

운동추적을 이용한 모션 깊이맵 생성

*김원희 **김만배

강원대학교 IT대학 컴퓨터정보통신공학과

*showman654@kangwon.ac.kr, **manbae@kangwon.ac.kr

Motion Depth Map Generation using Motion History

*Kim, Won-Hoi **Kim, Manbae

Kangwon National University

요약

본 논문에서는 애니메이션, 드라마, 영화 등 영상으로 제작된 콘텐츠를 OpenGL를 사용하여 3D영상으로 재구성한다. 먼저 현재영상과 이전영상의 움직임의 차이로부터 운동 추적 데이터를 사용하여 모션 깊이맵을 생성한다. 그 깊이맵을 사용하여 OpenGL에서 사용하는 텍스처 맵핑으로 영상을 출력하고 3D 영상을 구현하기 위해 좌영상과 우영상을 생성하여 3D 입체영상을 만든다.

1. 서론

오늘날 소비자들은 3D 영화, 3D 애니메이션, 3D TV와 같은 눈앞에서 펼쳐지는 실감 있는 콘텐츠를 소비하고 있다. 하지만 이러한 3D 콘텐츠들은 양안식 입체 카메라 등 일반인들이 쉽게 제작하기 어려운 과정을 요구한다. 따라서 2D이미지에서 3D영상을 만드는 기술을 제안하면서 일반인들도 쉽게 입체영상 콘텐츠를 제작하고 소비 할 수 있는 방법을 제안한다.

2D영상의 깊이맵(depth map)을 생성한 후에, 3D 입체영상을 제작하는 3D변환은 그동안 수많은 연구가 진행되어 왔다. 대부분의 방법들은 컬러의 특성을 이용하여 깊이맵을 구하고, 이를 3D 변환에 활용한다[1,2]. 정지 영상에서 깊이 단서를 구한 후에, 복합 단서들의 통합으로 깊이맵을 구한다. 또한 비디오의 모션을 활용하여 모션 깊이 데이터를 얻은 후에, 정지영상의 깊이와 통합하여 깊이맵을 얻기도 한다. 다음 식 (1)에서 보는 것처럼, 일반적으로 컬러, 모션 및 기타 데이터 x로부터 얻은 깊이맵을 함수 f에 의해서 통합하면 최종 깊이맵이 얻어진다.

$$f: (\text{color}, \text{motion}, x) \Rightarrow \text{depth} \quad (1)$$

비디오에서 모션을 추출하여 모션 깊이맵을 찾는 연구는 꾸준히 진행되어 왔다. Konrad 등은 연속 프레임간의 국부 운동을 구한 후에 모션 깊이맵을 생성한다[3]. Kim 등은 컬러 분할 및 KLT (Kanade-Lucas-Tomasi) 특징점 트래커를 이용하여 모션 정보를 얻는다[4]. Xu 등은 모션을 측정하기 위해서 KLT의 광유 계산법을 이용한다[5]. Po 등은 기본적으로 블록기반으로 모션을 예측하고, 영역 분할을 이용하여 모션 깊이맵을 얻었다[6]. 대부분의 연구는 모션값을 구하기 위해서, BBME, 광유 등을 사용한다. 이것의 단점은 낮은 정확도와 이를 보완하기 위해 영역 분할 기술을 필요로 한다.

본 논문의 구성은 다음과 같다. 2절에서는 본 논문에서 제안하는

방법의 개요 및 제작 단계들을 설명한다. 그리고 3절에서는 실험 결과를 보여주며 마지막 4절에서는 결론을 맺는다.

2. 제안 방법

운동추적 MA(motion accumulator) 데이터를 H_t 라고 가정할 때 H_t 를 구하는 식은 다음과 같다.

$$H_t(x, y, t) = \begin{cases} \tau & \text{if } D(x, y, t) = 1 \\ \max(0, H_t(x, y, t-1) - 1) & \text{otherwise.} \end{cases} \quad (2)$$



그림 1 (a) 5장 이미지 입력영상, (b) MA 영상

얻어진 H 값은 여기서 z값 역할을 한다. 얻어진 H버퍼의 값들은 격차가 커서 입체영상을 생성해 줄 때 MA값이 포함된 부분과 MA 값이 포함되지 않은 부분들의 차이가 커서 이질감을 느낄 수 있다. 이 이질감을 없애기 위해서 MA의 한 픽셀과 이웃한 8근방의 픽셀간의 평균값을 구해 새로운 MA 버퍼를 생성하여 새로운 MA 값으로 z값을 설정해 주어서 입체 영상을 만든다.

읽어온 BMP파일을 OpenGL에 있는 텍스처맵핑을 통해 Left 이미지와 Right 이미지를 만든다. 그리고 각 각의 이미지들은 각 각 Left 버퍼와 Right 버퍼에 저장한다. Left 이미지는 Left 버퍼에 있는 내용을 그

대로 출력하지만 Right 버퍼는 위에서 구한 z값을 이용하여 출력해 준다



그림 2 (a) Left 이미지, (b) z값을 이용한 Right 이미지

만들어진 Left 버퍼와 Right 버퍼를 통해 Top-Bottom 이미지를 만든다. Left 이미지가 위로 가도록 설정해 주고 Right 이미지가 아래로 오도록 설정해 준다. 각 각의 이미지 모든 버퍼를 출력 하지 않고 짝수 줄인 0, 2, 4, ..., 512 만 출력해준다.



