

# An Improved Method of Guaranteeing Frame Rates of Avionics Simulator based on HMD Motion

Jeong-Hoon Lee\*, Yong-Il Jo\*\*, Kyong Hoon Kim\*\*\*

## Abstract

In this paper, we propose an improved algorithm for rendering method to guarantee frame rates based on HMD (Head Mounted Display) motion in an avionics simulator. One of important issues in HMD simulators is to guarantee frame rates despite fast motion of HMD which is more rapid than the aircraft's moving speed to maintain a quality of images. Therefore, we propose an algorithm considering the moving speed of a pilot's head: Improved Speed-Based LOD (Level-Of-Detail) Control (ISBLC).

In the proposed algorithm, frame rates are improved by changing dynamic LOD which determines details of objects for rendering images. Throughout the experiments, we show the average frame rates are achieved up to 60 and minimum frame rates are guaranteed up to 40. The proposed algorithms will be used HMD simulation in avionics simulators.

▶ Keyword: Rendering, Avionics Simulator, Helmet Mounted Display

## 1. Introduction

현재의 항공 산업은 매우 기술 집약적 산업이며, 다양한 기술 분야가 접목되어 적용되는 산업으로 나아가고 있다. 그에 따라서 기술 비용도 기하급수적으로 증가하고 있다. 시뮬레이터는 다양한 기술들을 항공기에 접목하기 전에 발생할 수 있는 문제점을 조기에 파악하여 수정이 용이하게 하며, 개발 시간을 단축시킬 수 있다. 또한 실제 비행 시 발생할 수 있는 비용 및 위험 요소를 줄일 수 있기 때문에 시뮬레이터의 중요성이 커지고 있다.

HMD는 Helmet Mounted Display의 약자로서 머리에 쓰는 헬멧에 부착되어진 디스플레이를 의미한다. HUD(Head Up Display)의 정보를 보다 손쉽게 직관적으로 조종사에게 전달하기 위해 개발되어진 것으로 HUD와 다르게 조종사가 정면의 계

기판을 보지 않고 머리를 회전한 상태에서도 현재의 비행 상태를 파악하는데 도움을 준다. 항공기에 사용되는 HMD(Head Mounted Display)는 현재까지 지속적으로 연구 되고 있는 분야로서 기술 자체의 도입은 오래전에 이루어졌지만 장비의 무거움으로 인해 조종사의 신체에 부담을 주어 기체의 급격한 이동이 적은 헬리콥터와 같은 회전익에 주로 도입되어 사용되어 왔다. 하지만 기술의 발전으로 장비의 무게가 감소되어 헬리콥터가 아닌 다른 항공기에서도 필수 부품으로 자리 잡게 되었다.

HMD 장비를 이용한 시뮬레이터의 도입이 요구되면서 HMD 장비에 표시되는 각종 심볼과 비행 정보들을 출력해주는 HMD 시뮬레이터의 필요성이 대두되었다. Fig. 1의 좌측은 F15계열의 HMD이며, 우측은 HMD 시뮬레이터 영상이다. HMD 시뮬

• First Author: Jeong-Hoon Lee, Corresponding Author: Kyong Hoon Kim

\*Jeong-Hoon Lee (burning\_sun@naver.com), Appia Engineering, LTD

\*\*Yong-Il Jo (cruess@gnu.ac.kr), Dept. of Informatics, Gyeongsang National University

\*\*\*Kyong Hoon Kim (khkim@gnu.ac.kr), Dept. of Informatics, Gyeongsang National University

• Received: 2018. 07. 11, Revised: 2018. 07. 18, Accepted: 2018. 07. 26.

• This study was supported in part by the Korea Evaluation Institute of Industrial Technology (KEIT) under "Avionics Integration Software Development for Supersonic trainer (No.10059058)" and by the Human Resources Development of the Korea Institute of Energy Technology Evaluation and Planning (KETEP) under "Advanced Human Resources Track for Power System Resiliency from An Earthquake and Disasters (No.20174030201440)" funded by the Ministry of Trade, Industry & Energy.

레이터에서는 조종사가 보는 방향의 지형과 비행에 필요한 헤딩, 고도, 피치, 롤 등의 비행정보를 표시한다.

