시청피로 저감형 S3D 영상 재생 시스템 구현 및 실시간 처리를 위한 알고리즘 연산량 분석

이재성*

Implementation of Stereoscopic 3D Video Player System Having Less Visual Fatigue and Its Computational Complexity Analysis for Real-Time Processing

Jaesung Lee*

Department of Electronic Engineering, Korea National University of Transportation, Chungbuk 380-702, Korea

요 약

최근 박스 오피스 상위권 작품들의 상당수가 Stereoscopic 3D 상영을 병행하고 있으며 삼성, LG 등 세계 우수 가전 업체들이 3DTV 판촉에 열을 올리고 있다. 그러나 사람마다 양쪽 눈동자 간격이 다르고 시청 거리와 위치도 개인마다 다르다는 점을 무시한 채 동일한 양안 시차로 제작된 3D 콘텐츠를 시청하게 될 경우 실세계에서 느끼는 입체감과 커다란 괴리가 발생하게 되어 극심한 시각 피로와 두통을 유발하게 된다. 이를 해결하기 위해 본 논문에서는 양안 시차와 시청 거리를 반영하여 입체 콘텐츠를 실시간으로 보정, 재생하는 S3D 렌더링 시스템을 제안 및 구현하고 그 연산 복잡도를 분석한다. 분석 결과 Optical Flow 알고리즘 블록은 한 프레임당 수행 시간이 최대 732초에 이르러 반드시 하드웨어 가속기 형태로 전용칩화할 필요가 있음을 확인하였고 Warping 알고리즘 처리 블록도 프레임당 최대 5.7초의 시간이 필요해 HD급 또는 1080p Full HD 화면 재생을 위해서는 함께 전용칩화 할 필요가 있음을 확인하였다.

ABSTRACT

Recently, most of movies top-ranked in the box office are screening in Stereoscopic 3D, and the world's leading electronics companies such as Samsung and LG are getting the hots for 3DTV sales. However, each person has different binocular disparity and different viewing distance, and thus he or she feels the severe visual fatigue and headaches if he or she is watching 3D content with the same binocular disparity, which is very different from things he or she feels in the real world. To solve this problem, this paper proposes and implement a 3D rendering system that correct the disparity of 3D content by reflecting binocular distance and viewing distance. Then, the computational complexity is analyzed. Optical-flow and Warping algorithms turn out to consume 732 seconds and 5.7 seconds per frame, respectively. Therefore, a dedicated chip-set for both blocks is strongly required for real-time HD 3D display.

키워드 : 스테레오 3D, 연산 복잡도, 실시간 영상 처리, 하드웨어 연산 가속기

Key word : Stereoscopic 3D, Computational Complexity, Real-time Image Processing, Hardware Acceleration

접수일자 : 2013. 08. 07 심사완료일자 : 2013. 08. 21 게재확정일자 : 2013. 09. 16

* Corresponding Author Jaesung Lee(E-mail:jaesung.lee@ut.ac.kr, Tel:+82-43-841-5163)

Department of Electronic Engineering, Korea National University of Transportation, Chungbuk 380-702, Korea

Open Access <http://dx.doi.org/10.6109/jkice.2013.17.12.2865>

print ISSN: 2234-4772 online ISSN: 2288-4165

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Copyright © The Korea Institute of Information and Communication Engineering.

I. 서 론

2009년말 영화 ‘아바타’의 개봉, 2010년 ‘FIFA 월드컵’ 스테레오 3D (Stereoscopic 3D 또는 S3D) 중계를 시작으로[1] 입체 콘텐츠에 대하여 많은 사람들이 관심을 갖게 되었고 최근에 이르러서는 박스 오피스 상위권 작품들의 상당수가 S3D 상영을 병행하고 있으며 삼성, LG 등 세계 유수 가전업체들은 3DTV 판촉에 열을 올리고 있는 등 3D 입체 콘텐츠는 빠르게 우리의 생활의 한 부분으로서 자리매김을 하고 있는 중이다. 입체 영상은 양 눈에 각각 서로 다른 영상 신호로 전달되어 뇌에서 공간에 대한 개념과 함께 재구성되는 미디어이다. 그러나 이러한 인간의 입체 인지 시스템(3D perception system)은 극도로 민감하기 때문에 하나의 카메라로 촬영된 기존 2D 영상을 시청하는 경우와 달리 시청자의 양안 시차(binocular disparity)와 시청 거리가 콘텐츠의 제작 당시 적용된 기준 값들과 조금만 달라지더라도 실 세계에서 인간이 느끼는 입체감과 커다란 괴리가 발생하게 되고 이에 따라 극심한 시각 피로와 두통을 유발하게 된다.

