3차원 깊이 정보를 이용한 터치 스크린 구현

김호승, 장원석, 권순각
동의대학교 컴퓨터소프트웨어공학과

Implementation of Touch Screen using 3D Depth Information

Ho-Seong Kim, Won-Serk Jang, Soon-Kak Kwon
Dept of Computer Software Engineering, Dong-eui University

요 약

프레젠테이션을 위한 기존 방비들은 기능이 제한적이거나 또는 고비용이라는 문제점을 가지고 있다. 본 논문에서는 깊이 카메라를 이용한 손의 터치와 터치위치와 프레젠테이션 내용을 조합하여 프레젠테이션의 자유도를 높이는 방법을 제안한다. 제안 방법은 깊이 카메라로부터 스크린 영역의 깊이 값을 배경으로 설정하고, 배경과 차이나는 객체가 들어오면, 객체의 깊이 값과 배경의 깊이 값을 비교하면서 포인터를 추출한다. 모의실험 결과로부터 스크린의 상하좌우 중에 한곳에 카메라를 설치하고 프레젠테이션 페이지마다 서로 다른 위치에서 터치가 인식되고, 다양한 프레젠테이션 이벤트가 실행됨을 확인하였다.

1. 서론

프레젠테이션 소프트웨어는 발표 내용의 배치와 디자인 위주의 프로그램이 주류를 이루고 있다. 하드웨어로는 마우스, 키보드와 레이저 포인터 같은 포인트 도구들이 사용된다. 이러한 기기들 대부분은 보조인원이 필요하거나 이벤트가 2~3개뿐이라는 단점이 있다.

최근 몇 년간 깊이 감지 기술의 발달로 실시간 깊이 카메라를 이용한 포즈 추정 및 제스처 인식을 이용한 컴퓨터제어에 관한 연구가 많이 이루어지고 있다.[1,2,3] 특히, 2010년 말에 널리 보급되어 깊이 카메라를 이용한 프레젠테이션 제어 소프트웨어의 연구가 많이 이루어지고 있다.

깊이 카메라의 방식은 두 종류로 TOF(Time of Flight) 방식과 스테레오 정합방식이 있다. 이 방식들 중에서 조사파와 반사파의 위상차이를 이용하는 스테리오 정합 방식이 가장 많이 사용된다. Kinect와 Xtion계열은 빛을 조사하는 발신부와 반사파를 감지하는 수신부가 일정 간격 분리되어 있다. 따라서 특정한 패턴을 갖는 적외선 구조광을 조사하고, 물체에 반사되어 돌아오는 파를 조사한 파와 스테레오 정합하여 깊이 정보를 추출한다. 따라서 어느 파를 기준으로 하건, 물체의 좌측 또는 우측 면에 오차가 발생할 수밖에 없다[4].

본 논문에서는 스테레오 정합방식의 깊이카메라를 이용한 포인터 추출과 포인터의 깊이 정보와 프레젠테이션 내용에 따른 공간 좌표를 이용한 이벤트 제공 방법을 제안한다. 깊이 카메라에서 제공되는 거리 정보로부터 밝기에 영향을 받지 않는 배경영상을 만들 수가 있다. 이러한 배경 영상으로부터, 움직이는 관심영역을 추출하고, 추출한 영역을 라벨링한다. 라벨링한 영역의 크기를 비교하여 특정 크기의 영역을 추출한다. 이벤트는 깊이 정보와 공간 좌표를 이용하여 이벤트의 범위와 실행할 이벤트를 설정한다.

2. 제안하는 터치 스크린 방법

스크린의 상하좌우 중의 한곳에 깊이카메라를 설치하고, 스크린 터치를 통하여 다양한 프레젠테이션 이벤트를 제공하는 방법을 제안한다. 그림 1과 같이 깊이 정보를 이용하여 움직이는 관심영역만을 추출함으로써 일반적인 색상 카메라에서 나타나는 밝기 차에 대한 오차를 제거할 수 있다. 스크린을 터치하는 포인터의 깊이 정보와 공간 좌표를 이용하여 이벤트 인식을 확인할 수 있다. 이벤트는 설정해 놓은 각 이벤트 영역에 포인터가 들어왔을 때 실행이 이루어진다.