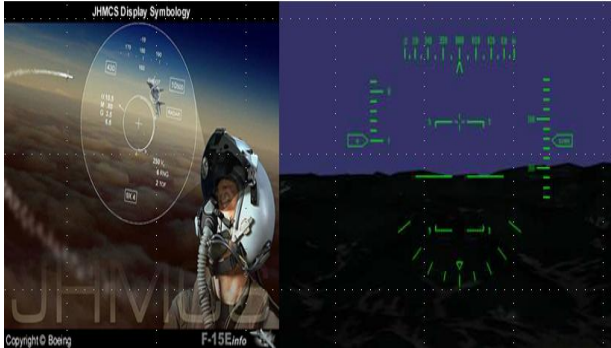


Fig. 1. The HMD and HMD simulation software[1][5]

HMD는 조종사의 머리회전에 따라 시야의 변화가 발생한다. 머리회전은 매우 빠른 속도로 움직이기 때문에 일반적인 시뮬레이터의 렌더링 메커니즘과는 다르게 HMD의 특성을 고려한 별개의 메커니즘이 필요하다.

따라서 본 논문에서는 HMD의 특성을 반영하여 초당 프레임 수를 보장하기 위한 방법을 제안하고자 한다. HMD 시뮬레이터는 최대한 조종사에게 실제 환경과 유사하게 구현하기 위해 상용 항공시뮬레이터와 연동하여 3차원으로 구성하고 상용 항공시뮬레이터에서 생성된 항공기 상태 정보에 따라서 동작한다.

제안하고자하는 초당 프레임 수 보장기법은 HMD의 특성 중 하나인 급격한 시야변화 속도를 반영한 동적 LOD(Level Of Detail) 변경 방법이다. 그리고 빠른 속도로 인한 지형의 일그러짐 현상을 해소하기 위해 모션 블러 효과를 사용하여 현실감을 높이는 방안도 제시한다. 제안하는 기법을 적용한 HMD 시뮬레이터는 OpenSceneGraph 라이브러리를 사용하여 소프트웨어를 구성하였고 10m급 지형데이터를 사용하여 렌더링 성능을 측정하였다.

본 논문의 구성은 다음과 같다. 2장은 관련연구에 대한 소개이며, 3장은 논문에서 제안하고자하는 HMD움직임 속도를 이용한 제안 메커니즘이다. 4장은 성능 평가를 보여줄 것이며, 5장에서는 결론을 제시한다.

## II. Related works

본 장에서는 HMD 시뮬레이터와 각종 영상 처리 관련 기법 등의 관련연구에 대해 설명한다.

### 1. HMD simulator

HMD는 운항에 필요한 네비게이션 정보뿐만 아니라 적기와의 교전을 위한 레이더 정보 및 탑재된 무기 정보 등 다양한 정보를 헬멧에 표시한다. 이와 같은 직관성 때문에 현재의 최신 전투기에는 HMD가 필수적으로 도입되어지는 추세이며 여러

가지의 비행정보를 직관적으로 조종사에게 전달하기 때문에 존재 유무에 따라 공중전의 판세에 많은 영향을 미친다.

최근에는 HMD와 RGB-D 카메라를 사용하여 실제 및 가상

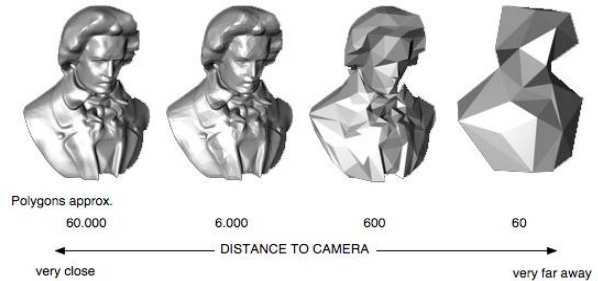


Fig. 2. Expression of static LOD by distance[2]

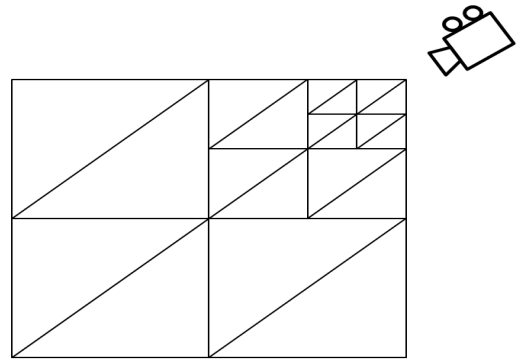


Fig. 3. The geographical precision of Dynamic LOD by distance[3]

