

## 프로젝터 화면상에서 레이저 포인터를 이용한 마우스 기능 구현에 관한 연구

김주국, 김상준, 이기원, 허 현, 이양희, 장홍순  
한국산업기술대학교

### Development of the Virtual Mouse on a Projector Screen using a Laser Pointer

Ju-Kuk Kim, Sang-Jun Kim, Ki-Won Yee, Heon Huh, Yang-Hee Yee, Hong-Soon Chang  
Korea Polytechnic University

**Abstract** - 반도체 기술의 비약적 발전에 힘입어 현재의 개인용 컴퓨터는 고성능 CPU를 탑재하고 1990대의 텍스트 기반의 운영체제에서 벗어나 그래픽 기반의 운영체제에서 다양한 멀티미디어 기능을 제공 한다. 이를 위한 입력장치로 텍스트 기반 운영체제에서 주로 사용된 키보드뿐 아니라 마우스, 카메라, 터치스크린 등의 다양한 장치들이 사용되고 있다. 그러나 빔 프로젝터를 이용한 프레젠테이션의 경우 아직도 레이저 포인터를 이용한 발표가 일반적이며 발표자와 빔 프로젝트를 PC와의 인터랙션이 없기 때문에 다양한 멀티미디어 기능 구현이 제한적이다. 본 논문에서는 USB 웹 카메라를 이용하여 프로젝터 화면을 촬영한 후 영상처리 라이브러리인 OpenCV를 기반으로 레이저 포인터의 위치와 동작을 검출하여 원거리에서도 사용자가 레이저 포인터를 이용하여 마우스 동작을 제할할 수 있는 시스템을 개발하고자 한다. 이를 활용하면 레이저 포인터를 사용하여 발표자가 별도의 입력장치 없이 PC와의 인터랙션이 가능해져서 다양한 멀티미디어 기반의 프레젠테이션이 가능해진다.

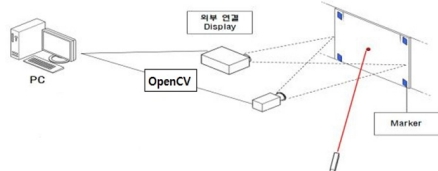
#### 1. 서 론

지난 20년 동안 반도체 기술의 발전과 더불어 컴퓨터 성능 및 기능도 급속도로 발전되었다. 초기에는 컴퓨터와의 인터페이스 장치로 유선 키보드와 유선 마우스와 같이 근거리에서 작업할 수 있는 장치가 주로 사용되었다. 근래에는 무선 통신 기술에 기반을 둔 다양한 입출력 장치들이 개발되어 사용자가 컴퓨터로부터 일정한 영역 밖에서도 컴퓨터와의 인터랙션이 가능하게 되었다. 그러나 빔 프로젝터를 이용한 프레젠테이션의 경우 아직도 레이저 포인터의 사용이 일반적인데 레이저 포인터를 통해 발표자가 정보를 컴퓨터에 입력하는 것은 불가능하다. 이와 같은 문제점을 해결하기 위해서 발표자의 입장에서 별도의 추가 입력장치 없이 레이저 포인터만을 이용하여 컴퓨터와 인터랙션을 가능케 하는 시스템을 개발을 목표로 하였다. 이를 활용하면 지금까지 프레젠테이션 사용자 제한적이었던 인터랙션 환경을 극복하고 다양한 분야에서도 응용될 수 있을 것으로 보인다.

#### 2. 본 론

##### 2.1 시스템 구성

본 가상 마우스 시스템은 빔 프로젝터 기반의 강의 시스템을 전제로 하고 있다. [그림 1]과 같이 호스트 PC의 외부에 빔 프로젝터와 USB 2.0 웹 카메라가 연결되어 있고, 빔 프로젝터를 통해 스크린에 투영된 컴퓨터 화면을 획득할 수 있는 거리에 카메라를 위치시킨다.



〈그림 1〉 시스템 개념도

사용자의 컴퓨터 화면이 빔 프로젝터에 의해 스크린에 투영되면, 사용자는 레이저 포인터를 발사하여 스크린에 레이저 도트를 형성한다. 이때 컴퓨터에 연결되어 있는 웹 카메라가 스크린 영상을 획득하여 OpenCV를 통해 해석한다. 영상해석을 통해 레이저 도트의 위치 및 움직임을 해석하여 이를 컴퓨터의 마우스 기능으로 연결시킨다. 따라서 발표자는 추가 장치 없이 레이저 포인터만을 이용하여 다양한 마우스 기능을 실현하고 응용 프로그램과의 인터랙션이 가능하게 된다.

