ 는 먹이 탐지 센서 또는 포식자 탐지 센서로 객체의 상태에 따라 가변적이고,  $L_o$ 는

장애물 탐지 센서로 Wander, Flee, Pursuit 상태에서 바 닥, 주변의 바위 등을 탐지하는 센서이다. 장애물 탐지 센서가 포식자/피식자 탐지 센서보다 작게 설계한 이유는 포식자가 피식자를 쫓거나 피식자가 포식자에게 쫓기는 상황에서 피할 수 있는 공간을 최대한 확보하기 위한 것이다.

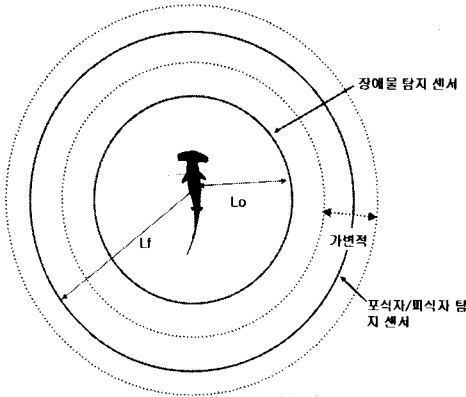


그림 5. 가변적 다중 센서

1.2 회피방향 결정 알고리즘

포식자와 피식자의 센서가 각각 서로에 대해 반응했을 때 피식자는 속도  $V_{evasion}$  로 회피를 하고 도망치는 방향  $D_{evastion}$  은 포식자와 주위의 장애물을 고려하여 결정된다.

피식자 센서 내에 장애물이 없는 경우 피식자는 포식자의 같은 방향을 가진다. 센서의 범위 내에 장애물 또는 또 다른 포식자가 있는 경우에는 다음과 같이 피식자의 방향이 결정되어 진다. 제 1 포식자의 위치를  $P_{predator}$  라고 하고 제 2 포식자 (또는 장애물)에 대한 위치를  $P_{obstacle}$  으로 한다.

$$P_{predator} = (x_{predator}, y_{predator}, z_{predator})$$

$$P_{obstacle} = (x_{obstacle}, y_{obstacle}, z_{obstacle})$$

$$P_{prey} = (x_{prey}, y_{prey}, z_{prey})$$

이때 피식자의 위치가  $P_{prey}$  라고 하면 피식자에 대한

포식자의 방향은  $\overrightarrow{P_{predator} - P_{prey}}$  가 되고 피식자에 대한 장애물의 방향은  $\overrightarrow{P_{obstacle} - P_{prey}}$  가 된다. 피식자의 회피 방향은 이 두 방향벡터의 합을 반대방향인  $D_{evastion}$  이 된다.

$$D_{evastion} = (-x_{predator} + x_{obstacle} - 2x_{prey}), -(y_{predator} + y_{obstacle} - 2y_{prey}), -(z_{predator} + z_{obstacle} - 2z_{prey})$$

그림 6은 피식자의 포식자와 장애물에 대한 회피 방향을 나타낸다.

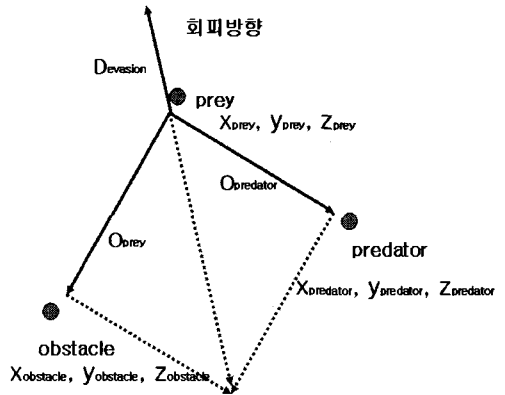


그림 6. 회피방향 벡터

2. 탄성 모멘텀을 적용한 행동 제어

3D 가상 공간내의 동적 객체에 작용하는 힘을 물고기의 추진력, 부력, 중력, 물의 점성에 의한 마찰력 등이 있다. 본 논문에서는 이러한 기본적인 힘에 자연스러운 행동을 위한 탄성 모멘트를 이용한다. 탄성 모멘트는 피식자와 포식자 사이의 가상의 스프링으로 생성된다. 동적 객체의 추진력에 의한 관성에 의해 스프링은 수축하다가 임계점을 벗어나면 반발하게 된다. 이는 빠른 속도로 다가가는 객체나 느린 속도로 가다가는 객체에 대해 서로 다른 탄성력을 주고 물고기의 무게나 크기에 의해 회피 속도를 하게 생성한다. 또한 탄성 계수의 조절을 통해 사실적인 물고기의 행동이 가능하게 된다.