쉽게 설명하자면, S3D 촬영 전용 양안 카메라와 똑같은 양안 시차를 갖는 사람이 피사체로부터 떨어진 거리와 똑같은 거리를 두고 영상을 시청해야 자연스럽게 편안한 입체감을 느낄 수 있는 것이다. 하지만 사람이 로봇이 아닌 이상 그러한 경직된 환경에서 시청은 불가능하다. 남녀노소 또는 인종 간 양쪽 눈동자 간격이 다르고 시청 거리와 위치도 달라 그에 따른 초점각 변화가 지속적으로 발생할 수 있기 때문에 재생중인 S3D 콘텐츠를 실시간으로 보정하는 작업을 해줘야 시각 피로 유발을 막을 수 있다. 하지만 현재 출시되고 있는 S3D 디스플레이 장치들은 기술적 한계로 그러한 기능을 제공하지 못하고 있다. 특히 어린 아이들을 대상으로 하는 유아 애니메이션 콘텐츠들이 어른 기준의 시차에 맞춰 제작 및 재생되고 있는 현실은 매우 안타까운 상황이다. 이러한 문제들 때문에 S3D 기기의 판매가 부진하고 해당 콘텐츠 시장도 성장이 더딘 것으로 알려지고 있다.

그림 1. S3D R-C anaglyph 영상 예제(자료 제공: 디즈니 엔터프라이즈)

Fig. 1 S3D R-C anaglyph of example video(Source: Disney Enterprise, Inc.)

좀 더 구체적인 예를 들어 설명을 하면, 그림 1 은 물체에 근접함에 따른 시차(disparity)의 변화를 표현한 것인데 상단의 그림인 경우 인간이 자연스럽게 느끼는 시차를 갖는 근접 장면(scene)을 보여주고 있는 반면 하단의 그림의 경우는 사람의 인지모델을 고려하지 않고 오직 콘텐츠 제작 장비의 시점에서 촬영된 장면을 보여주고 있다. 이렇게 사람의 인지 모델 특성에 대한 연구를 무시한 채 기계적 관점으로 제작된 콘텐츠를 그대로 보여지게 되면 근접 장면의 경우 시차가 과도해지면서 깊이감에 대한 왜곡이 발생하여 심각한 어지럼증을 유발하게 된다. 물론 여럿이 함께 보는 대형 스크린의 3D 영화는 현재로서 해결책이 없지만 혼자 보는 3D TV나 3D 게임의 경우는 시청자 1인의 시차를 3D 콘텐츠에 반영하기 위한 연구는 최근 디즈니사(Disney Interactive Studios)를 비롯해 nVidia 연구소 등에서 연구가 진행되고 있으며[2-7] 국내에서도 KAIST 등 몇몇 대학에서 관련 연구를 수행해 학술대회 중심으로 논문들을 발표하고 있다[8-12]. 그러나 지금까지 발표된 연구 내용들은 영상 콘텐츠의 재생산(Reproduction)을 염두해둔 연구로서 실시간 렌더링을 고려하지 않고 새 버전의 콘텐츠로 출시하기 위한 연구들이다. 쉽게 말하면 어린이를 위한 또는 소형 디스플레이디바이스용 버전의 영상 콘텐츠를 다시 출시하기 위한 연구들이다. 이러한 기존 연구들은 특정 세대 층의 시청자용 버전으로 제작이 되는 또 다른 버전의 콘텐츠이기 때문에 개개인의 양안간 거리나 시청거리의 변화를 반영할 수 없어 근본적인 시청 피로 제거에는 큰 효과를 얻을 수 없다.