##### 2.1.1 하드웨어 구성

하드웨어는 컴퓨터, USB 2.0 웹 카메라, 레이저 포인터, 빔 프로젝터와 스크린으로 구성되어 있다. 레이저 포인터는 강의나 프레젠테이션에 많이 쓰이는 적색 레이저를 사용하였고, USB 2.0 웹 카메라는 640×480 해상도의 CMOS 이미지센서가 장착되어 30 frame/sec의 속도로 스크린 영상 획득이 가능한 모델로 선정하였다. 빔 프로젝터는 1024×768의 해상도를 투영해주는 모델이고 스크린은 강의나 프레젠테이션에 많이 쓰이는 곡면 스크린을 사용하였다.

〈표 1〉 웹 카메라와 레이저 포인터 제원

삼성전자 SPC-A30M	레이저 포인터
<ul style="list-style-type: none"> <li>• USB 2.0</li> <li>• CMOS 이미지센서</li> <li>• 640 x 480 (Max)</li> <li>• 30만 픽셀 화소수</li> <li>• 30 frame/sec(Max)</li> </ul>	<ul style="list-style-type: none"> <li>• 반도체 레이저</li> <li>• 적색 : 650nm</li> <li>• 1mW 레이저 출력</li> </ul>

##### 2.1.2 소프트웨어 구성

소프트웨어 구성은 스크린에 투영된 실제 화면 인식 모듈, 레이저 포인터의 정확한 위치 인식 모듈, 그리고 마우스 왼쪽 버튼 클릭/더블클릭 등의 마우스 기능 구현 모듈 3가지로 크게 구분할 수 있다. 소프트웨어는 스크린에 투영된 영상을 카메라가 1/30초 단위로 획득해 온 이미지를 입력으로 동작한다.

〈표 2〉 소프트웨어 사양

PC 사양	영상처리 프로그램	마우스 프로그램
<ul style="list-style-type: none"> <li>• 윈도우 xp/ 7(64bit)</li> <li>• CPU : 1.3GHz</li> <li>• RAM : 4GB</li> <li>• 해상도 : 1366×768</li> </ul>	<ul style="list-style-type: none"> <li>• OpenCV 2.1 라이브러리</li> </ul>	<ul style="list-style-type: none"> <li>• Visual Studio 2008 버전</li> <li>• MFC 응용 프로그램</li> </ul>

#### 2.2 스크린 인식 및 왜곡 보정

##### 2.2.1 스크린 인식

일반적으로 웹 카메라로 찍은 영상 이미지는 스크린 화면에 투영된 컴퓨터 화면뿐만 아니라 주변 환경까지도 포함한다. 그러나 정확한 영상해석을 위해서는 최초 영상 이미지로부터 컴퓨터 화면에 해당하는 영역만을 정확히 분리해 내야한다. 본 시스템의 구현에 있어서 이와 같은 스크린 인식 문제를 최우선적으로 해결해야 한다. 스크린 인식을 통해서 최초의 영상 이미지로부터 컴퓨터 화면에 해당하는 영역만 추출된 이미지를 기반으로 레이저 포인터 도트의 위치를 추정하게 된다.

본 시스템에서는 빔 프로젝터가 투영시키는 컴퓨터 영상만을 획득하기 위해 화면 인식 프로그램 내에서 영상 확대와 마우스 이벤트함수를 활용하여 사용자가 직접 마우스를 이용해 영상의 특정 영역을 획득할 수 있도록 구현했다. 구체적으로 최초영상은 스크린과 컴퓨터 화면 그리고 주변 환경이 입력됐을 때 사용자는 마우스를 이용하여 영상내의 컴퓨터 화면 내 개의 모서리 클릭한다. 이때 추출된 좌표들은 컴퓨터 화면 이외의 주변 환경을 제거한 영상만을 1:1 매칭 시키기 위한 내부함수로 보내진다 [1]. 좌표는 내부함수에서 계산되어 컴퓨터 화면 외의 주변 환경화면은 제거되고 위치 인식 모듈로의 입력영상은 컴퓨터 화면만이 존재하게 된다.

##### 2.2.1 왜곡 보정

본 시스템에서는 [그림 1]과 같이 웹 카메라와 빔 프로젝터는 동일선상에 있다. 이때 웹캠의 렌즈와 스크린과의 각도, 거리로 인해 카메라가

획득해 오는 영상이 왜곡되어 화면의 실제 모습과는 다르게 영상이 곡선의 형태로 보임을 관찰 할 수 있다. 이것은 영상의 가장자리 부분으로 갈수록 심해진다. 이러한 문제점을 화면 인식 모듈 프로그램 내부함수와 체스보드를 이용한 보정을 통해 해결하였다.

