Quest3D 기반 해상부표 동적안정성의 시각적 표현 프로그램 설계 및 구현에 관한 연구

이서정* · 이재욱** · † 국승기

*, ** 한국해양대학교 IT공학부, † 한국해양대학교 해양경찰학과

A Research of Design and Implementation of Visual Program to Displaying External Factors of Marine Buoy using Quest3D

Seo-Jeong Lee* · Jae-Wook Lee** · † Seung-Gi Gug

*, ** Division of IT Engineering, Korea Maritime University, Busan 606-791, Korea

† Department of Coast Guard Studies, Korea Maritime University, Busan 606-791, Korea

요 약 : 선박의 대형화와 고속화에 따라, 발전된 안전운항 기술에 대한 요구가 최근 크게 늘어나고 있으며, 그 중, 해상 부표는 교통 안전정보를 제공하는 해상교통 분야의 주요 장비이다. 해상부표를 새롭게 설계하거나 설치 시, 안전성과 견고함을 판단하기 위해서 사전 검증은 반드시 필요한 과정이다. 본 논문은 새로운 해상 부표의 설계에 대해, 외부 환경 요소의 변화에 따른 설계물의 반응을 컴퓨터상에서 미리 보여주기 위한 프로그램을 구현했다. 즉, 외부 영향요소 즉, 바람, 파도 및 조류 값의 변화에 따라 설계물이 반응하는 결과를 시각적으로 보여준다. 특히, 영향요소 및 기능의 확장을 고려하여, 프로그램 내부 설계에 소프트웨어적 부품 개념을 도입한 아키텍처를 구성했다. 본 논문에서 지향하는 시각적 프로그래밍과 소프트웨어적 부품개념을 동시에 지원할 수 있는 그래픽 툴로써 Quest3D를 이용했다.

핵심용어 : 해상부표, 외부영향요소, Quest3D, 시각적 표현, 프로그램 설계

Abstract : As vessels are faster and bigger, higher safety navigation techniques have been required. In marine traffic area, buoy is to be one of the most important equipment, so that it should be built stable and robust. Verification in design level is good to accomplish it. This paper tries to implement visual program to confirm the external stability, which shows the movement on water by external factors such as wind, tide and current. Considering further requirements for additional influences or functions, design architecture for program introduces the concept of software component. Using Quest3D as the graphic tool, visual programming with software component concept can be implemented.

Key words : marine buoy, external factors, Quest3D, visual presentation, program design

1. 서 론

해운·수산 등 해양관련 산업의 발전에 따라 연안의 해상교통량은 지속적으로 증가하고 있으며, 선박의 대형화 · 고속화 등으로 급격하게 변화하고 있다. 이러한 변화에 따라 선박의 항로 통항시 안전운항을 도모하기 위해서 선박의 안전운항 기술과 더불어 안전한 정보를 제공할 수 있는 해상교통시설물 구축이 필수적이다(국토해양부, 2010).

해상교통시설은 항로표지를 항로표지법으로 정의한 명칭이며, 위치를 표시하는 수단에 따라 크게 광파표지, 형상표지, 음파표지, 전파표지, 특수신호표지로 구분한다(김, 2011).

이러한 항로표지 시설의 개발은 환경여건 분석, 분석에 따

른 설계, 설계를 준수하는 제작의 과정으로 구성되며, 설계와 제작 과정 사이에 설계 결과물에 대한 검증이 필요하다.

본 논문에서는 이러한 항로표지 시설의 설계과정에서 외부 환경 여건에 대한 검증을 위해, 그래픽 기술을 활용하여 시각적으로 표현이 가능하고 또한 확장이 용이한 프로그램을 설계하고 구현해 보았다. 이러한 소프트웨어 부품개념을 지원하면서 시각적 프로그래밍이 용이한 툴로써, Act-3D의 Quest3D제품을 활용했다.