본 논문에서는 이러한 기존 연구의 한계를 극복하고자 실시간 처리를 위한 시청 피로 저감형 S3D 렌더링 시스템을 제안하고 그 연산 복잡도를 분석한다. II 장에서는 제안하는 시스템의 구성을 위한 알고리즘파이프라인을 구성하고, III 장에서는 실제 시스템으로 구현한다. IV 장에서는 파이프라인 알고리즘 블록별 연산 복잡도 분석을 통해 실시간 구현 가능성과 그 방안에 대해 검토하고 V 장에서 결론을 내린다.

II. 본 론

제 2절에서는 제안하는 시청피로 저감형 S3D 렌더링 시스템의 구현을 위해 필요한 영상 처리 알고리즘 파이프라인을 구성하고 각 블록에 대해 자세히 설명한다.

2.1. 눈동자 인식(Eye-Pupil Detection)

본 논문이 제안하는 S3D 렌더링 파이프라인의 첫 단계는 시청자의 양쪽 눈동자(Eye-pupil)의 위치를 정확히 검지하는 것이다.

그 원리는 두 개의 HD급 USB 카메라(스테레오 카메라)를 3D 평판 TV 의 양쪽 끝 모서리에 설치한 후 눈동자의 정밀한 위치를 파악하여 양안간 거리를 계산하고 삼각 측량 방법으로 시청거리를 계산하는 것으로서 다음 그림과 같이 3단계의 세부 단계로 나뉜다.

그림 2. 눈동자 인식 3단계 과정

Fig. 2 Three Steps of Eye-pupil Detection

2.1.1. 얼굴 영역 검출(Face Detection)

눈동자 인식의 근간이 되는 알고리즘은 영상 이미지 속에서 원의 형태를 찾는 CHT(Circular Hough Transform) 알고리즘 또는 이를 개선한 알고리즘들이다[13,

14]. 그러나 촬영된 시청자의 얼굴과 배경 이미지에 바로 그 알고리즘들을 적용하면 원으로 추정되는 눈동자가 아닌 다른 많은 객체들도 검지되기 때문에 이들 사이에서 눈동자를 구별하기가 쉽지 않다. 따라서, 얼굴 영역을 먼저 파악한 후 그 안에서 다시 눈 영역을 파악하고 눈영역내에서 눈동자 인식 과정을 수행하여 양쪽 눈동자들의 위치를 인지한다.

얼굴 영역을 파악하는 알고리즘으로 널리 잘 알려진 Viola-Jones 알고리즘[15]을 적용하여 얼굴영역을 감지하는 것을 우선적으로 생각해볼 수 있지만 TV 앞 화면에서 시청을 하는 경우 TV에서 나오는 역광으로 인해 luminance 에 영향을 끼칠수 있다. 따라서, 본 논문에서는 luminance 변화에 강한 MCT(Modified Census Transform) 알고리즘[16]을 사용한다. MCT 알고리즘은 Viola-Jones 알고리즘과 마찬가지로 Machine Training 방식의 알고리즘으로서 큰 규모의 training set 을 사용하고 20x20 윈도우 슬라이딩(windows sliding) 방식으로 한픽셀씩 좌에서 우로 이동하면서 윈도우 영역 내의 픽셀 패턴 분석 결과를 cascade classifier 에 통과시키면서 판단을 진행을 한다는 점도 동일하다. 다만 전처리(pre-processing) 단계에서 3x3 마스크를 슬라이딩 이동시키며 평균값을 계산(averaging)한 후 각 픽셀값이 이 평균값보다 크면 1, 작으면 0으로 세팅 한 후 그 9 개의 이진수 패턴을 나열 후 그 자체를 픽셀값으로 저장하면 전체 이미지의 밝기가 균일하게 된다. 이 위에 얼굴 영역 검지를 위한 20x20 윈도우가 지나가면서 패턴 분석 및 평가를 수행하면서 얼굴 영역을 찾게 된다.

2.1.2. 눈 영역 검지(Eye-Zone Classification)

눈 영역도 Viola-Jones 알고리즘과 같은 training 방식으로 찾을 수도 있지만 얼굴 영역검지율에 비하면 검지율이 상당히 떨어지는 것으로 알려져 있다. 따라서, TV 시청환경에서 90% 이상의 높은 인식율을 보이는 얼굴 영역 인식 알고리즘으로 얼굴 영역 이미지를 확보한 후 그 위에 다음 그림과 같은 비율 마스크를 씌워 눈 영역을 유추하면 검지율이 얼굴 인식율과 거의 동일한 것으로 실험 결과를 얻었다. 대략적인 눈 영역은 얼굴을 위에서부터 아래로 1/3 씩 나누었을 때 그 중간 부분에 위치하나 좀 더 정밀하게 찾고자 할 때는 아래 그림의 마스크를 씌워 좀 더 정확하게 눈 영역을

잡아낼 수 있기 때문에 다음 단계(눈동자 감지)가 한결 수월해진다.