웹캠의 입력 영상에 0.5초의 짧은 시간 동안 체스보드를 노출시킨다. 프로그램 내부에서 체스보드를 구성하고 있는 각각의 코너들은 위치에 따라 인덱스를 부여받아, 영상의 왜곡된 정도를 나타내는 왜곡지도를 구성한다 [2]. 이 왜곡 지도와 카메라 렌즈의 왜곡 계수 값을 이용하는 라이브러리 내부함수를 통해 왜곡이 보정된 영상을 생성 후 출력하게 된다.



〈그림 2〉 스크린 왜곡 현상

### 2.3 레이저 포인터 도트의 인식과 위치 추정

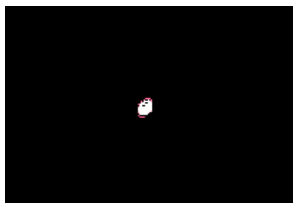
본 시스템에서는 적색 반도체 레이저를 사용하였다. 레이저가 갖는 특성으로는 한 가지의 색상만 가지는 단일성과 한 가지의 파장만을 가지는 단일 파장, 빛이 퍼지지 않고 직진하는 지향성 등이 있다. 이런 특성으로 인해 레이저 포인터 도트에 상당히 강한 에너지를 집중시킬 수 있다. 본 시스템에서는 레이저의 특성 중 단일성에 착안한 RGB 컬러모델을 이용하여 레이저 포인터를 인식해 내었다

#### 2.3.1 컬러모델을 이용한 인식

본 시스템은 레이저 포인터 도트의 RGB 값을 이용해 이미지 상에서 도트의 위치를 추출한다. 최적화된 환경 (빛이 적은 혹은 단일색상의 배경)의 영상에서 포인터 도트의 색상을 찾아서 좌표 값을 획득하려고 하였지만 도트와 배경 색상이 같은 경우 포인터 도트를 구분하지 못하는 문제가 발생하였다. 또한 주변 환경이 밝을 때 혹은 빔 프로젝터의 밝기가 매우 밝을 때는 많은 노이즈가 발생하였다. 이러한 문제를 해결하기 위해서 RGB 컬러모델을 이용하여 포인터 도트를 추출하였다.

1×1 픽셀은 RGB 컬러 공간에서 서로 가산될 수 있는 R(빨강), G(초록), B(파랑)값으로 구성된다. 이것에 착안하여 레이저 포인터와 스크린에 투영된 컴퓨터 화면, 노이즈 및 기타 물체들을 분리시키기 위해 레이저 포인터 도트의 RGB값을 데이터로 환산해 추출한 결과 다른 배경에 비해서 상대적으로 RGB 컴포넌트 값이 모두 높게 측정되었다. 이 데이터를 바탕으로 프로그램 내에서 레이저포인터가 갖는 특정 RGB 값만을 임계값으로 설정한다.

설정된 임계값을 이용하여 프로그램은 획득해온 영상의 픽셀 전체를 스캔한다(이진화 과정[3][4]). 이때 임계값에 충족되는 픽셀 데이터 값만이 흰색으로 남게 되어 [그림 3]과 같은 영상을 획득한다.



〈그림 3〉 RGB값을 이용하여 획득한 laser pointer 도트

#### 2.3.2 좌표 추출

레이저 포인터를 이용한 마우스 기능구현을 위해서 영상 속에 위치하고 있는 포인터 도트의 좌표 값을 추출해야 한다.

실험을 통해 측정된 임계값을 이용하여 각 픽셀의 RGB값을 스캔하며 영상 이진화 단계를 거치게 된다 [3], [4]. 스캔을 하는 동안 임계값을 만족시키는 모든 픽셀의 좌표가 저장된다. 이렇게 저장된 좌표 값과 그 개수를 이용하여 레이저 포인터의 좌표평균을 구하여 좌표를 추출한다.

좌표 추출 시 외부 환경(빛, 프로젝터)에 의한 노이즈로 레이저 포인터의 밝기가 약해져 임계값에 미치지 못해 좌표를 계속적으로 추출하지 못하는 문제가 발생한다. 이에 따라 좌표의 불안정한 추출이 유발 될 수 있다. 따라서 본 시스템은 이러한 문제를 해결하기 위해 현재 프레임에서 좌표가 추출되고 그 다음 프레임에서 좌표가 추출 되지 않았다면, 그 전 프레임의 좌표를 계속 사용하여 불안정한 좌표의 움직임을 막을 수 있다. 이 프레임간

의 보간 기법(interpolation)은 30f/s의 영상 이미지에 대해서 진행하기 때문에 시간적으로는 자연스러운 마우스 움직임에 큰 영향을 주지 않고 안정적이며 정확한 인터랙션을 구현 할 수 있다.