2장에는 Quest3D를 소개하고, 3장에서 Quest3D를 이용한 프로그램 설계에 대해 설명하고 4장에서 구현결과를 보여준다.

* 정회원, sjlee@hhu.ac.kr 051)410-4578

** 연회원, wookis6766@naver.com 051)410-4888

† 교신저자, cooksg@hhu.ac.kr 051)410-4227

2. Quest3D 소개

2.1 Quest3D 특징

Quest3D는 Act-3D사에서 개발한 개발 환경으로, 시물레이션이나 가상현실 같은 실시간 3D 그래픽 소프트웨어를 쉽게 구현할 수 있도록 도와준다(Act-3D, 2009).

Quest3D프로그래밍에 사용되는 모든 객체는 채널(Channel)이라는 형태로 제공하며, 각 채널을 그래프의 형태로 서로 링크하여 하나의 프로그램을 구축할 수 있다. 모든 과정은 위지윅(WYSIWYG, What You See Is What You Get)방식으로 진행되며, 프로그램 컴파일 과정 없이 실시간으로 결과 확인이 가능하다. 이러한 특징과 사용의 편리함으로 게임분야를 비롯해서 건설, 환경, 부품조립 등의 분야에서 시물레이션이나 가상체험용 프로그램 개발에 사용되고 있다(전, 2010;이, 2009).

2.2 Quest3D 작동 메카니즘

Quest3D에서는 일반적인 프로그래밍과는 다르게 일체의 코드 없이 개발 가능한 채널 시스템이라 부르는 시각적 개발 방법을 제공한다. Fig. 1은 카메라와 연동된 3D 렌더링(rendering)을 위한 채널 시스템의 예를 보여준다.

하나의 채널은 소리, 텍스처, 특정 동작 실행 명령이나 작은 정보의 조각으로서 역할을 한다. 각각의 채널은 타입이 존재하며, 그 타입 별로 기능이 부여된다. 모든 채널은 베이스 타입이 존재해서 같은 베이스 타입의 채널은 같은 형태(type)의 데이터를 가진다.

필요한 변수를 채널을 통해 입력받을 수 있고, 채널 사이 트리 구조로 연결하여 프로그램이 수행된다. 즉, Quest 3D에서는 채널이라 부르는 개별 부품으로 구성되며, 각 채널 내부에 필요한 동작에 대한 명세를 한다. 또한, 채널의 연결 구조를 통해 필요한 변수 값을 주고받는다.

Fig. 1 Sample of Quest3D Channel System

2.3 Quest3D 유저인터페이스

프로그래밍의 진행 과정 중에 별도의 컴파일(compile)이 아닌 컴포넌트에 대한 인터프리터(interpreter) 방식으로 실제 구

현 내용을 실시간으로 프리뷰하면서 진행할 수 있다. 인터프리터 방식은 사용자 명령이나 요구를 즉시 반영한 결과를 보여준다는 장점이 있다. 그 결과, Quest3D를 사용할 경우 다른 방식에 비해 보다 빠른 개발이 가능하다.

Fig. 2의 좌측 하단에서 채널의 배치에 따라 실시간으로 렌더링 되는 모습을 확인할 수 있다.

Fig. 2 Preview in Quest3D

2.4 Quest3D 객체지향 환경

데이터의 중복을 줄이고 다양한 사용자 요구사항 변경에 용이한 대처를 위해 도입된 객체지향 프로그래밍은 3D모델 기반 시각화 프로그래밍에도 많이 활용되고 있으며, Quest3D에서도 마찬가지로 적용하고 있다(Quest3D,2012).

특히, Quest3D에서는 소프트웨어 패턴(pattern)기술을 도입하여 프로그래머의 목적에 맞는 패턴을 활용할 수 있는 기반을 제공한다. 패턴을 이용한 프로그래밍은 안정된 확장성과 기능배분의 균형 및 재사용성의 극대화 등의 강점을 갖는다.