그림 3. 얼굴 비율 마스크
Fig. 3 Face Ratio Mask

그림의 상단 양쪽에 짙은 색 영역이 눈 영역이다.

2.1.3. 눈동자 감지(Eye-Pupil Detection)

검지된 눈 영역 세그먼트에 대하여 이미지 속에서 원의 형태를 찾는 CHT(Circular Hough Transform) 알고리즘 또는 이를 개선한 알고리즘[13, 14]를 적용한다. 이 알고리즘들을 적용하기 위해서는 먼저 에지 인지(edge detection)을 수행하여 이미지상의 윤곽선 정보를 제공해줘야 한다. 다양한 에지 인지(edge detection) 알고리즘 중 본 논문에서는 canny edge detection 알고리즘을 사용한다. 본 알고리즘은 연산 복잡도는 높으나 실제 에지를 놓치거나 찾은 에지가 real 에지가 아닌 (Spurious Response) 경우가 적어 검출 어려움이 낮으며, 에지 점들의 위치가 정확히 측정되어(Well Localized) 찾아진 에지가 real 에지와 가깝고(이 특성은 찾은 에지 점과 진짜 에지 점의 중심간의 거리를 측정해 판단), real 에지 점당 한 점만 반환하여 혼동없이 처리가 가능하기 때문이다.

canny 에지 검출 과정이 끝난 이진(binary) 이미지에 대하여 화이트 픽셀들마다 그것을 중심으로 원들을 그려나가고 그 원들이 중첩이 (Hough space 상에) 많이 발

생한 지점을 원 그림의 중심으로 인지하므로써 CHT 과정을 마치게 된다.

그림 4. Canny 에지 검출 과정 및 CHT
Fig. 4 Several Steps of Canny Edge Detection and CHT

아래 그림은 대략적으로 평가한(1/3 방법) 눈 영역에 대해 CHT 알고리즘을 적용한 결과이다. 좌측의 canny 에지 검출 알고리즘으로 얻어진 윤곽선 정보로부터 눈으로 의심되는 원들을 우측 원이미지위에 빨간색으로 표시한 것이다.

그림 5. 1/3 영역에 대한 눈동자 검출 예제
Fig. 5 Eye Detection Examples on the 1st One-Third Area

상기 그림의 결과에서 확인할 수 있듯이 대략적으로 인지된 넓은 범위의 눈 영역 이미지에서는 눈동자로 의심되는 원들이 너무 많이 검출되어 판단에 어려움이 있다. 따라서, 그림 3의 eye-zone 마스크를 씌우면 아래와 같이 3, 4, 5, 6 번의 원만 남게 되어 눈동자 판단이 용이해진다.

그림 6. 비율 마스크의 눈동자 영역에 대한 눈동자 검출 예제
Fig. 6 Eye-Pupil Detection Examples on Area Detected by the Ratio Mask

본 예제에서는 눈 영역(eye-zone)에서 눈동자로 의심되는 원이 좌우 각각 2개씩 발견이 되었다. 이중 진짜 눈동자를 검출하는 과정을 다음과 같이 제안한다. 먼저 주어진 좌우 각각의 눈 영역에 대해서 히스토그램 분포를 먼저 파악 후 누적 픽셀 개수가 10% 인 지점의 픽셀 개수를 Threshold 값으로 정하여 이미지를 역이진화한다.

그림 7. 눈 영역(eye-zone)에 대한 히스토그램 분포
Fig. 7 Histogram of Eye-Zone

그러면 다음의 그림과 같이 눈동자 부분이 화이트 픽셀 밀도가 높은 형태의 이진화된 이미지를 얻을 수 있으며 이 결과로부터 검지된 원 영역 내의 픽셀 분포가 $\text{Dark pixel ratio} < \text{defined ratio (threshold. for now, 30\%)}$ 이면 눈동자로 판정을 내리도록 하면 99%의 확률로 눈동자로 인식되었다.

