

+ 조원현, 이상준, 김두원, 박승근 · KBS 창원총국 기술국

BROADCAST BULLETIN

■ 모션캡처를 이용한 동적 프레젠테이션

1. 이야기

20XX년 호주오픈 테니스 대회. 사실 박철수가 이번 포인트만 획득하면 한국인 최초로 호주오픈에서 우승하게 된다. 우승 후 금의환향 할 자신의 모습을 떠올리며 마음을 가다듬는다.

'침착하게... 마지막은 멋있게 서브에이스로 끝내자!'

관중, 취재진 모두 긴장한 탓인지 숨죽인 채 철수의 서브만 기다리고 있다. 손을 축축하게 적신 땀을 티셔츠에 닦은 뒤 자신의 라켓으로 경기장의 정적을 힘차게 가른다. 잠시 후, 심판은 라인아웃을 선언했고 상대편 선수가 두 손을 번쩍 들었다. 이게 웬일인가? 자신의 눈에는 분명히 라인에 걸친 서브였다. 두 눈을 가리키며 심판에게 잘못 보았다는 보디랭귀지를 취하고 그래픽 판독을 요청했다.

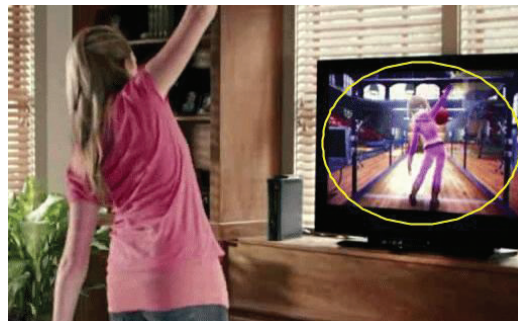
"IN!!!" 해냈다. 드디어 해냈다. 웃기게도 철수의 우승에 결정적 역할을 한 것은 심판이 아니라 카메라 기반의 그래픽 판정 시스템이었다. 철수에게 그것이 무슨 소용이라. 그는 그날 자신의 생애 최고의 순간을 누렸다.

2. 도입

혹시 국제 테니스 경기를 유심히 본 적 있는지? 수년 전부터 메이저 테니스 대회에서는 호크아이(Hawk-eye)라는 전자 판정 시스템을 쓰고 있다. 이 시스템은 경기장에 설치된 6대의 카메라로 테니스공을 촬영하여 위치를 추적한 뒤, 판정 요청이 있을 시 공의 궤적을 그래픽으로 리플레이해 주는 시스템이다.

이와 비슷한 원리로 카메라를 이용, 특정 목표물을 촬영하여 우리가 원하는 용도로 사용하는 시스템이 이미 여러 산업계에서 쓰이고 있는데 축구경기에서 각 선수의 활동량과 활동반경을 분석하는 프로그램, MicroSoft에서 출시한 Xbox Kinetic 게임, 차량번호인식 시스템 등이 그 예이다.

'컴퓨터 비전(Computer Vision) 기술'을 기반으로 삼고 있는 이런 시스템들은 기본적으로 카메라의 영상입력을 실시간으로 분석하며 작동한다. 이어지는 장에서는 비전기술과 이것을 방송기술에 접목할 수 있는 하나의 방법을 간단히 소개한다.



3. 컴퓨터 비전 기술



⇒



[일반적으로 우리가 말하는 사진은 [좌]이지만, 디지털 기기 입장에서 본 사진은 [우]일 것이다

디지털 카메라로 찍은 호랑이 사진이 있다. '사진 속에 호랑이가 있다'는 사실은 우리 인간이 보고 알아낸 정보일 뿐이다. 갑자기 이게 무슨 말이냐고? 카메라로 찍은 사진을 물리적, 기술적 관점에서 보면 단지 1과 0으로 이루어진 데이터 다발일 뿐 그 이상의 의미는 없다. 다르게 말해, 카메라(또는 모든 디지털 기기)는 사진 속에 호랑이가 있는지 없는지 인식하지 못하며 단지 주어진 절차에 의해 렌즈로 들어온 빛을 1과 0의 데이터로 변환해 주는 역할을 한다.