### 2.4 마우스 기능구현 프로그램

본 시스템에서 가장 기본이 되는 기능은 마우스 이동과 클릭이벤트이다. 추출된 좌표로 포인터가 움직이게 되고 마우스가 일정 좌표에 일정 시간(사용자 지정)동안 머물러 있다면 클릭 이벤트가 발생하게 된다. 이 기본 기능을 바탕으로 일반적인 마우스 기능인 더블클릭, 드래그, 그리기 등의 기능은 프로그램으로 구현하였다.

#### 2.4.1 Mouse Move

레이저 포인터를 따라 마우스가 이동하고 이벤트를 발생하기 위해서는 안정적인 좌표 값을 받아와야 한다. 하지만 사용자의 손 떨림으로 좌표가 미세하게 떨리는 현상이 발생하여 마우스의 불필요한 움직임이 발생하였다. 따라서 본 시스템에서는 현재의 좌표와 이전 프레임의 좌표간의 거리 차이가 10 이하인 경우에는 사용자의 손 떨림으로 간주하였고 거리 차이가 10 이상인 경우에는 일반적인 움직임이라고 판단하도록 설계하였다. 이러한 필터링을 통해 손 떨림 현상을 보정하여 보다 안정적인 마우스 사용이 가능하게 하였다.

#### 2.4.2 Mouse Click

레이저 포인터의 On-Off만으로 클릭 기능을 구현하려 하였지만 사용자가 레이저 포인터의 스위치를 누르는 순간 압력이 발생하여 평균 15 픽셀의 좌표차이를 발생하며 동일한 곳을 포인터링 하지 못하고 Mouse Move로 인식이 되는 문제점이 발견되었다. 본 시스템에서는 이런 문제점을 극복하고자 추출된 포인터의 좌표 값이 클릭하고자 하는 좌표 값의 상하 10픽셀 범위 안에 2초간 머무를 경우 클릭이 되도록 하였다.

### 3. 결 론

본 연구에서는 빔 프로젝터 기반의 강의 시스템을 전제로 일반적으로 사용되고 있는 반도체 레이저 포인터와 USB 2.0 웹 카메라를 이용하여 원거리에서도 마우스 인터랙션이 가능한 시스템을 구현하였다. 본 연구에서 만든 시스템은 강의실이나 프레젠테이션 실에 있는 컴퓨터에 최초 설치 및 세팅 하고나면 사용자의 추가적인 설치가 필요하지 않으며, 사용자가 가지고 있던 레이저 포인터만을 이용해 컴퓨터와 인터랙션을 할 수 있다는 장점이 있다.

또한 스크린 상에 투영되는 빔 프로젝터의 영상과 카메라를 통해 획득된 영상간의 왜곡이 발생하였지만 체스보드의 코너인식을 통한 영상 재해석 기술로 왜곡된 영상을 최적의 상태로 보정하였다. 그 결과 스크린 상에서 레이저 포인터의 위치가 마우스 커서의 위치와 정확히 일치하는 결과를 얻을 수 있었다.

하지만, 가시광선을 내보내는 레이저 포인터인 만큼 영상을 획득해 오는 카메라 역시 가시광선 카메라이기 때문에 빔 프로젝터가 설치된 장소의 광량에 따라 레이저 포인터 및 스크린 인식률에 변동이 생긴다. 따라서 빔 프로젝터가 설치된 장소를 비교적 어둡게 해야 하는 불편함은 본 연구가 추가적으로 해결해야 할 문제이다. 또한 사용자가 한명이 아닌 멀티유저, 즉 두 개 이상의 레이저 포인터가 동시에 사용되거나 다양한 색상의 레이저 포인터가 사용되는 경우로 연구를 확장시킬 필요가 있다.

### 감사의 글

본 논문은 지식경제부의 전력기술 인프라 사업(I-2010-0-025)과 산학협력 중심대학 육성 사업(기업 연계형 캡스톤 디자인) 지원을 통한 연구 결과에 의해 작성 되었습니다.

### [참 고 문 헌]

- [1] 황선규, "IT EXPERT 영상 처리 프로그래밍 by Visual C++", 한빛미디어, 2007.
- [2] 정성환, 이문호 공역, "컴퓨터비전 실무 프로그래밍-기본편", 홍릉과학출판사, 2007.
- [3] OpenCV, "http://tech.groups.yahoo.com/group/OpenCV"
- [4] OpenCV코리아, "http://www.opencv.co.kr"