그림 8. 눈 영역(eye-zone)에 대한 히스토그램 분포
Fig. 8 Histogram of Eye-Zone

상기 과정은 원이 아닌 동일 지름의 사각형으로 영역 내에서 판정을 하도록 하여 원영역내의 픽셀 위치를 계산해야 하는 연산 복잡도를 줄였다.

그림 9. 눈 영역(eye-zone) 및 검지된 눈동자
Fig. 9 Eye-Zone and Detected Eye Pupils

2.2. 눈동자(Eye Pupils) 간 간격(Disparity)과 시청 거리 계산

우선 체크 보드를 이용하여 스테레오 카메라에 대한 교정(calibration)을 수행하면서 초점 거리(focal lengths) f , 주점(principal points), 두 카메라 렌즈 중심 간 거리(baseline distance) T 를 계산한다.

삼각 측량은 두 개의 평행 카메라를 이용하여 거리를 측정하는 표준 방법이다. z 축이 TV 화면으로부터 떨어진 거리, 즉, 시청거리를 측정한다고 할 때 공간상의 임의의 점 (X, Y, Z) 가 스테레오 카메라로 촬영된 후 각각 좌, 우 이미지 상에 좌표 (X_L, y) 와 (X_R, y) 로 나타난다면 $X_L = (f/Z)*X$, $y = (f/Z)*Y$, $X_R = (-f/Z(T-X))$ 식이 성립한다. 이를 세 식을 상호 대입해 계산을 수행하면 시청 거리 $Z = T*f/(X_L - X_R)$ 식이 성립한다.

그림 10. 삼각 측량
Fig. 10 Triangulation

본 수식을 사용하여 한 눈동자의 z축상의 거리(Z)를 계산한다. 그런 다음 앞서 이미 측정된 X_L 과 X_R 값을 이용해 X 값을 구하고 calibration을 통해 Y 값을 구하여 최종적으로 x 축과 y 축 좌표로 피타고라스 정리를 이용하여 주점으로부터 눈동자까지의 거리를 계산한다. 본 과정을 좌우 눈동자에 각각 적용함으로써 두 눈동자간의 거리(disparity)와 시청 거리가 정해진다.

그림 11. 시청 거리 측정.
Fig. 11 Measurement of Watching Distance

아래의 표는 눈동자 간 간격 및 시청 거리를 pupilo-meter 와 레이저 거리 측정기를 이용해 실제 측정한 결과와 본 논문에서 사용한 방법으로 측정한 값에 대한 비교표이다. 5가지의 다른 환경에서 각 경우별로 100회 이상 눈동자 감지 및 영상을 통한 눈동자 간격 측정 결

과에 대한 평균값과 표준 편차를 나열하였다. 실험 결과로부터 오차율이 0.25% 이하임을 알 수 있었다. 참고로 카메라간의 사이 간격은 30cm로 하였다. 아래 표에는 기재하지 않았지만, 시청 거리의 경우 약 2%의 오차율을 보였다.

표 1. 눈동자 간격 실험 데이터와 측정 데이터간 비교표
Table. 1 Comparison of measured distance between eye pupils

카메라거리	초점거리	실제 눈동자 간격	측정된 눈동자 간격	측정 표준편차	변이계수
60cm	60cm	64.5mm	62.5mm	0.14mm	0.22%
60cm	1m	64.5mm	63.2mm	0.12mm	0.19%
60cm	5m	64.5mm	64.2mm	0.13mm	0.20%
2m	2m	64.5mm	64.1mm	0.16mm	0.25%
2m	6m	64.5mm	64.3mm	0.16mm	0.25%

2.3. 스테레오 이미지 렌더링(S3D Image Rendering)

2.2 절까지 과정을 통해 얻어진 disparity 정보와 시청 거리 정보를 반영하여 좌우 스테레오 이미지를 재구성(rendering) 해주는 과정이 필요하다. 본 과정은 매크로 블록별로 좌우 이미지에 대한 Optical-Flow 벡터[17]를 계산한 후 disparity 정보와 시청 거리 정보에 비례 매핑(mapping)하여 그 벡터값을 조절한 후 그 조절된 벡터값 만큼 매크로블록을 찌그러트려(Warping) S3D 이미지를 재구성하는 원리이다. 이렇게 함으로써 스테레오 이미지의 depth 가 조절되어 깊이감이 일정 범위 내에서만 조성되도록 하여 시청피로를 저감시키게 된다. 다음 그림은 지금까지 설명한 과정을 도시하고 있다.