제안된 방법은 깊이카메라를 이용하여, 깊이 카메라로부터 촬영 범위에 대한 모든 깊이 정보를 얻는다. 잡음은 모든 깊이 정보에 연속되지 않고 불연속적으로 특정 시간에 존재한다. 따라서 잡음 요인을 제거하기 위하여 일정시간 동안 깊이 정보를 축적하고 깊이 정보의 특정 범위를 찾기 위한 배경 영상을 구한다. 배경영상 생성시, 깊이 정보의 범위만큼 저장 공간을 만든다. 일정 개수의 화면이 입력될 때까지 깊이 정보의 평균을 구한다.

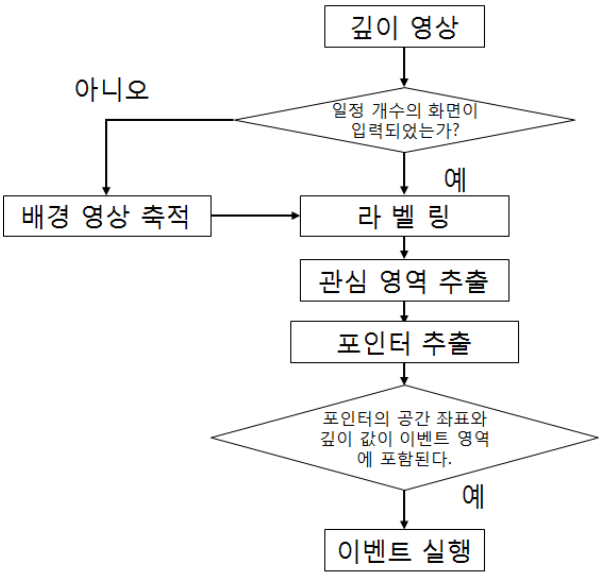

그림 1. 제안된 프레젠테이션 제어 방법

깊이 카메라의 촬영범위에 움직이는 객체가 나타나거나 배경영상을 생성하는 동안 나타나지 않았던 잡음들은 임계값의 범위에 포함되지 않는다. 배경영상에서 포함되지 않는 영역의 화소 값을 '255'로, 포함되는 영역은 '0'으로 바꾼다.

그림 2는 관심영역을 추출한 영상이다. 배경 영상의 첫 화소부터 순차적으로 화소 값이 '255'인 화소들을 그룹화하고 라벨링하는 과정을 수행한다. 라벨링된 객체들 중 잡음과 불특정 객체들을 제거하기 위해 화소 개수가 최소 100개 이상의 객체만을 추출하여 관심 영역을 추출한다.

(a) 색상 영상

(b) 배경 영상

그림 2. 배경영상에서 관심 영역의 추출

추출한 객체에서 포인터를 추출하는데, 포인터는 스크린에 가장 가까운 위치를 얻어 포인터로서 사용한다. 스크린과의 가까운 위치를 찾는 것은 카메라의 설치 방향에 따라 다르다. 카메라가 스크린의 왼쪽에 설치 된 경우는 스크린 깊이 영상의 방향 기준으로 객체 내에서 가장 왼쪽에 있는 화소의 위치를 포인터로 사용하며, 카메라가 오른쪽일 때는 가장 오른 쪽에 있는 화소로, 위쪽일 때는 가장 위쪽에 있는 화소로, 아래쪽일 때는 가장 아래쪽에 있는 화소를 포인터로 사용한다.

포인터가 스크린에 접촉하면 터치하는 것으로 판단한다. 스크린에 접촉한 경우를 터치한 것으로 고려한다면 포인터 위치의 깊이 값과 스크린 배경 깊이 영상에서 해당되는 위치에서의 깊이 값을 비교하여 두 값의 차가 특정 값 미만일 때 터치한 것으로 판단한다.

그림 3은 스크린을 기준으로 포인터의 3차원 좌표를 검출하는 방법이다. 포인터의 깊이 값과 공간 좌표를 비교하여 이벤트 범위에 포함될 시 이벤트를 실행한다. 이벤트의 판단을 위한 범위지정과 설정은 프레젠테이션의 내용을 확인하고 자신이 터치할 부분을 먼저 고려한다. 그리고, 촬영되는 범위에 대해 깊이 정보와 배경영상의 공간 좌표를 이용하여 원하는 곳의 수평 및 수직축을 구분 및 지정하고, 실행할 이벤트를 설정한다. 포인터의 깊이 값과 화소 위치가 이벤트의 깊이 값 영역과 공간 좌표 영역에 대응될 때 이벤트를 실행한다.

