MS Kinect 를 이용한 Free Viewpoint TV System 설계

이준협, 양윤모, 오병태
한국항공대학교
junhyeop18@gmail.com, yym064@naver.com, byungoh@kau.ac.kr

Design of Free Viewpoint TV System with MS Kinects

Jun Hyeop Lee, Yun Mo Yang, Byung Tae Oh
Korea Aerospace University

요 약

본 논문에서는 Microsoft 에서 나온 여러 대의 Kinect 를 이용하여 Free Viewpoint TV System 을 구현해 보고자 한다. Kinect 로부터 얻어진 색상 영상과 깊이 영상을 통하여, 실시간으로 두 대의 카메라 사이에서의 가상시점에서 영상이 출력되는 시스템을 설계한다. 또한, 여러 대의 Kinect 를 이용할 때, 간섭현상으로 인해 IR 패턴을 제대로 인식하지 못하여 홀이 생성되는 문제점을 확인하고, Nearest Neighbor 방식과 Inpainting 기법을 사용하여 홀을 제거하는 방식을 소개한다. 실험 결과, 홀의 주변과 비슷한 값으로 홀을 채울 수 있었지만, 홀의 크기에 따라 Edge 경계가 부정확해 지는 현상을 확인할 수 있다.

This paper provides the design and implementation of Free Viewpoint TV System with multiple Microsoft Kinects. It generates a virtual view between two views by manipulating texture and depth image captured by Kinects in real-time. In order to avoid this, we propose the hole-filling scheme using Nearest neighbor and inpainting. As a result, holes generated by interference are filled with new depth values calculated by their neighbors. However, the depth values are not accurate, but are similar with their neighbors. And depending on the frequency of running a Nearest Neighbor method, we can see that edge's border would be shifted inner or outer of the object.

1. 서론

TV 방송기술의 역사를 살펴보면, 기술의 전환점이 두 번이 있었다. 첫째, 흑백 TV에서 컬러 TV로, 둘째, 아날로그 TV에서 디지털 TV로의 변화이다. 지금도 또 한번의 전환점을 만들기 위한 움직임이 일어나고 있다. 차세대 영상 시스템 개발을 위해 많은 전문가들이 TV 방송 기술의 발전 방향을 모색하고 있는 것이 바로 그 노력이다.

대표적인 차세대 영상 시스템으로 깊이 감을 느낄 수 있는 3D 실감 영상에 대한 관심이 점차 커지고 있다. 3DTV는 화면 상에서만 존재하던 2D 영상에서 벗어나, 입체감을 통해 좀 더 현실적인 느낌을 갖도록 하는 영상 시스템이다.

현재, 그 방법의 일환으로, 크게 스테레오(Stereo) 방식과 다시점 깊이영상 (Multiview plus depth)두 가지로 그 분야를 나눌 수 있다.

전자인 스테레오 방식은 실제, 사람의 두 눈에 상이 맺히는 것과 동일한 원리로 이루어 진다. 실제로 이 방법은 Avatar와 같은 3D 영화 및 여러 3D Contents에서 많이 사용되고 있다.

후자를 이용한 방식으로 대표적인 것이 FTV(Free Viewpoint TV System)라 할 수 있다 [1-2]. 이 것을 구현하는 것이 본 논문의 목적이다.

본 논문에서는 깊이 영상 기반의 FTV를 구현하고자, Texture와 Depth정보를 쉽게 얻을 수 있는 MS Kinect를 사용하였다. 이 Kinect 카메라를 통하여, 어떤 방법으로 가상시점에서의 영상을 얻는지를 논할 것이다.

본 논문의 구성은 다음과 같다. 2장에서는 제안하는 시스템의 물리적인 구조 및 제한사항을 설명하고, 3장에서는 FTV 알고리즘을 소개한다. 4장에서는 생성된 가상시점 결과를 살펴본 뒤, 5장에서는 본 논문에 대한 결론을 맺는다.

2. 시스템 구성

본 논문에서는 2대의 Kinect를 사용하였다. 카메라는 Parallel하게 위치시키는 것이 아닌, 각각의 카메라가 물체를 바라보는, 즉, 수렴이 되도록 하는 방향으로 좌, 우에 위치시켰다. 이 때, 각각의 카메라는 같은 높이에 위치시키고, 틀어져 있는 각도는 20° 에서 60° 사이로 맞춰놓았다. 이는 사용자가 원하는 각도로 카메라를 위치시킬 수 있도록 하기 위함이다. Kinect 카메라가 수평을 유지하도록 하는 것이 매우 중요하다. 서로 수평을 유지하고, 물체를 바라볼 경우에, 쉽고 빠르게 좌표계를 변화시킬 수 있기 때문이다. 만약 x, y, z축 모두 틀어져 있는 상황이라면 각각의 각도를 고려하여야 한다. 하지만 카메라의 수직방향, 즉 카메라의 지면 수직방향인, y축만 틀어져 있다고 한다면, 계산은 간단해진다. 자세한 식은 다음 장에서 논의한다.