그림 12. Optical-Flow 벡터 계산 및 워핑 과정
Fig. 12 Optical-Flow Calculation and 3D Warping Process

참고로 본 장의 눈동자(Eye Pupils) 간 간격(Disparity) 거리 계산 방법을 응용하면 수평선상의 눈동자간 거리뿐만 아니라 비스듬한 수평선상의 눈동자측, 시청자가 비스듬한 자세나 누워서 볼 때도 양안 간격을 산출할 수는 있다. 하지만 3D 콘텐츠 자체가 두 눈이 수평하다는 가정하에 제작된 것이기 때문에 상기의 워핑 과정을 통해 시청자의 양안 시차에 맞춘 보정된 콘텐츠를 생성할 수가 없다. 양안 시차가 일치하는 3D TV 콘텐츠를 시청하는 경우도 누워서 보게 되면 입체효과를 제대로 느낄 수 없이는 마찬가지이다. 본 연구는 정자세에서 시차가 다른 경우 이를 보상해 입체감을 살리는 연구이며 시청자의 자세에 따른 입체감 외곽 해결을 위한 연구는 별개의 연구 영역이다. 일반적으로 3D 콘텐츠를 정자세가 아닌 자세로 보면 매우 어지럽기 때문에 그렇게 시청하는 사람은 매우 드물 것으로 판단한다.

III. S3D 영상 재생 시스템 구현

제 2장에서 설명한 알고리즘을 시스템으로 구현하기 위한 하드웨어 구조를 다음과 그림과 같이 구성하였다.

그림 13. S3D 영상 재생 시스템 구성 및 블록 다이어그램

Fig. 13 S3D Video Rendering System Architecture and Block Diagram

미니 PC 보드(2가지 모델의 머신을 준비함)에 각각 리눅스, 윈도우즈 운영체제를 설치하고 HDMI 포트를 통해 3D TV로 영상 신호를 출력하도록 케이블을 연결하였다. 향후 연산량이 많아 s/w로 처리가 어려운 알고리즘 블록을 FPGA로 하드웨어 가속 처리를 위한 FPGA 보드도 일단 연결해두었다. 알고리즘 블록별로 Open CV로 프로그래밍하여 Eye Detection, Calibration, Disparity Detetion(Optical Flow 벡터 계산), Warping Image Processing으로 이어지는 전체 파이프라인을 완성하였다. 또한, 보드 제어 및 알고리즘 블록별 처리 속도 측정을 위한 관리 프로그래밍도 병행하였다.

다음 그림은 본 시스템을 이용하여 좌우 예제 이미지(a)간 optical flow를 구한 후 그에 비례하여 gain을 0.15만큼 조절(warping) 후 재구성한 즉, disparity가 조절된 이미지(b)를 나타내고 있다.

(a)

(b)

그림 14. S3D 영상의 disparity 조절 과정 (a) 원래의 스테레오 이미지(레드/시안) (b) 양안 시차가 조절된 스테레오 이미지(레드/시안) (c) 원 스테레오 이미지의 옵티컬 플로우
Fig. 14 Disparity Control Process of S3D Video (a) original stereo images (red/cyan anaglyph) (b) disparity controlled stereo images (red/cyan anaglyph) (c) optical flow of the original stereo images

(a)와 (b) 간 깊이감이 바뀐 것을 알 수가 있다. 일부 버그가 있긴 하지만 이는 일괄적인 gain 값 control에 따른 것으로 각 매크로 블록별 gain 값을 어떤 값으로 매핑(mapping)조절하느냐는 별개의 연구 분야이며 본 논문에서는 다만 gain 값을 조절하여 disparity 변화가 가능하다는 것을 확인하고 이때 필요한 알고리즘의 연산 복잡도를 분석하는 것이 목표이다. 다음 절에서 그 연산 복잡도를 분석하였다.