컴퓨터 비전 기술은 여기에서 시작한다. 인간이 사진 속의 정보를 알아내는 것처럼 컴퓨터에게도 '이런 능력'(사진 속 픽셀들의 패턴이나 특징을 소프트웨어로 분석하는 것)을 부여해서 여러 분야에 응용을 해보자는 것이다.

실제로 사진에서 호랑이 픽셀 부분만 어떻게 가려낼까? 단순하게 생각해 보자.

- ① 모든 하늘색 픽셀을 지운다.
- ② 모든 초록색 픽셀을 지운다.

아쉽게도 끝이다. 물론, 풀밭에 노란색 꽃들도 있기 때문에 모든 노란색 픽셀도 지워야하지만 그렇게 하면 호랑이 몸에 있는 노란색 계통의 픽셀도 함께 지워질 것이다. 이때는 좀 더 복잡한 알고리즘을 추가해야 하지만 글의 목적과는 다소 어긋나므로 여기까지만...

비전 기술은 사진뿐만 아니라 실시간 영상에도 적용된다. 사진에서 하던 픽셀 분석을 1초에 프레임 수만큼 수십 번 반복하면 되는 것이다.

4. 시스템 구성 및 설명

시스템의 기본 의도는 진행자의 손 움직임을 카메라로 캡처, 분석한 후 프레젠테이션 화면을 동적으로 진행해 나가는데 있다. 먼저, 정보전달이 주목적인 시사, 보도, 교양 프로그램을 제작하는 환경에서 진행자의 뒤에 있는 디스플레이 장비를 통해 방송 내용과 관련된 그래픽을 디스플레이 하는 상황이라고 가정한다. 현 시스템이라면 그래픽 장면의 전환은 진행자의 진행과는 별개로 CG요원에 의해 스튜디오 밖에서 이루어진다.



[시스템 구성도]

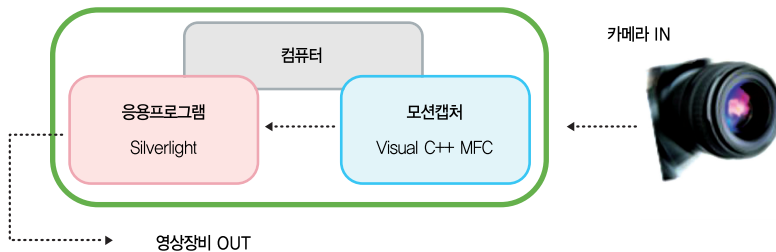
이것을 더욱 발전시켜 진행자의 동작으로 원하는 순간에 그래픽 효과를 직접 제어하면 어떨까? 동작을 결들인 프로그램 진행 방식은 시청자들의 몰입도를 증대시켜 프로그램 제작의도를 보다 쉽게 전달할 수 있다.

시스템 작동은 다음과 같은 순서로 진행된다.

- ① 진행자는 진행과 함께 다양한 동작을 취한다.
- ② 카메라로 진행자의 동작을 실시간으로 찍어 컴퓨터에 전달한다.
- ③ 카메라로부터 전달된 영상을 분석하여 진행자가 사전에 약속한 동작을 취했는지 여부를 판단한다.
 - 약속된 동작이었을 경우: 다양한 효과를 디스플레이 장비에 나타냄.
 - 약속된 동작이 아니었을 경우: 필요 없는 동작으로 판단하고 다음 프레임을 받아들임.
- ④ 위의 과정을 계속적으로 반복하는 방식으로 영상을 분석한다.

5. 세부 설명

카메라에서 전달된 영상을 캡처 알고리즘으로 분석 한 후, 약속된 동작이 발생했을 경우 마우스 이벤트를 발생시켜 응용 프로그램을 조작, 화면에 디스플레이하는 방식으로 작동한다. 응용 프로그램은 마우스로 제어할 수 있어야 하므로 기존의 CG 시스템이 아닌 실버라이트 프로그램을 사용한다.