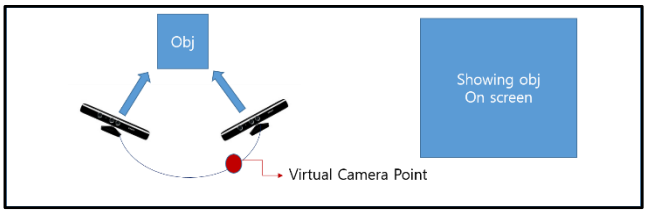

Fig. 1. 시스템 구상도

3. FTV Algorithm

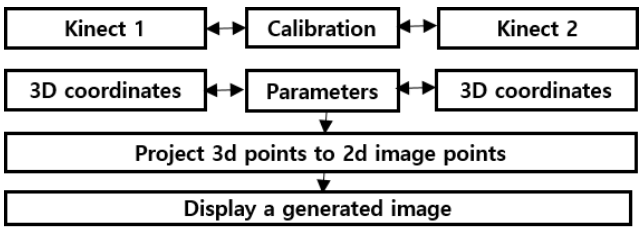

Fig. 2. 시스템 순서도

각각의 Kinect 들은 Calibration 을 통해 내부 파라미터를 구한다(K_{in}), 그 후, Epipolar Geometry 를 통하여, 카메라간의 기하학적 관계를 구한다. 각각의 카메라에서 나온 영상들에서 특징점들을 찾고, 그것들을 이용하여 Fundamental Matrix 를 구한다. 그 후, 각각의 카메라 내부 파라미터를 통하여 Essential Matrix 를 구한다.

$$E = K_{in\,1} F K_{in\,2} \tag{1}$$

Essential matrix 에서 Rotation matrix 를 추출할 때의 잘 알려진 방법은 SVD decomposition 이다. $E = UDV^T$ 를 이용하여 아래의 식들을 만들 수 있다 [3-6].

$$\mathrm{R}_1 = UWV^T,\ \mathrm{R}_2 = UW^TV^T \text{ with } \mathrm{W} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

가상시점의 영상을 만들기 위해서는 우선, 두 카메라에서 얻은 Depth 영상의 정보를 이용하여 좌표계 변환이 필요하다. Pixel 좌표를 World 좌표로 변환시키고, Calibration 을 통해 얻은 카메라 내부 파라미터와 두 카메라간의 각도를 통하여 가상시점으로 point 들을 옮긴다.

Translation Model 은 카메라가 가상의 원 위에 위치해 있고, 카메라들 사이의 가상의 원 위에서 가상시점을 선택한다는 가정하에 아래와 같은 모델로 설정하였다.

$$\mathrm{R} = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix},\quad \mathrm{T} = \begin{bmatrix} -d\sin(\theta) \\ 0 \\ d\,(1-\cos(\theta)) \end{bmatrix} \tag{3}$$

$$M_{ex} = \begin{bmatrix} cos(\theta) & 0 & sin(\theta) & -dsin(\theta) \\ 0 & 1 & 0 & 0 \\ -sin(\theta) & 0 & cos(\theta) & d(1-cos(\theta)) \end{bmatrix} \tag{4}$$

$$\begin{bmatrix} \mathrm{xh} \\ yh \\ w \end{bmatrix} = K_{in} \begin{bmatrix} wx \\ wy \\ wz \end{bmatrix},\quad \begin{bmatrix} wx \\ wy \\ wz \end{bmatrix} = K_{in}{}^{-1} \begin{bmatrix} \mathrm{xh} \\ yh \\ w \end{bmatrix}$$
$$,\ where\ \ wx, wy, wz = world\ coordinates \tag{5}$$

$$x_{im} = xh/w\,,\qquad y_{im} = yh/w\,,\qquad \mathrm{w} = \mathrm{wz}$$
$$,where\ \ x_{im}, y_{im} = pixel\ coordinates \tag{6}$$

식 (5)-(6)을 이용하여, Pixel 좌표를 World 좌표로 변환시키고, 식 (4)를 이용하여, 새로운 가상의 지점의 좌표로 변환을 시킨다. 그 후, 다시 World 좌표를 Pixel 좌표로 변환시킨다.

두 대 이상의 Kinect 를 사용하게 되면, 각각에서 나오는 IR 패턴이 간섭을 일으켜 홀을 발생하게 한다. 이는 불확실한 깊이 값을 제공하여, 좌표 변환과 3D Reconstruction 을 함에 있어 큰 문제를 발생시킨다. 그래서 이 홀 부분을 채우기 위해 Nearest Neighbor(이하 NN) 방식과 Inpainting 기법을 사용하였다 [7].