3. Quest3D를 이용한 프로그램 설계 및 구현

새로운 해상부표를 도입하거나 기존 해상부표의 안정성은 정적 안정성과 동적 안정성에 대한 검증을 통해 이루어진다. 본 논문은 동적 안정성의 외부 영향요소 중 일부에 대해 컴퓨터 그래픽 툴을 이용해 시각적으로 테스트한 결과를 설명한다. 테스트 결과는 기존 외부 환경요소에 대한 실제 반응 공식에 부합되는지를 적용하여 검증한다.

즉, 본 논문에서는 Quest3D를 활용하여 안정성 검증을 위한 소프트웨어 시스템 구성을 체계화하기 위해 소프트웨어 부품을 이용하는 프로그래밍 기술을 도입했다. 확장 용이한 프로그램 구조를 설계하고 구현하였으며, 해상부표의 일부 환경요소를 적용하여 검토함으로써 추후 작업을 위한 기초를 마련하였다.

3.1 Quest3D 도입 및 고려사항

최근의 그래픽 기반 프로그래밍 툴은 이전에는 복잡하고 고난이도의 기술을 요구하던 작업을 비교적 간단하게 구현할 수 있는 환경을 제공한다. 본 논문에서는 다양한 툴 중에서 Quest3D를 채택하여 실험을 시도했다. 플랫폼 적응성은 낮지만, 기능의 확장성(extensibility), 크기조절에 대한 유연성(scalability) 등에서 비교적 안정적 평가를 받고 있다(Birch, 2009).

Quest3D를 선택하더라도 객체지향 및 패턴의 활용에 대한 이해와 적용을 적절하게 프로그래밍 해야 그 효과를 극대화할 수 있다. 프로그램의 복잡도(complexity)로 설명할 수 있다.

예를 들어, 채널의 연계로 동작이 이루어지는 특징으로 다음 고려사항이 충분히 이루어져야 한다. 왜냐하면, 채널 단위나 기능 설계에 대한 고려를 하지 않은 경우, 기능 추가 시 불필요한 채널을 생성하거나 그에 따른 과도한 링크를 정의하게 되어, 시스템 규모가 커지고 복잡해질수록 성능에 차이 많이 나게 된다.

일반적으로 객체지향 시스템 복잡도에 따르는 내부 속성 중 본 논문에서는 다음 사항을 고려하였다(Singh, 2010; Sastry, 2011).

- 채널의 크기(클래스 크기) : 물리적 또는 개념적 크기를 채널간 균형있게 분배함
- 채널의 역할(함수 정의) : 채널의 역할 및 기능을 구별하여 채널 사이의 중복을 최대한 감소시킴
- 링크의 크기(함수 호출 시 변수) : 함수 호출에 수반되는 변수의 종류 및 크기 조절
- 링크의 역할(함수 호출) : 불필요한 링크를 제거하고, 중복을 감소시킴

Quest3D의 채널은 시스템에 포함된 부품 즉, 객체의 개념으로 수행되므로, 채널의 설계에 있어 기존 객체지향 프로그래밍 환경에서 객체 또는 컴포넌트를 설계하는 개념을 도입하여 프로그램을 설계 할 수 있다(Quest3D, 2012).

3.2 전체 시스템 구성요소와 현재 적용범위

본 논문에서는 해상부표의 안정성 검증을 위한 개발될 전체 시스템의 일부를 설계하고 구현했다. 개발이 완료될 전체 시스템의 주요 구성요소와 본 논문에서의 구현 및 적용 현황은 다음과 같다.

(1) 안정성 알고리즘

프로그램에서 부표의 정적 안정성 계산을 외력의 영향이 없는 환경에서 평가하는 정적안정성과 외력 영향 요소가 반영되는 동적안정성이 있다. 정적 안정성의 경우, 부력 이외의 외력을 모두 0으로 설정하여 알고리즘을 적용하고, 동적 안정성의 경우, 바람, 조류, 파도와 같은 외력이 작용하여 나타내는 운동성을 평가한다.