객체와 동적 3D 상호작용을 가능하게 하는 시스템에 관한 연구와[12] 가상현실 구현을 위해 여러 종류의 HMD 장비를 여러 사용자들에게 체험하게 하고 각 HMD 장비에 적용된 3차원 렌더링 기술이 사용자의 체감 품질에 미치는 영향을 파악하기 위한 연구[9] 등 가상 물체와 상호 작용 관계를 파악하고 이를 향상시키기 위한 연구가 대부분이다. 또한 시뮬레이션의 frame rates의 변화에 따라 LOD를 조절하여 성능을 향상시키기 위한 방법도 연구되었지만[5] HMD의 급격한 움직임 속도를 반영하여 성능을 향상시키는 연구는 아직 미비하다. 따라서 본 논문에서는 조종사의 빠른 시야변화에 적합한 메커니즘을 반영하여 동적으로 LOD 레벨을 조정하는 방법을 사용하여 시뮬레이터의 성능을 향상시키고자 한다.

### 2. The LOD technique to Improve image processing speed

LOD(Level Of Detail)는 렌더링 대상이 될 객체, 그리고 그 객체와 거리를 두고 존재하는 카메라의 거리에 따라서 객체의 세밀함 정도를 다르게 표현하는 기법이다. LOD 단계별로 카메라와 거리가 가까운 객체의 세밀함은 자세하게 표현하고 멀리 있는 객체의 세밀함은 낮게 표현함으로써 성능을 향상시킬 수 있다. 이러한 LOD는 정적 LOD와 동적 LOD로 나뉘며 하위 절에서 두 가지 기법에 대해 설명한다.

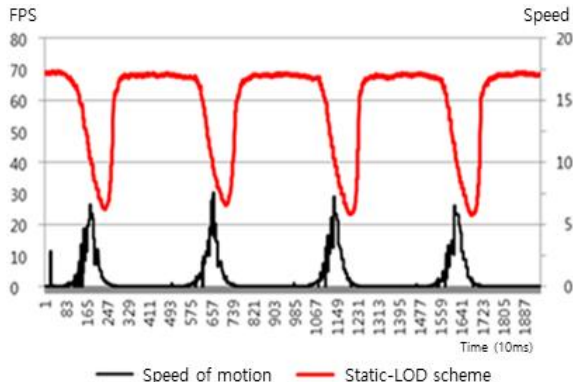


Fig. 4. A graph of FPS when the HMD motion occurs

### 2.1 Static LOD

정적 LOD는 객체의 정밀도에 따라 데이터를 분할하는 기법이다. Fig. 2를 보면 정적 LOD는 어떠한 객체에 대해 단계별로 모든 메쉬를 생성하는 것을 알 수 있다. 카메라가 이동하면서 객체와 카메라간의 거리가 변경이 되면 거리에 따른 단계별 메쉬를 교체하여서 디스플레이에 출력한다. 사전에 단계별 메쉬를 메모리에 적재하기 때문에 속도가 빠른 장점이 있지만 모든 메쉬를 메모리에 적재해야하기 때문에 메모리 낭비가 심하며 거리에 따라서 메쉬의 단계가 급격하게 변하기 때문에 사물과 같은 작은 객체에 주로 사용된다.

### 2.2 Dynamic LOD

동적 LOD는 정적 LOD와 다르게 거리에 따라 실시간으로 정밀도를 변화시키는 기법이다[5][10]. Fig. 3을 보면 근거리에서는 메쉬가 세밀하게 구성되며 중간거리에서는 그보다 덜 세밀하게 구성되고 먼 거리에서는 메쉬가 덩성덩성 구성되어지는 것을 볼 수 있다. 이처럼 메쉬를 실시간으로 거리에 따라 조정하기 때문에 정적 LOD와는 다르게 한 개 분량의 메쉬만을 사용하며 단계별 메쉬를 저장함으로써 낭비되는 메모리의 누수 또한 적다. 또한 LOD의 단계 변화가 자연스럽게 이루어지기 때문에 지형과 같은 대용량 객체 처리에 매우 용이하다.