[소프트웨어 개념도]



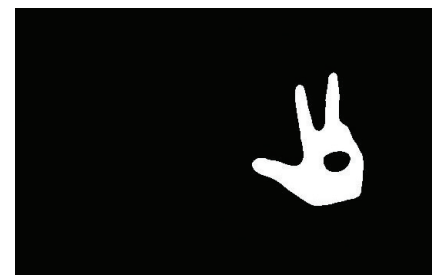
[MFC 구현 예시]

가. 모션캡처 알고리즘(Microsoft Visual Studio/Visual C++ MFC)

원본영상에서 손의 동작을 보다 쉽게 추출하기 위해 색깔이 있는 장갑(붉은색)을 착용한 후 테스트 했다. 카메라 입력으로 들어온 원본영상에서 우리에게 필요한 것은 붉은색 손이므로 영상을 Y/Cb/Cr 성분으로 분리해 Y, Cb 성분은 제거하고 Cr 성분만 남겨 프로세스를 진행한다.



[원본영상을 Y/Cb/Cr 성분으로 분리해 Cr 성분만 추출한 그림]

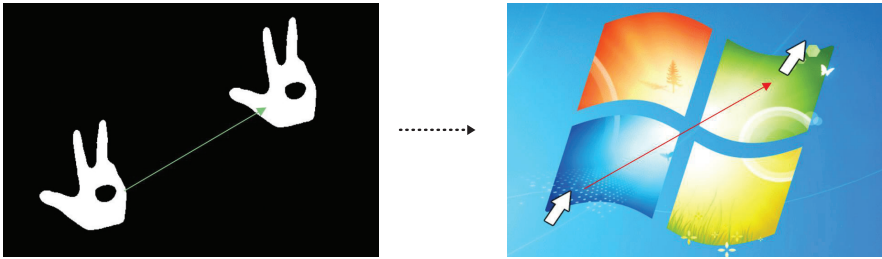


[Threshold를 적용한 Cr 성분 그림]

사실 붉은색 성분은 손에만 있는 것은 아니다. Cr 성분에 나타난 영상을 보면 손부분에 붉은색이 많이 포함되어 있기 때문에 가장 밝게 나타났지만 손 이외의 얼굴, 배경 등에서도 붉은색 성분이 포함되어 있는 것을 알 수 있다. 사실 우리가 알고 싶은 것은 바로 '손' 이므로 손 이외의 영상이 나타난 부분은 쓰레기 데이터(Garbage Data)가 되는 것이다.

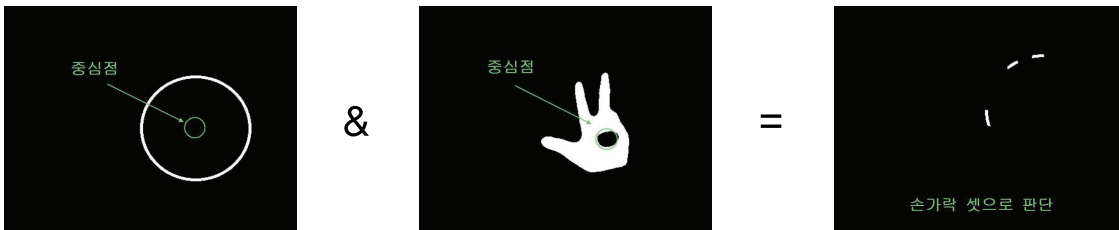
그러므로, Cr 성분 영상의 모든 픽셀에 Threshold 값을 적용해서 임계값 이상의 픽셀에는 255(White) 값을, 이하의 픽셀에는 0(Black) 값을 매칭하여 그레이 영상을 흑백영상으로 단순화시킨다. 이런 단순화 과정은 필요한 부분을 찾는 과정이라는 것 이외에도 필요 없는 정보를 제거함으로써 프로그램의 연산속도를 높이고 동작오류의 가능성도 줄여준다.