NN 의 알고리즘은 다음과 같다. 지정한 Mask 안에서 중심을 기준으로 주변에 홀이 T_1 보다 작다면, Mask 내부의 홀이 아닌 값들의 평균값을 그 Mask 중심에 대입한다. 주변의 홀이 T_2 보다 크고, T_3 보다 작다면, Mask 내부의 홀이 아닌 값들의 최대값을 대입한다. 이 과정을 일정 횟수만큼 반복하여 간섭현상으로 생긴 홀을 제거한다.

4. 결과

NN 만 사용하였을 때보다, Inpainting 기법을 사용하면 경계잡음 구역, Occlusion, 그리고 간섭현상으로 생긴 구역들을 다 채울 수 있었다. 하지만 Inpainting 은 객체와 배경영역을 잘

구분하지 못하여, 채워진 영역이 부자연스럽게 보이는 단점이 있다. 따라서, Kinect 에서 영상을 처음 받아올 때만 Inpainting 을 적용하였다.

그림 [Fig. 2.]은 간섭현상이 생겨서 발생한 홀을 NN 과 Inpainting 기법을 통하여 채운 영상이다. 기존에 영상 가운데에 많이 생겼던 간섭현상 홀들이 제거된 영상을 얻을 수 있었다.

Mask 내부의 홀의 개수에 따라 동작하는 NN 기법은 Edge 에서 배경과 전경의 구분에 대한 알고리즘이 적용되지 않았다. 따라서, NN 기법이 적용되는 횟수를 많이 설정하면 할수록, 전경과 배경의 경계가 무너지게 되고, 만족할만한 결과를 얻기가 어려워진다.

Kinect 에서 받은 영상을 위의 그림과 같이 두 기법을 적용시켜 홀을 채운 후, 가상시점에서의 영상 정합 시, 두 영상의 색상이 빛에 의해 다르게 나오는 문제와 두 영상 내의 물체의 위치가 정확히 겹치지 않는 문제가 존재한다. 이는 Kinect 로부터 받은 깊이 값이 매우 정확한 값이 아니고, 홀을 채우는 과정에서 홀 주변과 비슷한 깊이 값이 들어갔기 때문이다. 또한, Kinect 자체의 깊이 영상 Error 도 존재한다. 거리에 따른 깊이 측정 오차율이 기하급수적으로 증가하기에, 이 영향도 존재한다 [8].

Fig. 2. 간섭현상이 생긴 영상에 Inpainting 기법과 NN 기법 적용 결과

5. 결론

제안 시스템을 통해 2 대의 Kinect 를 사용하여 임의의 시점의 영상을 합성하는 시스템을 개발하였다. 하지만, 거리에 따른 깊이 값 측정 오차와 간섭현상으로 생긴 홀을 정확하지 않은 주변의 비슷한 값으로 채웠기 때문에, 정합 오차가 존재하였다.

추후에는, 깊이 영상 내에서 홀을 효과적으로 채워 보다 더 정확하게 두 영상을 정합하는 연구와 CUDA 등의 병렬프로그래밍을 사용하여 속도 향상에 대한 연구를 진행할 예정이다.

감사의 글

이 논문은 2013 년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임 (NRF-2013R1A1A1057779).

참조문헌

[1] 윤국진, 엄기문, 김진웅, 이광순, 허남호, *"3DTV 기술 동향"* TTA Journal, No. 122, pp. 92-97, March - April. 2009.

[2] A. Kubota, A. Smolic, M. Magnor, M. Tanimoto, T. Chen, and C. Zhang, *"Multiview Imaging and 3DTV"*, IEEE Signal Processing Magazine, pp. 10-21, Nov. 2007

[3] G. Slabaugh, "Computing Euler angles from a rotation matrix", Technical Report, http://www.gregslabaugh.name/publications, 1999.

[4] Li Ling, Eva Cheng, I. S. Burnett, *"Eight solutions of the essential matrix for continuous camera motion tracking in video augmented reality"*, ICME, 2001 IEEE International Conference on, July. 2011.

[5] R. I. Hartley, *"Estimation of relative camera positions for uncalibrated cameras"*, Computer Vision-ECCV'92, Lecture Notes in Computer Science Volume 588, pp. 579-587, May. 1992

[6] W. Wang, H. T. Tsui, *"A SVD decomposition of essential matrix with eight solutions for the relative positions of two perspective cameras"*, Proceedings. 15th International Conference on Pattern Recognition, Vol. 1, pp. 362-365, Sept. 2000.

[7] A. Telea, *"An image inpainting technique based on the fast marching method"*, Journal of Graphics Tools, Vol. 9, No. 1, pp. 25-36, 2004.

[8] C. Limin, C. Mingyu, X. Shizhe, L. Yin, *"Analysis of interference between multiple Kinect sensors and a noise reduction method for mobile robots"*, JCIT, Vol. 7, No. 21, Nov. 2012.