### 2.3 Motion blur

모션 블러는 어떠한 객체가 이동하였을 때 그 객체의 잔상을 남겨서 움직임 효과를 증대하는 기법이다[5][11][17]. 사람의 시야변화가 있을 때 사물이 흐릿하게 보이는 것처럼 움직임의 잔상을 표현하여 객체를 흐릿하게 구현하고 잔상을 통하여 생동감 있는 화면을 구현한다. 또한 잔상의 강도를 변경하여 속도감을 조절할 수 있다.

## III. The proposed schemes

항공기에서 사용되는 HMD의 경우 장비의 움직임만 존재하

는 것이 아니라 항공기의 비행 움직임과 조종사의 움직임 또한 포함되어진다. 이 경우 급격한 시야의 변화가 발생하여 시물레

```

Algorithm ISBLC
/*
Scurr : the current motion speed of HMD
Smin : the minimum motion speed of HMD
Sstand : the standard of speed for estimating LOD
LODcurr : current LOD level
LODest : estimated LOD
LODmin : minimum value of LOD
LODmax : maximum value of LOD
α : the weight for estimating LOD
*/
1: while true do
2:   if Scurr <= Smin then
3:     LODcurr ← max(LODcurr-1, LODmin)
4:   else
5:     LODest ← Scurr * α
6:     if LODest > LODcurr then
7:       LODcurr ← min(LODcurr+1, LODmax)
8:     endif
9:   endif
10:   Renders the geographical image using LODcurr
11: endwhile
    
```

Fig. 5. A pseudocode of ISBLC algorithm

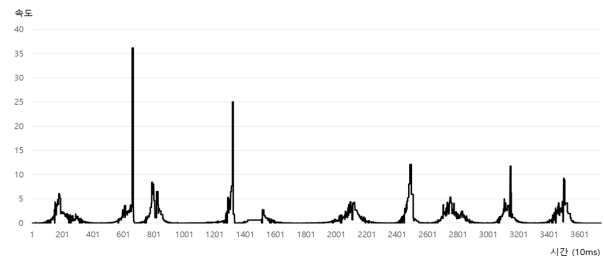


Fig. 6. A motion scenario for experiment

이터에서 렌더링 시 FPS(Frame Per Second) 저하가 발생한다. Fig. 4는 HMD의 움직임이 발생했을 때 나타나는 FPS 변화를 나타낸 그래프이다. 아래쪽의 실선은 HMD의 움직임 속도를 나타내고, 위쪽의 실선은 시물레이터의 FPS 변화를 나타내며, HMD의 움직임 속도가 증가할 때마다 FPS의 저하 현상이 발생하는 것을 알 수 있다. 최악의 경우 22FPS까지 하락이 발생하며, 이는 화면 끊김 현상의 원인이 되어 시물레이션의 현실감을 떨어뜨리기 때문에 급격한 FPS의 하락에 대한 개선이 필요하다.

이전 연구에서는 HMD를 사용함으로써 발생하는 시야의 급격한 변화로 인한 FPS의 저하 요소와 실제 HMD 시물레이터 사용자의 현실감 있는 영상 품질 요소를 고려하여 동적 LOD변경 방법과 모션블러를 이용한 연구를 진행하였다[3]. HMD의 움직임 속도에 비례하여 LOD를 동적으로 변경하여 성능을 향상시켰으며, 알고리즘을 적용하기 전보다 개선된 성능을 보였다[3]. 하지만 속

도가 변할 때마다 LOD를 변화시켰기 때문에 렌더링 연산의 과부하를 일으켜 렌더링 속도를 떨어뜨리는 원인이 되었다. 따라서 이번 장에서는 잦은 LOD 변경에서 발생하는 과부하를 줄여 성능을 향상시키기 위한 새로운 방법을 제안한다.

**1. ISBLC (Improved Speed-Based on LOD Contorl) algorithm**

이 알고리즘은 이전 연구에서 조금 더 개선된 알고리즘으로써 HMD의 움직임 속도가 변할 때마다 변경되는 LOD로 인하여 발생하는 과부하를 줄이기 위하여 제안하였다. 이전 연구의 주요 내용은 HMD의 움직임 속도가 변할 때마다 LOD 레벨을 수시로 변경시키는 것이었다. 하지만 동적 LOD 변경은 시스템에서 많은 연산이 필요하기 때문에 잦은 LOD 변동은 시스템에 과부하를 야기하며 성능 저하의 요인이 된다. 따라서 LOD의 잦은 변동을 줄이기 위해 속도가 감소하는 경우에는 LOD 레벨을 감소시키지 않다가 속도가 HMD의 움직임 속도가 최소수준으로 떨어진다면 LOD를 한 단계 밑의 레벨로 감소시키는 방법을 제안한다.