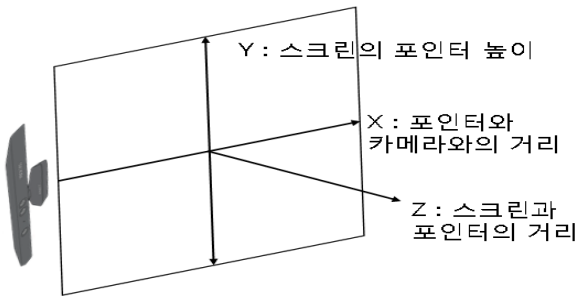

그림 3. 포인터의 좌표 구분

3. 모의실험 및 결과

본 논문에서는 깊이 카메라로 Microsoft사에서 개발한 Kinect[5]를 사용한다. Microsoft사는 Prime-Senst사의 움직임 센서 도임, GestureTek사의 기업 인수 등을 통하여 게임분야에 적용하기 위한 Kinect를 개발하였다. Kinect 기술은 레이저를 이용해 사용자에 적외선을 투사하고 나서 적외선 점(dot)들의 반사파 강도를 측정함으로써 이미지들을 나눈다[6].

깊이 정보는 PrimeSense사의 오픈 라이브러리인 OpenNI[7]을 사용하여 얻어 냈다. 촬영되는 범위에 대해 깊이 정보를 얻어올 수 있으며 초당 30화면에 대한 깊이 정보를 얻어올 수 있다.

포인터로는 손가락을 이용하였고, 카메라는 스크린의 좌측에 설치하였다. 이벤트의 테스트에는 Microsoft사의 Power Point를 사용하였다.

그림 4. 실제 카메라의 설치 위치

초기의 배경영상은 30장의 화면에 대하여 평균 깊이 값의 하위 또는 상위 임계값을 생성하였다. 상위 및 하위 임계값을 설정하기 위한 특정치는 100을 사용하였다. 축적된 배경영상의 임계값의 영역에 포함되지 않는 깊이 값에 대하여 그룹화하고 라벨링을 수행한다. 라벨링된 객체들 중 잡음과 불특정 객체들을 제거하기 위해 화소 개수가 최소 100개 이상의 객체 만을 추출하여 관심영역을 추출한다. 추출한 객체의 화소들 중 가장 왼쪽에 있는 화소를 포인터로 사용한다. 매 화면마다 포인터의 깊이 값과 공간 좌표를 확인하여 스크린의 수평축을 구분하고, 공간 좌표의 위치를 구분하여 이벤트를 판단 및 실행한다. 그리고, 현재 이벤트 실행 후 다음 이벤트 사이의 지연 시간(30장)을 주어 의도치 않은 이벤트의 연속실행을 방지하게 한다. 그림 4에서와 같이 모의실험에 사용된 이벤트는 페이지 좌우 이동, 확대, 축소, 종료, Enter, 이전 슬라이드, 다시 시작, 마지막 슬라이드의 9개를 사용하였다. 프레젠테이션 페이지마다 터치하는 위치와 이벤트를 달리하도록 설정하였다. 키보드 이벤트는 작동이 원활하지만 마우스 포인터를 이용한 이벤트는 실행 전 입력할 좌표를 미리 입력해야 한다. 좌우 이동, 종료, Enter 등 버튼 이벤트는 인식 즉시 반응한다. 하지만 확대, 축소 이벤트 같이 마우스의 기능을 사용하거나 시간이 필요한 이벤트는 마우스의 좌표 및 사용자의 설정이 어려워지기 때문에 다양하지 못하였다.

프레젠테이션시에 이벤트의 실행 순간은 포인터가 스크린에 직접 접촉하든지 등으로 사용자의 편리성에 맞게 구현할 수 있다. 접촉 하지 않고 사용할 때는 사용자의 포인터와 스크린 사이의 거리 간격을 깊이 영상 내에서 관심영역의 포인터 위치와 스크린 사이의 거리는 깊이 값에서 일정 화소만큼의 여유 간격을 두게 함으로써 가능하며, 표 1에서는 스크린 수직축 방향으로 2~10화소에 해당하는 좌표로 포인터 위치를 2화소씩 바꿔가며 실제 포인터와 스크린 사이의 거리를 조사하였다.