2.1 관성력(추진력), 점성에 의한 마찰력

부력과 중력은 서로 힘의 평형을 이룬다는 가정 하에 물고기가 앞으로 나아가는 힘, 즉 관성력은 다음과 같이 정의된다.

$$F_{prey-prop} = \frac{d}{dt}(m_{prey}v_{prey}) = m_{prey}a_{prey} + \frac{dm_{prey}}{dt}v_{prey} \quad (2)$$

- $F_{prey}$  = 피식자의 관성력
- $a_{prey}$  = 피식자의 가속도
- $m_{prey}$  = 피식자의 무게
- $v_{prey}$  = 피식자의 속도

그러나 피식자와 포식자가 정면으로 충돌하는 것이 아니라 각기 다른 방향으로 움직이고 있으므로 피식자의 포식자 상대 속도를 구하여야 한다.

상대속도  $v_{p/p} = v_{prey} - v_{predator}$  가 된다.

그러나 여기서는 스칼라량만 고려하므로  $v_{p/p} = |v_{prey} \times \cos \theta - v_{predator}|$  가 된다. [그림 6]은 피식자와 포식자 간의 상대속도를 나타낸 그림이다.

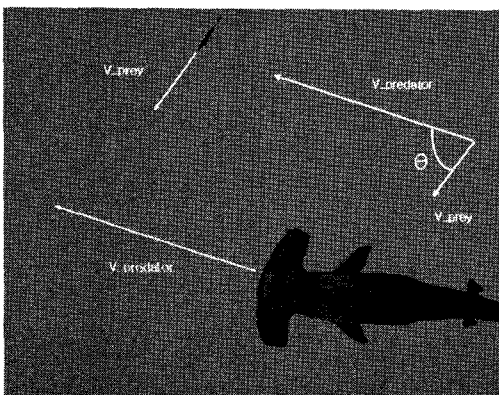


그림 6. 피식자의 포식자에 대한 상대속도

전체 관성력을 구하는데 있어서 고려해야 할 또 하나의 환경변수로는 물의 점성에 의한 마찰력이다. 마찰력은 추진력에 반대되는 방향으로 작용하며 그 힘의 크기

$F_{prey-fric}$  은 식3 과 같이 정의된다. 여기서  $c$ 는 비례상수인 스칼라 값이다.

$$F_{prey-fric} = -F_{dissipative} \frac{v_{p/p}}{|v|} = -c(|v|) \frac{v_{p/p}}{|v|} = -cv_{p/p} \quad (3)$$

전체 힘은 관성력에서 점성에 의한 마찰력을 뺀  $F_{positive}$  이다.

$$F_{positive} = F_{prey-prop} - F_{prey-fric} = m_{prey}a_{prey} - cv_{p/p} \quad (4)$$

2.2 탄성 모멘트 적용을 위한 가상 스프링 생성

센서에 반응한 피식자의 경우 바로 이벤트를 받아 회피 행동을 취하게 된다. 그러나 센서는 탄성 스프링을 생성하는 이벤트를 발생시키고 관성력과 탄성력의 평형을 이루기 위해 일정 거리인  $x$ 만큼 움직인 후  $x$  최대값을 갖게 되면 회피 이벤트를 보여준다.

이때 탄성력에 대한 힘을 운동 에너지로 변환하여 속도에 반영한다. [그림 7]은 두 객체 사이에 생성된 가상의 스프링을 보여준다.