Fig. 5는 제안하는 알고리즘의 속도코드이다. HMD의 움직임과 속도에 비례한 LOD 조절은 이전 연구의 알고리즘과 동일하다. 하지만 움직임이 발생하였을 때 머리 움직임의 최소기준인  $S_{min}$ 을 넘지 못한다면 움직임이 발생하지 않은 걸로 판단하여 LOD를 한 단계 밑의 레벨로 낮추게 된다. 만약 HMD의 움직임 속도가 최소수준으로 떨어졌을 때 LOD를 한 단계 밑이 아닌 최소 수준으로 낮추게 된다면 렌더링 복원 시 연산량이 급격하게 증가하여 오히려 성능이 저하의 원인이 되므로 점차적으로 LOD 레벨을 감소시킨다.

이후 움직임이 발생한 것으로 판단되면 속도에 비례한 LOD를 추정하여 현재의 LOD 레벨을 비교하고 추정된 LOD 레벨이 더 크다고 판단되면 속도가 감소한 것이기 때문에 현재의 LOD를 이전의 LOD 그대로 유지하게 된다. 반대의 경우에는 속도가 증가하고 있음이 판단되므로 LOD 단계를 상승시키게 된다.

**IV. Evaluation**

실제 HMD의 움직임과 최대한 유사하게하기 위해 순간 감속과 가속을 추가하여 Fig. 6 과 같이 실험 속도 시나리오를 구성하였다. 순간적으로 움직임을 발생시켜 많은 가속을 하였다가 급격한 감속으로 움직임이 멈추거나 급격한 머리 방향 전환으로 속도가 순간적으로 감속되는 구간을 추가하여 최대한 실제 조종사의 급격한 시야전환과 유사하게 시나리오를 만들었다.

실험 결과는 실험 시나리오에서 작성한 움직임 시나리오와 Table 1의 실험환경, Table 2의 실험 데이터를 바탕으로 2가지 알고리즘에 대해 동일하게 실험을 진행하였으며 알고리즘을 적용하였을 때의 FPS 변화와 각 알고리즘의 평균 FPS, 평균 LOD에 대해서 측정하였다.

Table 1. Experimental environment

CPU	Intel Core i5-3570 3.40Ghz
RAM	12GB DDR3
GPU	NVIDIA GeForce 9800 GT
OS	Windows 7
Library	OpenSceneGraph
Language	C++

Table 2. The data for experiments

Range of Map	100km around Yecheon Airport
Data Size	3.1 GB
The Number of Vertices	4,320,857 EA
LOD Level	3 Level LOD
Range of FOV	120°
Range of LOS	1. Heading : 180° 2. Pitch : 90°

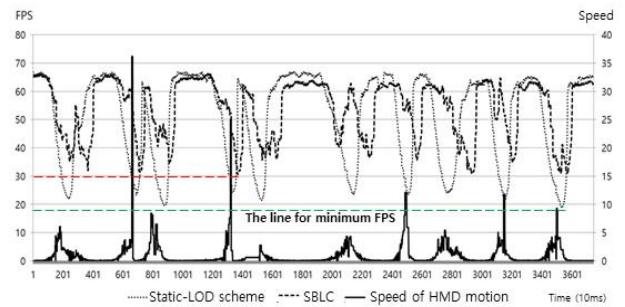


Fig. 7. A FPS graph of SBLC algorithm [3]

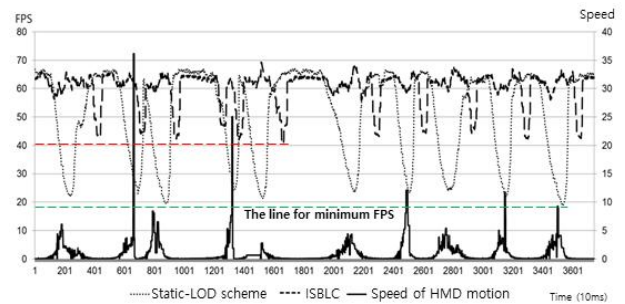


Fig. 8. A FPS graph of ISBLC algorithm

**1. FPS change of algorithm**

각 알고리즘을 적용하여 동적 LOD를 적용하지 않은 상태와 비교를 하였다. Fig. 7, Fig. 8의 그래프를 보면 모두 동적 LOD를 적용하지 않았을 때 보다 개선된 FPS를 보여준다.