IV. 연산 복잡도(Time Complexity) 분석 결과

720p와 1080p 크기의 테스트 영상들에 대해 각각 Linux CentOS와 Windows XP 머신상에서 연산 복잡도를 측정하였다. Linux와 Windows에 사용된 프로세서 성능은 다음과 같다.

<Linux 머신 사양>

Processor : Intel i5 750 CPU @ 2.67GHz

Installed memory (RAM) : 8.00 GB

<Windows 머신 사양>

processor: intel i5- 3450S CPU @ 2.80GHz

Installed memory (RAM) : 4.00 GB

표2는 각 머신에서 수행한 S3D 렌더링 파이프라인의 각 알고리즘 블록별 수행요구시간 평균을 정리한 표이다.

표에서 Camera initialization은 USB 카메라 모듈로부터 시청자 얼굴 영상을 얻어내는 부분까지를 말하는데 Windows머신에서 상대적으로 값이 좀 큰 이유는 데이터 USB 드라이버의 data type의 차이에서 발생하는 문제인 것으로 파악이 되었다. Distance measurement는 Eye detection된 이미지 상 위치를 입력으로 받아 3차원 공간상의 위치로 바꿔주는 단순 계산이어서 연산 수행시간이 작은 편이다. Gain control (optical flow vector mapping 과정)이나 Warping 과정도 요구 복잡도가 그리 크지 않은 것으로 판명되었다.

반면, Optical Flow 계산은 상당한 시간이 요구됨을 알 수 있었다. 따라서, Optical Flow 부분은 FPGA로 하드웨어 가속화가 필요함을 알 수 있다.

표 2. 리눅스 머신과 윈도우 머신 상에서 각 알고리즘의 수행 요구 시간

Table. 2 Execution time of each algorithm in Linux and Windows Machine

	Execution time in Linux machine (in sec)		Execution time in Windows machine (in sec)	
	720p	1080p	720p	1080p
Camera Initialization	3.2433	3.267	11.94	11.66
Eye detection	0.3041	0.3095	1.24	1.231
Distance measurement	0.00019	0.000356	0.000414	0.000455
Optical flow	243.588	732.07	179.88	524.43
Gain control	0.0487	0.148	0.0155	0.0457
Warping	0.5363	1.4652	0.89776	5.702

V. 결 론

시청자가 TV를 보면서 자주 움직이지 않는 이상 Camera Initialization 과 Eye detection 은 실시간 처리가 필요한 것은 아니다. 따라서 이 두 과정을 제외하고 보면 Distance measurement 와 Gain control 은 초당 30프레임이상의 화면 재생을 위한 실시간 처리가 가능한 것으로 판단되었다. 하지만 Optical Flow 와 Warping계산은 상당한 시간이 요구되어 실시간 처리가 불가능한 것을 알 수 있었다. 따라서, Optical Flow 벡터 계산 부분은 하드웨어 가속기 형태로 반드시 전용칩화할 필요가 있음을 알 수 있었다. 또한 Warping 알고리즘 수행도 HD급 또는 1080p Full HD 화면 재생을 위해서는 반드시 전용칩화 할 필요가 있음을 확인하였다.

상업적 측면에서도 PC 와 같은 보드 형태의 시스템으로 구현할 경우 큰 부피에 고비용, 고전력 소모의 문제가 있어 3DTV 등 관련 제품의 상용화에 걸림돌이 될 것이다. 시장은 별도의 set-top box 형태로 제작된 시스템보다 3DTV 나 3D 모니터 등에 내장된 형태를 선호할 것이 자명하며, 스마트폰, 태블릿 PC 등 모바일 기기 부문도 S3D 콘텐츠 재생 기능이 내장된 제품에 대한 수요가 발생할 것 또한 자명하기 때문에 본 연구의 필요성은 명약관하하다. 따라서 시청피로 저감형 S3D 영상 시스템을 위한 반드시 전용(application-specific)의 렌더링 프로세서(rendering processor)의 개발이 필요하다. 향후 후속 연구에서는 이 두 알고리즘의 하드웨어 가속 처리를 위한 회로를 HDL 로 설계 후 FPGA 로 구현할 예정이다.