(2) 3D 모델

프로그램 내에서 사실적인 부표의 표현을 위해서 실제 상황에 도입 가능한 모델을 대상으로 했다. 본 논문에서는 실제 모델을 대상으로 테스트하기 위해, Quest3D에서 사용가능한 형태로 작업이 완료된 3D Max기반의 모델을 도입했다.

(3) 물(water) 모델

물의 경우, 해상부표의 외부환경 검증을 위한 주요 요소로써 실제와 흡사한 요건을 수용하는 환경을 시각적으로 구현해야 한다. 본 논문에서는 Height Field(Heightmap, 2012)를 응용해 물의 효과를 표현한 기존 모델을 활용하여 필요한 변수 값을 지정하여 사용하였다.

Height field 방식의 수면 구현을 위해서는 실시간으로 각 그리드의 주변 높이 값 변화에 따라 현재 높이 값을 수정하면 수면 구현이 가능해진다(Müller-Fischer, 2008).

(4) 외부영향 알고리즘

외부 영향 요소의 종류는 바람, 파도, 조류, 조석 4가지로 구분된다(한국선급, 2011). Table 1은 각 외부영향에 대한 세부 요소를 보여준다. 본 논문에서는 해상부표에 영향을 미치는 외부 영향요소 중 본 논문에서는 바람, 파도 그리고 조류에 대해 구현하고 테스트하였다.

Table 1 External Factors

No	외부영향 종류	세부요소
1	바람	풍향, 풍속
2	파도	파고, 파향
3	조류	조류방향, 조류세기
4	조석	수심의 변화로 정의하여 구분

각각의 외부 영향 요소에 대한 부표 모델의 움직임은 수학적 알고리즘에 의해서 결정된다. Fig. 3은 바람과 파도, 그리고 조류에 관한 사례에 대해 산식을 이용한 결과를 보여준다(국토해양부, 2010). 주어진 결과는 부표를 포함한 해상 구조물에 대한 실험의 일부를 보여주는 것으로, 무게 5,743kg의 해상 구조물에 대해 적용한 사례이다.

바람과 조류에 의한 경사각(θ)은 회복원력(Gm)을 바람과 조류에 대해 저항력을 계산하고, 저항에 대한 복원모멘트의 $\sin\theta$ 를 구하여 경사각을 도출하였다. 파도에 대해서는 수심 20미터 및 파고 5미터의 조건에서의 결과이다.

본 논문에서는 2011년 개정된 부유식 해상 구조물의 구조에 관한 정부 고시(국토해양부, 2011)에서 제시하는 바람에 의한 구조물의 복원력 검사를 위한 풍속 조건을 제시하고 있다. 본 실험에서는 이를 최대한 수용할 수 있도록 프로그램을 설계하였다.

가) 바람에 의한 경사각

$$\text{풍력} : F_a = 1/2 \times \rho_a C_D A_a V_a^2$$

$$\text{복원모멘트} : M_R = W \cdot G_M \cdot \sin\theta = 772.147 \text{ kgf} \cdot \text{m}$$

$$5743 \times 0.309 \times \sin\theta = 772.147$$

$$\text{최대경사각 } \theta_{\max} = 19.69^\circ$$

나) 파도에 의한 경사각

$$\theta_{\max} = \frac{\alpha}{1 - t_o^2 / T^2}$$

$$\lambda_o = \frac{g T^2}{2\pi} = \frac{9.8 \times 10^2}{2\pi} = 155.972 \text{ m (여기서 } \lambda_o : \text{심해파의 파장)}$$

$$d / \lambda_o = 20 / 155.972 = 0.128 \text{ (여기서 } d \text{ 는 수심)}$$

$$d / \lambda = 0.1649$$

$$\lambda = 121.29 \text{ m}$$

$$\alpha = \frac{\pi H}{\lambda} = 3.14 \times 5 / 121.29 = 0.129 \text{ (여기서 파고 } H \text{ 는 5m)}$$

$$\theta_{\max} = \frac{0.129 \times 10^2}{10^2 - 7.84^2} = 0.335 \text{ rad} = 19.18^\circ$$