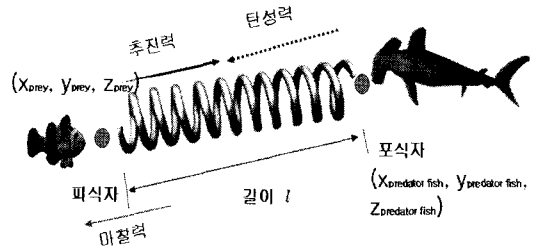


그림 7. 가상 스프링 생성

포식자의 위치를  $P_{predator}$  라 하고 피식자의 위치를  $P_{prey}$  라 하면 피식자와 포식자 사이의 거리는  $l$ 이 된다.

$$P_{predator} = (x_{predator}, y_{predator}, z_{predator})$$

$$P_{prey} = (x_{prey}, y_{prey}, z_{prey})$$

$$l = \sqrt{(x_{predator} - x_{prey})^2 + (y_{predator} - y_{prey})^2 + (z_{predator} - z_{prey})^2}$$

피식자와 포식자의 센서의 서로 감지상태가 True가 되면 두 객체 사이에 길이가  $l$ 이고 탄성계수가  $k$ 인 스프링이 생성된다. 이 스프링의 탄성 퍼텐셜 에너지  $V_e$ 는 식 5와 같다.

$$V_e = \int_0^x F dx = \int_0^x kx dx \quad (5)$$

피식자의 센서 길이를 스프링의 길이와 같이 놓지 않는 이유는 장애물로 인해 어느 순간에는 센서의 길이보다 더 작은 스프링 길이가 생성 될 수 있기 때문이다.

$F$ 는 물고기에 의한 추진력이고  $k$ 는 스프링 상수  $x$ 는 변형률을 나타낸다.

여기서 포식자의 크기는 피식자에 무한히 크다고 가정한다면 포식자가 가하는 압축은 이루어지지 않고 전체 위치만 변경하므로 고정된 격벽이라 놓는다. 그러므로 포식자의 대한 상대 추진력은 고려하지 않는다. 피식자의 추진력  $F_{positive}$ 에 대해 변형률  $x$ 가 생기는데

$\int_0^x kx dx = \frac{1}{2} kx^2$ 이므로 물고기가 스프링에 한일  $W$ 가 에너지가 모두 탄성 퍼텐셜 에너지로 바뀌었으므로 변형률  $x$ 의 최대값은  $x_{max}$ 가 된다.

$$\begin{aligned} F_{positive} &= m_{prey}a_{prey} - cv_{prey} \\ W &= (ma - cv) \times x \\ \frac{1}{2} kx^2 &= (m_{prey}a_{prey} - cv_{p/p}) \times x \\ x_{max} &= \frac{(cv_{p/p} - m_{prey}a_{prey})}{k} \end{aligned}$$

즉  $x$ 가  $x_{max}$  되었을 때 회피 이벤트를 주고, 센서 시스템에 의해 얻어진 회피 방향은  $(-(x_{predator} + x_{obstacle} - 2x_{prey}), -(y_{predator} + y_{obstacle} - 2y_{prey}), -(z_{predator} + z_{obstacle} - 2z_{prey}))$ 이 된다.

$$\frac{1}{2}mv^2 = \frac{1}{2}kx^2 \quad (6)$$

초기 회피 속도  $v_{escape}$ 는 탄성 에너지가 운동에너지로 복원된 후 속도이므로 식 6에 의해  $v_{escape}$ 을 구할 수 있다.

$$v_{escape} = \frac{\sqrt{m_{prey}k}(cv_{p/p} - m_{prey}a_{prey})}{m_{prey}k} \quad (7)$$

위 내용들을 요약하면 다음과 같다.

- ① 속도  $v_{prey}$ 로 움직이는 무게  $m_{prey}$ 인 피식자는  $v_{predator}$  속도로 움직이는 포식자를 만나면 상대속도  $v_{p/p}$ 를 갖게 된다.
- ② 피식자는 회피 명령 이벤트가 일어나면  $(x_{predator} + x_{obstacle} - 2x_{prey}), -(y_{predator} + y_{obstacle} - 2y_{prey}), -(z_{predator} + z_{obstacle} - 2z_{prey})$  방향으로 회피한다.
- ③ 일반적으로 센서 크기 포식자와의 거리  $R$ 에 접근했을 때 회피 이벤트가 일어나지만 피식자의 관성력에 의해  $R - \frac{(cv_{p/p} - m_{prey}a_{prey})}{k}$ 까지 접근한 후 회피 이벤트를 발생하게 된다.
- ④ 초기 포식자에 대한 피식자의 회피 속도는  $v_{escape}$ 이 된다.