그림 3 (a) Top-Bottom 이미지

3. 실험

입체영상을 만들기 위해 영상을 프레임 단위로 2D BMP 이미지로 나눈 뒤 한 장씩 BMP파일을 읽어온다. 입체영상을 만들기 위해서 현재 영상과 이전 영상의 깊이맵을 저장하는 MHI버퍼를 만들어야 한다. 영상을 Bitmap 이미지로 분할 시켜 OpenGL에 있는 텍스처맵핑을 이용하여 영상을 출력한다. 이 때 텍스처맵핑은 일정한 크기를 가지는 Mesh로 이루어져 있다. 2D 이미지에서 입체감을 주기 위해서는 현재 2D 이미지와 이전 2D 이미지들의 구성성분들의 차이점을 MHI를 사용하여 깊이맵을 얻는다. 입체 영상을 구현하기 위해 Left 이미지와 Right 이미지 각 각 버퍼에 저장한 뒤 Left 이미지라 Top, Right 이미지가 Bottom에 있는 Top-Bottom 이미지를 제작한다. 여기서 Right 이미지는 MHI를 통해 얻은 깊이맵을 사용하여 입체적으로 보일 수 있게 해준다. 만들어진 여러 개의 Top-Bottom 이미지로 영상을 제작한 뒤 3D를 지원해주는 TV를 통해 2D 이미지로부터 만들어진 입체영상을 감상할 수 있다.

제안 방법의 결과물을 만들기 위해, 애니메이션 3종류와 수족관 영상 1종류를 사용하여 24fps를 가지는 입체영상을 제작하였다. 결과는 그림 4에서 확인할 수 있으며 해당 그림은 3D를 지원하는 TV이지만 위 결과물을 통해 움직임이 존재하는 물체들은 입체감 있게 움직이는 걸 볼 수 있다.



그림 4 (a) 3D TV에서 시청한 결과물

4. 결론

본 논문은 2D 이미지를 OpenGL을 통해 3D 입체 영상을 제작하는 기법을 제안하였다. 이 기법으로 기존에 존재하는 2D 영상들을 3D 입체 영상으로 제작하여 시청할 수 있으며 일반인들은 접근하기 어려웠던 입체영상제작을 보다 쉽게 접근 할 수 있다.

이 MA는 타 모션 획득 방법들이 가지고 있는 문제점 또한 있지만, BBME, 광유 등의 방법과 달리 새로운 모션정보를 획득할 수 있는 것을 사용하여 가능성을 보여주는 것이 목적이다. MA는 타 방법과 달리 연산량이 적고, 객체의 운동 방향을 잘 표현할 수 있다. MA는 이동하는 객체의 내부의 운동값이 점진적으로(gradual) 감소하기 때문에, 즉 내부가 일정 시간동안 채워있기 때문에 얻어지는 깊이값 또한 점진적으로 감소하여 3D 입체 시청시에 상대적으로 피로도가 낮게 된다.

운동 추적 데이터를 활용하여 얻어진 운동 깊이맵 생성 방법은 3DTV, 3D 디스플레이서 3D변환 기술의 깊이맵으로 활용이 가능하며, 가상현실, 증강현실에서는 HMD기 등을 이동할 때 움직이는 물체의 3D 효과를 증가할 수 있다. 또한 비디오 감시 시스템에서 움직이는 사람 등의 객체의 움직임을 용이하게 검출할 수 있고, 사람 트래킹에서도 가지고 있는 정보를 제공해준다.

5. 참고문헌

- [1] S. Kim and J. Yoo, "3D conversion of 2D video using depth layer partition," *Journal of Broadcast Engineering*, Vol. 15, No. 2, Jan. 2011.
- [2] J. Jung, J. Lee, I. Shin, J. Moon and Y. Ho, "Improved depth perception of single view images", *ECTI Transactions on Electrical Engineering, Electronics and Communications*, Vol. 8, No. 2, Aug. 2010.
- [3] J. Konrad, F. M. Wang, P. Ishwar, C. Wu, and D. Mukherjee, "Learning-based, automatic 2d-to-3d image and video conversion", *IEEE Tran. Image Processing*, Vol. 22, No. 9, Sep. 2013.
- [4] D. Kim, D. Min and K. Sohn, "A stereoscopic video generation method using stereoscopic display characterization and motion analysis", *IEEE Trans. Broadcasting*, Vol. 54, No. 2, June 2008.
- [5] F. Xu, G. Fr, X. Xie, and Q. Dai, "2D-to-3D conversion based on motion and color merge", *3DTV Conference: The True Vision-Capture, Transmission and Display of 3D Video*, pp. 205-208, May 2008.
- [6] L. Po, X. Xu, Y. Zhu, S. Zhang, K. Cheung and C. Ting, "Automatic 2D-to-3D video conversion technique based on depth-from motion and color segmentation", *IEEE ICSP*, 2010.
- [7] A. Bobick and J. Davis, "The recognition of human movement using temporal templates," *IEEE Trans. Pattern Recognition and Pattern Analysis*, Vol 23, No. 3 Mar. 2001.