다) 조류에 의한 경사각

$$\text{조류력} : F_c = 1/2 \times \rho_w C_D A_c V_c^2$$

$$\text{복원모멘트} : M_R = W \cdot G_M \cdot \sin\theta = 763.47 \text{ kgf} \cdot \text{m}$$

$$5743 \times 0.399 \times \sin\theta = 763.47$$

$$\text{최대경사각 } \theta_{\max} = 19.46^\circ$$

Fig. 3 Example of Calculating Tilt Angle on External Factors

(5) 물리엔진

물리 엔진은 강체동역학, 연체동역학, 유동역학과 같은 물리 시스템을 시뮬레이션 하는 컴퓨터 소프트웨어이다. 주로 비디오 게임, 컴퓨터 그래픽, 영화 등에 사용된다(Physics, 2012).

물리엔진에는 여러 가지 이론을 적용한 다양한 종류가 있지만, 본 논문에서는 Quest3D에서 기본적으로 내장한 ODE(Open Dynamics Engine)를 사용했다. ODE는 실시간 강체동역학 시뮬레이션이 가능한 엔진으로 속도와 안정성에 중점을 두고 설계되었다.

외부 영향요소간의 복합적 관계나 해상부표 모델의 세부 특징에 대해 최적화된 물리엔진이 필요하지만 이점은 본 논문의 단계에서는 고려하지 않았으며 기존의 엔진을 최대한 활용한 수준에서 적용하였다. 향후 목적에 최적화되도록 개조 하거나 개발해 적용할 예정이다.

3.3 프로그램 설계

본 논문에서는 2.4절에서 언급한 Quest3D를 이용한 프로그래밍의 고려사항을 반영하여 프로그램 구조를 설계했다. Fig. 4는 본 논문에서 개발한 프로그램의 전체 구조를 보여준다. 프로그램을 작동하기 위한 기본 단위블록을 다음과 같이 구분하였다.

- 해상부표 모델 정의: 3D Max기반 모델을 Quest3D에 적용

가능한 형태로 프로그래밍

- 해상부표 모델의 환경 적응을 위한 계산: 해상부표 모델을 물 모델 등 주변 환경에 맞도록 위치를 조절하기 위한 프로그래밍
- 사용자지정 외부환경 변수 적용 : 사용자가 정의한 외부환경을 시각적으로 표현하기 위한 계산식 프로그래밍
- 물 모델 적용 : 기존 Quest3D에서 제공하는 물 모델에 해상부표의 외부환경 변수를 수용하도록 수정
- 렌더링 : 시각적 표현의 화면 디스플레이를 위해 사용자인터페이스(User Interface)에 사용자 요구대로 그리는 그래픽 프로그래밍

각 단위블록의 기능은 겹치는 부분이 없이 정의되었으며, 계산이나 내부 함수에 필요한 데이터 중 내부에 생성이 되지 않는 경우, 그 데이터 값을 보유한 블록과 연계하여 데이터를 취득하는 방식이다.

이는 추후 다른 모델을 추가하거나, 정적 안정성을 위한 기능을 추가하거나, 정적 및 동적 안정성 검증을 위한 환경변수를 추가하더라도 구조변경 없이 프로그래밍이 가능한 기반을 제공한다.

Fig. 4 The Quest3D Program Structure for Marine Buoy Test

각 단위블록은 채널로 구현되며, 내부에 필요한 변수와 기능을 구현하고, 데이터 송수신이 필요한 다른 채널로 링크를 하는 방식으로 프로그래밍 한다. 기본 단위블록에 대한 설명은 다음과 같다.