제안하는 알고리즘은 최저 40FPS를 넘는데 비해 이전 연구에서 [3] 제안한 알고리즘은 최저 30FPS를 만족하는 것으로 나타났다. 이는 실제 움직임이 발생하여 LOD를 동적으로 변화시키는 과정에서 많은 과부하가 발생하고 그에 따른 렌더링 연산의 속도저하로 FPS의 하락이 발생한 것이다.

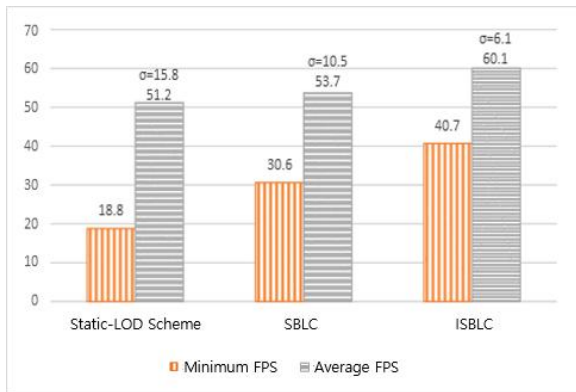


Fig. 9. The data graphs of each algorithm

## 2. Performance comparison of each algorithm

실험 시나리오를 적용한 각 알고리즘의 평균 FPS는 Fig. 9에서 보여주듯이 제안하는 ISBLC 알고리즘이 약 60FPS로 가장 좋은 성능을 나타냈다. HMD의 움직임 속도에 따라 LOD를 변경하는 SBLC 알고리즘과는 달리 ISBLC 알고리즘에서는 HMD의 움직임 속도가 최소수준일 경우에만 LOD 레벨을 낮추어 LOD 변동 빈도를 줄이기 때문에 향상된 성능을 나타내는 것을 알 수 있다. 표준편차 또한 각 알고리즘 중 6.1로 가장 낮아 안정적인 성능을 가지는 것을 알 수 있다.

## V. Conclusions

본 논문에서는 항공 시뮬레이터에 초점을 맞춘 것이 아닌 HMD 시뮬레이터에 초점을 맞추어서 HMD만의 요소라고 할 수 있는 조종사의 머리 움직임에 맞추어 동적으로 LOD를 변동하는 방법을 제안하였다. HMD 움직임 속도에 맞추어 LOD 레벨을 변경하는 알고리즘을 제안하였으며 이전 연구에서[3] 제안한 알고리즘과 비교하여 개선된 성능을 보여주었다.

실제 HMD의 움직임은 매우 빠르기 때문에 잦은 LOD 레벨의 변경이 발생하고 그로 인해 렌더링 과정에서 많은 과부하가 발생하는 것을 발견하였다. 따라서 과부하를 감소시키기 위해 LOD의 변동 빈도를 낮추기 위해 알고리즘을 제안하였고 그런 알고리즘들의 적용으로 과부하를 어느 정도는 해소 하는 결과를 얻게 되었다. 제안한 알고리즘들이 평균 50FPS 이상의 성능을 보였으나, 실제 HMD의 구동환경에서는 또 다른 연산요소가 발생할 수 있기에 그보다 높은 60FPS의 기준에는 제안한 ISBLC 알고리즘만이 만족하는 결과를 얻게 되었다.

향후 연구로는 LOD변화가 발생하였을 시에 지형의 일그러짐 현상을 해소하기 위한 모션 블러를 속도에 따라 단계적으로 적용하는 방안과 HMD의 움직임 속도가 아닌 가속도를 이용하여 동적 LOD를 구성하는 방안에 대하여 연구할 예정이다.