$$v_{escape} = \frac{\sqrt{m_{prey}k}(cv_{p/p} - m_{prey}a_{prey})}{m_{prey}k}$$

#### IV. 실험 및 고찰

본 논문에서는 물고기의 관계를 포식자와 피식자로 한정한다. 포식자는 Pursuit과 Wander의 성질을 가지고 피식자는 Flee와 Wander의 성향을 가진다. 이 행동들을 포식자와 피식자 사이의 상호작용에 의해 생성되고 소멸된다. Artificial Fish의 행동을 구현하기 위해

본 논문에서는 피식자로 감성돔을, 포식자로 귀상어를 선택하여 모델링 하였다. 모션은 Flee, Wander, Pursuit등의 행동을 표현하기 위해 Swimming, Turning, Rapid Swimming등의 단위 애니메이션을 제작하여 각 상황에 맞게 취하도록 설정하였다. 표2는 센서의 반응에 따른 각 객체의 이벤트들을 나타낸다.

센서감지 여부를 나타내는 O와 X는 피식자와 포식자의 탐지 센서의 크기가 다르므로 어느 한쪽이 먼저 상대 객체를 탐지하였을 경우를 나타내기 위해 설정하였다. 피식자와 포식자가 취해야할 행동은 서로 감지하지 못했을 경우와 하고 포식자가 피식자보다 탐지 센서가 더 크므로 포식자가 먼저 발견할 경우와 포식자가 피식자를 추적한 후 피식자도 포식자를 발견할 경우 세 가지 행동을 정의하였다. 대상 객체로는 감성돔과 귀상어를 3Ds Max 8.0으로 제작하였고 본 애니메이션으로 행동을 표현하였다. VR 환경은 Virtools 4.0을 통해 구현하였다.

표 2. 센서 반응에 따른 이벤트

조건	포식자	피식자	객체 반응		
센서 감지 여부	1	X	X	포식자	Wander
				피식자	Wander
	2	O	X	포식자	Pursuit
				피식자	Wander
3	O	O	포식자	Pursuit	
			피식자	Flee	

조건	Object Behavior
센서 감지 여부	1 랜덤 Move(Wander)
	2 포식자의 Orientation 값 변동 (target : prey)
	3 포식자와 피식자의 거리값 계산을 통한 탄성 모멘텀 값 적용 피식자의 Flee 방향 결정 알고리즘 적용

[그림 8]은 감성돔의 탐지 센서를 나타낸다. 탐지 센서는 탐지 센서는 반지름이 일정한 완전 구형으로 되어 있고 객체의 크기에 비례하도록 설계하였다. 앞서 밝힌바와 같이 장애물에 대한 회피 센서보다 포식자나 피식자에 대한 탐지 센서를 크게 구현하였다.

장애물 탐지 센서      포식자 탐지 센서

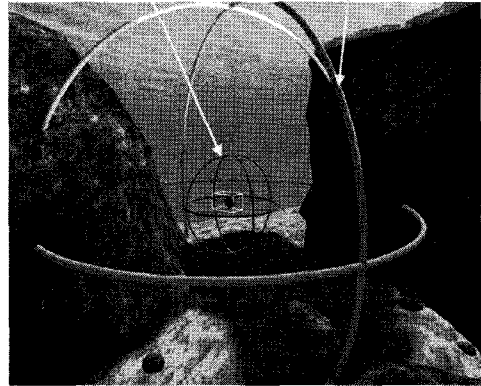


그림 8. 가변형 다중 탐지 센서

[그림 9]는 피식자가 포식자를 탐지하고 회피방향을 결정하는 행동을 나타낸 것이다.