(1) 부표 모델(Buoy Model)

부표 모델은 설계가 완료된 모델을 3D MAX 툴을 통해서 Quest3D에서 사용이 가능한 포맷으로 Export하여 사용한다. 각 모델들은 Quest3D에서 Channel Group 단위로 Import 된다. Import된 각 모델의 채널 그룹에 별도로 부력이나 각종 변수 채널들을 추가하고 3D 모델의 동작이나 형상을 결정하는 Motion Matrix를 각종 연산에 의한 결과로 나온 채널로 대체한다. Fig.5는 본 논문의 실험에 적용한 모델의 사례이다.

Fig. 5 Buoy Model

(2) 물리적 연산(Physical Calculation)

Fig. 6은 부표의 움직임의 물리적 연산을 위한 채널들을 보여준다. 그림에 자세히 보이지 않는 채널명은 번호로 표기하고 명시하였다. 부표의 각 모델에 관한 각종 변수들과 환경 변수를 입력받아 부표의 동작 상태를 나타내는 Motion Matrix를 실시간으로 계산한다.

Fig. 6 A part of physical calculation channels

(3) 사용자 지정 환경변수(User Specific Environment Variable)

각종 환경 변수들과 사용자인터페이스를 표현하기 위한 채널 묶음이다. Fig.7의 구조를 가지며 환경 변수를 조절할 수 있고, 물리 연산과 물의 표현을 위한 값으로 입력한다.

Fig. 7 A part of environment parameter channels

(4) 물 모델(Water Model)

물의 표현을 위한 Quest3D 기본 제공 모듈이다. 물에 관한 사실적인 표현을 위해서 20여개의 변수들을 입력 받고 있다. Fig.8의 변수들은 모두 사용자에게 의해서 조절 가능하다.

Fig. 8 Water Model Channel

(5) 사용자인터페이스 렌더링(User Interface Rendering)

Quest3D의 GUI 채널을 사용하여, Fig.10과 같이 버튼타입으로 부표상태에 대한 상세 명세(Description)를 정의하고 이를 렌더링 한다.

Fig. 9 User Interface Render Channels

(6) 모델 렌더링(Model Rendering)

본 논문의 실험에서 렌더링 할 수 있는 부표의 수를 20개 이상 가능하도록 설계했다. GUI에서 버튼을 누름에 따라 각 버튼에 해당하는 부표들이 화면에 렌더링 된다. Fig.10과 같이 각 부표별로 번호를 부여하고 부표 'Channel Switch' 채널을 이용해서 각 선택된 부표들을 '3D Render' 채널로 연결하였다.

Fig. 10 3D Render Channel

4. 구현 결과

본 논문에서 구현된 프로그램은 설계 시나리오에 따라 두 개의 화면으로 구성하였다.

Fig. 11은 모델과 실행할 안정성 모드 선택 화면이며 프로그램의 시작 시 처음에 나타나는 화면이다. 화면 좌측의 ①에서 모델을 선택할 수 있다. 20개 이상의 모델들을 실험할 수 있는 확장 구조를 구현하였으며, 각 모델에 대한 선택버튼을 한 화면상에 배치하고, 선택된 버튼과 선택되지 않은 버튼의 색을 다른 색으로 표시하여 구별 할 수 있도록 구현하였다.

본 논문의 실험 대상 모델이 화면 한가운데에 3D로 구현되어 있는 모습이다.

Fig. 11 Model Selection Scene

Fig. 11의 ③에서는 선택된 모델의 상세 내역을 볼 수 있다. 부표의 설계에 의한 정적 정보로서 크기와 무게 등의 내역을 보여준다. ④와 ⑤는 각각 정적 안정성과 동적 안정성모드를 선택하기 위한 버튼과 그에 대한 간단한 설명이다. 사용자는 ⑥의 시작 버튼을 눌러 프로그램을 시작하고 정지할 수 있다.