표 1. 이벤트 인식을 위한 포인터와 스크린 사이의 거리(단위 Cm)

카메라로부터 이벤트 위치	스크린과 포인터 사이의 여유 간격				
	2화소	4화소	6화소	8화소	10화소
55	2	2.3	3.2	3.5	4.3
135	0	0	0	2	2
235	1	1.5	2	4	5

이벤트 범위가 카메라와 가까운 위치(55Cm), 스크린의 중앙 위치(135Cm), 카메라와 먼 위치(235Cm) 등 세 가지 범위에 대해 이벤트가 실행될 때 포인터와 스크린 사이의 간격을 측정하였다. 여유 화소가 클수록 포인터와 스크린 사이의 거리가 먼 거리에서 실행이 가능했다. 그리고 스크린과 가까울수록 측정 거리의 차가 큰 것을 확인할 수 있었다. 카메라와의 거리가 가까울수록 차가 크게 나타나는 원인은 카메라 촬영 시에 나타나는 관심영역의 원근감인 것을 확인할 수 있었다. 스크린의 수직구분은 포인터의 깊이영상에서 화소단위로 구분하기 때문에 오차 없이 구분하는 것을 확인하였다. 위의 실험에서 추출한 화소간의 거리를 이용하여 발표자의 발표 위치, 스크린과 발표자의 거리 등을 고려하여 설정할 수 있을 것으로 기대된다.

제안된 방법의 인식률은 카메라에 촬영되는 영역에 영향을 받을 것이다. 스크린외의 영역의 복잡도가 단순할수록, 불특정 객체의 수가 작을수록 인식률은 크게 높아진다. 또한, 설치 위치가 변경될 때마다 영상에서 스크린의 영역만을 인식하고 터치 판단시 원근감에 따른 오차를 개선할 수 있으면 인식률과 효율성이 증대될 수 있을 것이다.

4. 결론

본 논문은 깊이카메라를 이용하여 포인터를 추적하고, 이벤트 범위의 설정과 실행방법을 제안하였다. 객체의 추적은 깊이 정보를 이용한 다양한 작업으로 추출한다. 추출한 최종 포인트 지점이 이벤트 영역에 대응되는 깊이 정보와 화소 좌표에 위치하지 않으면 이벤트가 수행되지 않고, 다시 깊이 정보획득으로 되돌아가게 한다. 이러한 제안된 방법은 보조 인원 없이 포인터 인식으로 설정한 범위에 대해서 이벤트의 실행을 확인하였다. 현재는 깊이 카메라의 높은 가격으로 인해서 다양한 방식의 카메라를 이용한 연구에 제한이 있다. 또한, 현재 깊이 정보를 다양한 분야에 적용된 사례가 미미하다. 본 논문에서 제시된 깊이 카메라를 이용한 프레젠테이션 방법은 현재 깊이 카메라의 가격이 저렴해지고 정확도가 올라간다면 마우스, 키보드, 레이저 포인터 같은 제어 장비들이 깊이 카메라 하나로 통합될 것으로 기대된다. 소프트웨어 측면에서는 Power Point 뿐만 아니라, 다양한 프레젠테이션 소프트웨어에 적용이 가능할 것이고, 프레젠테이션 제어 외에 TV, 홈 원격제어 등 HCI분야의 다양한 분야에 응용이 가능할 것으로 기대된다. 현재 본 논문의 내용은 특허출원중이며, 프레젠테이션 분야로 사업화 가능할 것으로 기대된다.

참고문헌

[1] M. Siddiqui and G. Medioni, “Human pose estimation from a single view point, real-time range sensor”, In Workshop on Computer Vision for Computer Games at Conference on Computer Vision and Pattern Recognition, 2010.

[2] R. Munoz-Salinas, R. Medina-Carnicer, F.J. Madrid-Cuevas, and A. Carmona-Poyato, “Depth silhouettes for gesture recognition”, Pattern Recognition Letters, vol.29, no.3, pp.319-329, 2008.

[3] P. Suryanarayan, A. Subramanian, and D. Mandalapu, “Dynamic hand pose recognition using depth data”, In International Conference on Pattern Recognition, 2010.

[4] Ja-Myung Koo, Young-Ho Seo, Hyun-Jun Choi, Ji-Sang Yoo, and Dong-Wook Kim, “Intermediate Depth Image Generation using Disparity Increment of Stereo Depth Images”, JBE , Vol. 17, No. 2, pp.363-373, 2012.

[5] http://www.microsoft.com/en-us/default.aspx

[6] Soon-kak Kwon, Seong-Woo Kim, “Motion Estimation Method by Using Depth Camera”, JBE Vol. 17, No. 4, pp.676-683, July 2012.

[7] http://openni.org