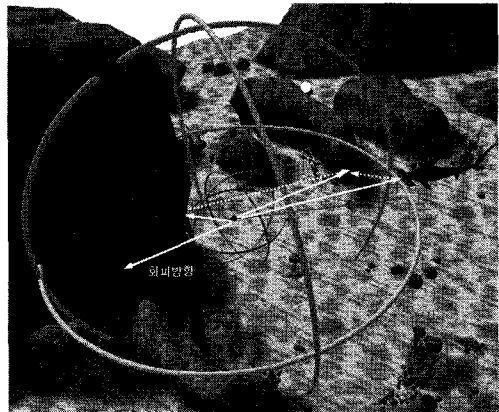


그림 9. 피식자의 회피 행동

[그림 10-1]과 [그림 10-2]는 기존의 탄성 모멘텀을 적용하지 않는 Fish와 탄성 모멘텀을 적용한 Fish의 속도 변화 및 방향을 나타낸 그림이다. [그림 10-1]의 그래프는 시간  $t$ 에 대한 피식자의 속도의 변화를 나타내며  $t_6$ 일때 포식자를 감지하여 바로 방향 전환 및 회피 속도로 전환한다. 이때 애니메이션이 자연스럽게 연결되지 않는다. [그림 10-2]는  $t_{4.5}$ 에서 피식자가 포식자를 탐지한 후 관성력에 반하는 탄성력에 의해 속도가 감쇄되고 기존의 Fish 보다  $x$ 만큼 더 포식자에게 근접



한다.  $x$ 가 최대값을 가질 때 속도는 최소값인 0을 갖고 반대의 회피방향을 취한다. 이때 탄성력은 최대값을 갖는다. 회피속도는 유영 속도보다 빠른 값으로 정의하였으며 같은 값을 갖게 되지만 자연스러운 행동 표현을 위해 식8과 같이 물고기에 따른 능력치 상수값  $C_{porgy}$ 를 더하였다.

$$v_{escape} = \frac{\sqrt{m_{prey}k(a_{prey})}}{k} + C_{porgy} \quad (8)$$

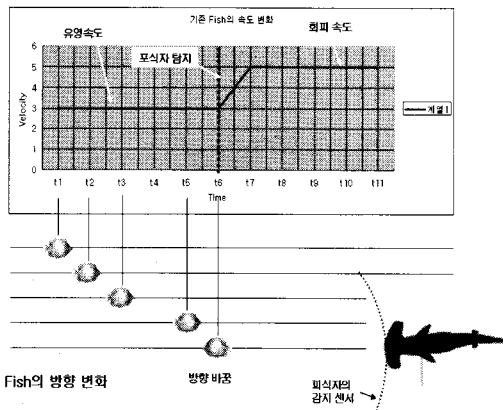


그림 10-1. 탄성 모멘텀을 적용하지 않는 피식자의 속도 및 방향 변화

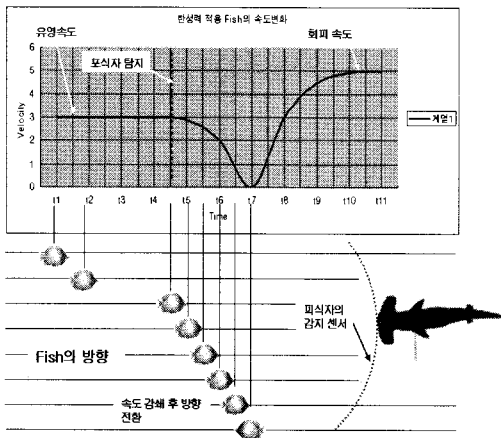


그림 10-2. 탄성 모멘텀을 적용한 피식자의 속도 및 방향 변화

<http://blog.naver.com/askalan>에서 기본 회피와, 탄성 모멘텀 적용, 회피방향 결정 알고리즘을 적용한 구현 결과를 볼 수 있다.

## V. 결론

본 논문에서는 실시간으로 렌더링 되는 3D 가상공간에서의 Artificial Fish의 자연스러운 행동을 생성하는 기법을 제시하였다. 이는 정해진 이벤트에 대해 반응하는 일대일 대응 규칙의 한계성에서 벗어나 대상 객체의 속도와 무게 등의 환경 변수에 의해 반응하는 행동이 실시간으로 계산되어 각기 다른 행동을 하게 된다. 또한 가변형 다중 탐지 센서를 적용하여 현실 세계와 비슷한 현상을 나타내도록 설계하였다. 이를 위해 회피방향 결정 방법과 포식자와 포식자의 접근 거리 계산 방법, 회피 속도 결정 방법을 제시하였다. 디지털 크리처의 사실적인 행동표현은 섬세한 모션 제어에서 시작된다. 본 논문은 주어진 변수에 대해서만 실시간으로 계산하여 정형화된 틀에 따라 다음 행동을 처리하는 방법을 제시한 것이다. 따라서 더욱 적은 Artificial Fish의 생성을 위해서는 유속이나 온도에 따른 물의 점도의 변화 등의 환경 변수뿐만 아니라 객체에 학습을 상태에 이용한 향상된 인공지능이 필요할 것이다.