감사의 글

이 논문은 2012년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. 2012R1A1A1038515)

REFERENCES

- [1] Tekalp, A. M., Smolic, A., Vetro, A., and Onural, L., Eds. "Special issue on 3-D Media and Displays," vol. 99, 4. *Proceedings of the IEEE*. 2011.
- [2] Land, M., Hornug, A., Wang, O., Poulakos, S., Smolic, A., and Gross, M. H. "Nonlinear disparity mapping for stereoscopic 3D," *ACM Trans. Graph.* 29, 4. 2010.
- [3] Oskam, T., Hornug, A., and Bowles, H., Mitchell, K., and Gross, M., "OSCAM-Optimized Stereoscopic Camera Control for Interactive 3D," *ACM Trans. Graph.* 29, 4. 2010.
- [4] Gateau, S., and Neuman, R. 2010. "Stereoscopy from xy to z," In *SIGGRAPH ASIA Courses*.
- [5] Koppal, S. J., Zitnick, C. L., Cohen, M. F., Kang, S. B., Ressler, B., and Colbrun, A. "A viewer-centric editor for 3D movies," *IEEE Computer Graphics and Applications* 31, 1. 2011.
- [6] Didyk, P., Ritschel, T., Eisemann, E., Myszkowski, K., and Seidel, H.-P. "A perceptual model for disparity," *ACM Trans. Graph.* 30, 4. 2011.
- [7] Heinzele, S., Greisen, P., Gallup, D., Chen, C., Saner, D., Smolic, A., Burg, A., Matusik, W., and Gross, M. H. "Computational stereo camera system with programmable control loop," *ACM Trans. Graph.* 30, 4. 2011.
- [8] Seok-Hyon Kang and Hyung-Ki Hong, "In Watching 3D Stereoscopic Display Using the Binocular Disparity, the Effect of Pupillary Distance of Adults and Children on the Perception of 3D Image," *Journal of Korean Ophthalmic Optics Society*, vol. 16, no. 3, pp.299~305, 2011.
- [9] Jae-Ho Lee and Chang-Ick Kim, "Stereoscopic Image Generation with Optimal Disparity using Depth Map Preprocessing and Depth Information Analysis," *Journal of broadcast engineering*, vol. 14, no. 2, pp.164~177, 2009.
- [10] Dong-Wook Kim, Kwang-Hoon Lee, Sung-Kyu Kim, and Eun-Young Chang, "Simulation of Distortion in Image Space due to Observer's Rotation Movement in Stereoscopic Display, and Its Correction," *Korean Journal of Optics & Photonics*, vol. 20, no. 2, pp.65~144, 2009.
- [11] Cheon Lee and Yo-Sung, "Boundary Noise Removal in Synthesized Intermediate Viewpoint Images for 3D Video," *Journal of Broadcast Engineering*, vol. 9, no. 4, pp.109~112, 2008.
- [12] Jong-Min Lee, Dae-Hyun Kim, Jong-Soo Choi, "A Study on Stereo Matching Algorithm using Disparity Space Image,"

- Journal of the institute of electronics engineers of Korea, vol. 41, no. 6, pp.9-18, 2004.
- [13] Atherton, T.J., D.J. Kerbyson. "Size invariant circle detection." *Image and Vision Computing*. Volume 17, Number 11, 1999, pp. 795-803.
- [14] Davies, E.R., Machine Vision: Theory, Algorithms, Practicalities. 3rd Edition. San Francisco, CA: Morgan Kauffman Publishers, 2005, Chapter 10.
- [15] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," *Proceedings of the Computer Vision and Pattern Recognition*, 2001.
- [16] B.Froba, A. Ernst, "Face Detection with the Modified Census Transform", *Automatic Face and Gesture Recognition*, 2004.
- [17] Baker, S., and Matthews, I. "Lucas-Kanade 20 years on: A unifying framework", *IJCV* vol.56, no.3, pp.221-255, 2004.

이재성(Jaesung Lee)

1999년 2월 아주대학교 전자공학부 학사
2001년 2월 아주대학교 전자공학과 석사
2008년 8월 서울대학교 전기컴퓨터공학부 박사
2001년 2월 ~ 2011년 8월 한국전자통신연구원(ETRI) 선임연구원
2011년 9월 ~ 2013년 9월 한국교통대학교 조교수
2013년 10월 ~ 현재 한국교통대학교 부교수
※관심분야: 멀티미디어 SoC 설계, 3D 영상처리