## REFERENCES

- [1] Helmet Mounted Display (HMD), <https://namu.wiki/w/HMD#s-3>.
- [2] Geometry LOD (Level of Detail), <http://www.indigo.idv.tw/?p=41>.
- [3] J.H. Lee, Y.I. Jo, K.I. Kim, and K.H. Kim, "A method of guaranteeing frame rates for HMD simulator," The 13th IEMEK Symposium on Embedded Technology, pp. 226-228, May 2018.
- [4] Y.I. Jo, J.H. Seo, K.I. Kim, and K.H. Kim, "Applying multi-threaded computing in aircraft simulators," The 13th IEMEK Symposium on Embedded Technology, pp. 207-209, May 2018.
- [5] J.H. Lim, C.J. Lee, K.H. Kim, and K.I. Kim, "Preventing rendering speed degradation for helmet mounted display simulation," International Journal of Signal Processing, Image Processing and Pattern Recognition, Vol. 10, No. 7, pp. 9-20, July 2017.
- [6] B. Oliver, A. Fuhrmann, and S.M. Grünvogel, "The effectiveness of changing the field of view in a HMD on the perceived self-motion," IEEE Symposium on 3D User Interfaces, pp. 225-226, March 2017.
- [7] K. Wu and C. Yuksel, "Real-time cloth rendering with fiber-level detail," IEEE Transactions on Visualization and Computer Graphics, Vol. 14, No. 8, pp. 1-12, August 2017.
- [8] S.H. Yoon, H.J. Cho, S.W. Cho, and Y.H. Kim, "HMD-based virtual multi-screen control system and its gesture interface," International SoC Design Conference, pp. 270-271, November 2017.
- [9] A. Basu, C. Ball, B. Manning, and K. Johnsen, "Effects of user physical fitness on performance in virtual reality," IEEE Symposium on 3D User Interfaces, pp. 233-234, March 2016.
- [10] J. Zienkiewicz, A. Tsotsios, and A. Davison, "Monocular, Real-Time Surface Reconstruction Using Dynamic Level of Detail," International Conference on 3D Vision, pp. 37-46, October 2016.
- [11] J.H. Lim, C.J. Lee, K.H. Kim, and K.I. Kim, "Employing Blur Effect on Fast Moving Object," Advanced Software Engineering and Its Applications, Vol. 139, pp. 470-473, November 2016.
- [12] N. Sugiura and T. Komuro, "Dynamic 3D interaction using an optical see-through HMD," IEEE Virtual Reality Conference, pp. 359-360, March 2015.
- [13] F. Biljeck, H. Ledoux, J. Stoter, and J. Zhao, "Formalisation of the level of detail in 3D city modelling," International Journal of Computer Environment and Urban Systems,

Vol. 48, pp. 1-15, November 2014.

- [14] J.A. Jones, E.A. Suma, D.M. Krum, and M. Bolas, "Comparability of narrow and wide field-of-view head-mounted displays for medium-field distance judgments," *Proceedings of the ACM Symposium on Applied Perception*, pp. 119-119, August 2012.
- [15] G. Bao, H. Li, X. Zhang, and W. Dong, "Large-scale forest rendering: Real-time, realistic, and progressive," *Computers and Graphics*, Vol. 36, pp. 140-151, May 2012.
- [16] W. Ying and W. Yanjie, "Implementation of a fast simulation algorithm for terrain based on Dynamic LOD," *IEEE International Conference on Electronic Measurement and Instruments*, Vol. 3, pp. 189-193, August 2011.
- [17] M.M. Wloka and R.C. Zeleznik, "Interactive real-time motion blur," *Journal of Visual Computer*, Vol. 12, No. 6, pp. 283-294, June 1996.

**Authors**



Jeong-Hoon Lee received the B.S. degree in Management Information Systems and M.S. degrees in Informatics from Gyeongsang National University, Korea, in 2015 and 2017, respectively. He joined Appia Engineering, LTD, Sacheon, Korea, in 2018,

as he is currently a researcher to develop a program for avionics systems. He is interested in real-time systems, application software, and avionics systems.



Yong-II Jo received the B.S. and M.S. degrees in Informatics from Gyeongsang National University, Korea, in 2012 and 2014, respectively. He has been started Ph.D course in the department of Informatics at Gyeongsang National

University, Jinju, Korea, in 2014. He is interested in real-time systems, avionics systems, and mobility models.



Kyong Hoon Kim received the B.S., M.S. and Ph.D. degrees in Computer Science and Engineering from Pohang University of Science and Technology, Korea, in 1998, 2000 and 2005, respectively. He joined the faculty of the Department of Informatics

at Gyeongsang National University, Jinju, Korea, in 2007. He is interested in real-time systems, mobile computing, cloud computing, and avionics systems.