Fig. 12은 Fig. 11에서 사용자가 원하는 설정을 완료한 후, 시각화 렌더링이 실행되고 있는 화면이다. ①은 프로그램에서 각종 변수를 조절하기 위한 슬라이드 바(Slide Bar)기능이다. 풍향, 풍속, 조류 방향, 세기 등의 프로그램에 영향을 끼치는 변수를 조절하여 해당 값을 자동으로 적용할 수 있다.

Fig. 12 Simulation Scene

화면의 ②는 ①에서 적용된 자연환경 변수에 의해 변화하는 부표의 모습을 보여준다. 우측의 ③번은 프로그램상에서 모델의 기울기와 같은 수치들을 보여준다. 이 수치들은 동적 정보로서 실시간으로 부표와 자연 환경 변수의 변화에 대한 정보를 보여준다.

④의 버튼으로 프로그램을 정지하거나 종료할 수 있다.

5. 결 론

해양관련 산업의 발달과 더불어, 선박은 대형화되고 고속화되어가고 있다. 이러한 변화에 따라 선박의 항로 통항시 안전운항을 도모하기 위한 해상교통시설물 구축이 필수적이다.

본 논문은 새로운 해상부표를 도입하거나 기존 해상부표의 안정성을 검토하기 위해 컴퓨터 그래픽 기술을 도입해보았다. 그래픽 기술을 활용하여 시각적으로 표현이 가능하고 프로그램 설계 시 확장을 고려하여 객체지향 기반 프로그래밍 환경을 제공하는 Quest3D를 도입하였다.

본 실험은 안정성을 평가하는 다양한 요소 중 동적안정성의 일부요소에 대해 적용해 본 결과이며, 상용화되어 사용되고 있는 정적 및 동적 안정성 검증을 위한 소프트웨어와 같은 수준의 기능을 위해서는 본문에서 언급한 연구와 실험이 필요하다.

참 고 문 헌

- [1] 국토해양부(2010), “해상여건에 적합한 특수 부표류 개발 및 등탑상부 표준화용역 보고서,” 국토해양부 해양교통시설과.
- [2] 국토해양부(2011), “부유식 해상구조물의 구조 및 설비 등에 관한 기준,” 국토해양부.
- [3] 김국, 김진구(2011), “항로표지와 관련해사영어의 표준화 방안,” 한국항해항만학회지, 제35권, 제1호, pp. 31-38.
- [4] 이근왕, 박종천(2009), “Quest기반의 환경교육용 3D 가상 체험 소프트웨어 설계,” 한국산학기술학회 2009년도 춘계 학술발표논문집, pp. 44-46.
- [5] 전완식(2010), “Quest3D를 이용한 건설기계 시뮬레이터의 설계와 구현,” 청주대학교 대학원.
- [6] 한국선급(2011), 선급기술규칙, <http://www.krs.co.kr/>
- [7] Act-3D(2009), “About Quest3D,” <http://www.quest3d.com/>
- [8] Birch, D.(2009), “Mapping Game Engines for Visualization,” Map of Game Engines
- [9] Heightmap(2012), “Heightmap,” <http://en.wikipedia.org/wiki/Heightmap>
- [10] Sastry, J. S. V. R. S., Ramesh, K. V.(2011), “Measuring Object-Oriented Systems based on the Experimental Anaysis of the Complexity Metrics,” International Journal of Engineering Science and Technology, Vol. 3, No. 5.

- [11] Matthias Müller-Fischer(2008), "Fast Water Simulation for Games Using Height Fields," Game Developers Conference, <http://www.gdcvault.com>
- [12] Physics(2012), "Physics engine," http://en.wikipedia.org/wiki/Physics_engine
- [13] Quest3D(2012), "Quest3D pattern," <http://support.quest3d.com/index.php?title=Tutorial/OO/Patterns/Introduction>
- [14] Singh, S., Kahlon, K. S.(2010), "Static Analysis To Model & Measure OO Paradigms," ACM Symposium On Applied Computing

원고접수일 : 2012년 1월 4일
심사완료일 : 2012년 8월 8일
원고채택일 : 2012년 8월 20일