## 참고 문헌

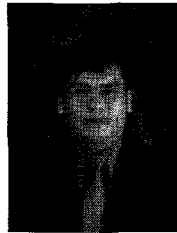
- [1] D. Terzopoulos, X. Tu, and R. Grezeszczuk, "Artificial Fishes," Published in Artificial life, Vol.1, No.4, pp.327-351, 1994.
- [2] K. Stephens, B. Pham, and A. Wardhani, "Modelling Fish Behaviour," Proceedings of the 1st international conference on Computer graphics and interactive techniques in Australia and South East Asia, Vol.11, No.14, pp.71-78, 2003(2).
- [3] G. S. Helfman, B. B. Collette, and D. E. Facey,

*The Diversity of Fishes 3rd edition.*, Blackwell Science, 1999.

- [4] J. Grey, *Animal Locomotion*, Weidenfeld and Nicolson, 1968.
- [5] A. Twibell and A. Watt, "Using Programmable Hardware to Render and Animate Fish," Undergraduate project dissertation, The University of Sheffield, UK, 2005.
- [6] C. Reynolds, "Flocks, Herds, and Schools: A Distributed behavioral model," *Computer Graphics*, Vol.21, No.4, pp.25-34, 1987.
- [7] F. P. Beer, *Vector Mechanics for Engineers 2nd edition*, McGraw Hill, 1990.
- [8] S. Goren and F.J. Ferguson, "CHESMIN: A Heuristic for state Reduction in Incompletely Specified Finite State Machines," *Proceeding of 2002 Design, Automation, and Test in Europe Conf., and Exhibition*, pp.248-254, 2002.
- [9] E. M. Clarke, O. Grumberg, and D. E. Long, "Model Checking and Abstraction," *ACM Transactions on Programming Language and Systems*, Vol.16, No.5, pp.1512-1542, 1994.

정 승 문(Seung-Moon Jeong)

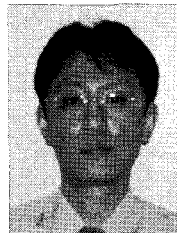
정회원



- 1991년 2월: 동신대학교 컴퓨터 공학과(공학사)
  - 1999년 2월: 동신대학교 컴퓨터 공학과(공학사)
  - 2004년 2월: 동신대학교 컴퓨터 공학과(공학박사)
  - 2005년 2월 ~ 현재: 동신대학교 교수
- <관심분야> : 디지털 콘텐츠, 3D 애니메이션, 이미지 프로세싱, DRM, 유비쿼터스 컴퓨팅, VR

신 민 우(Min-Woo Shin)

정회원



- 1990년 2월: 전남대학교 전산통계학과(이학사)
  - 1992년 2월: 전남대학교 전산통계학과(이학석사)
  - 2002년 2월: 전남대학교 전산통계학과(이학박사)
- <관심분야> : 컴퓨터이론, 입체영상처리, 가상현실, 인공지능

저 자 소개

김 종 한(Chong-Han Kim)

정회원



- 1999년 8월: 전남대학교 기계공학과(공학사)
- 2004년 2월: 전남대학교 소프트웨어 협동과정(공학석사)
- 2004년 8월 ~ 현재: 동신대학교 디지털콘텐츠협동연구센터

선임연구원, 전남대학교 전산학과 박사과정

<관심분야> : 디지털 크리쳐, 가상현실, 인공지능, 정형기법

강 임 철(Im-Chul Kang)

정회원



- 1991년 2월: 전남대학교 전산통계학과(이학사)
- 1997년 2월: 전남대학교 경영학과(경영학석사)
- 2005년 2월: 전남대학교 전자상거래협동과정(경영학박사)

<관심분야> : 전자상거래, 디지털콘텐츠, 3D 애니메이션, 가상현실