이제 손을 추출했으니 이것으로 마우스 이벤트를 발생하는 문제가 남아 있다. 모션캡처 알고리즘이 손(백색영역)의 중심점 좌표를 주시하고 있다가 손이 이동할 경우 중심점 좌표의 이동방향, 거리 값을 읽어 낸 후 이동 값과 같은 양만큼 마우스 포인터 무브(Move)이벤트를 발생시켜 똑같은 방향으로 포인터를 움직이게 한다. 이때 손의 움직임에 대한 여러 가지 오차 값, 노이즈 등을 감안하기 위해 수 프레임간의 영상을 받아들여 움직임이라고 정확한 판단이 될 때 이벤트를 발생하는 조건을 추가한다.



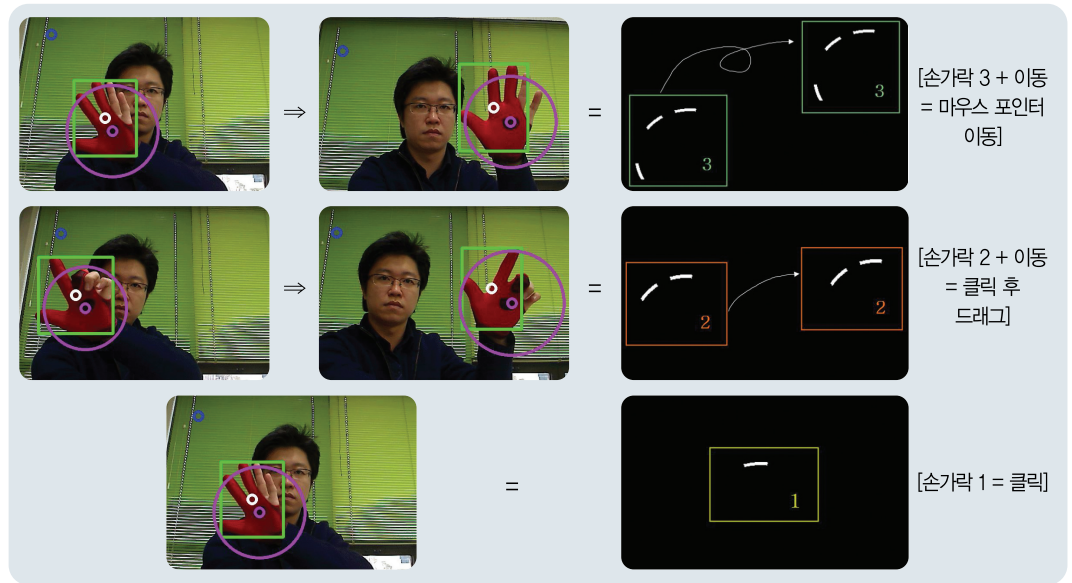
[손의 이동이 마우스 포인터의 이동으로 이어진다]

단지 손의 이동 정보만으로는 응용 프로그램을 조작하기에 어려움이 따르기 때문에 추가정보 획득을 위해 손가락 개수를 파악하기 위한 프로세스를 추가해야 한다. 그러기 위해 손의 중심점을 기준으로 일정크기의 원을 그린 새로운 영상을 구한 후, 원 영상과 손 영상의 각 픽셀들을 AND(교집합) 연산을 하면 두 연산에서 흰색인 픽셀들만 남고 나머지는 검은색으로 변하게 된다. 이때 교집합 영상에서 떨어진 각각의 조각들의 개수가 손가락 개수가 된다.



[AND 연산을 실행하면 손가락 마디만 남는다]

이렇게 구한 손의 이동정보와 손가락의 개수 정보를 조합하여 다양한 모션을 정의할 수 있다. 정의된 모션을 바탕으로 각 모션에 포인터의 이동, 드래그, 클릭 등의 이벤트 매핑을 한 후, 동작으로 매핑 된 이벤트를 발생시키면서 응용 프로그램을 조작하게 된다.



[모션 - 이벤트 매핑 예시]

나. 프레젠테이션용 응용 프로그램(Microsoft Expression Blend/Silverlight)

응용 프로그램은 앞에서 언급한 것과 같이 마우스를 통해 제어할 수 있어야 한다. 여기에서는 제어 조건을 만족하면서 다양한 효과를 구현할 수 있는 실버라이트를 선택했다. 실버라이트란 Microsoft에서 2007년경에 배포한 플랫폼으로서 다양한 UX(User eXperience: 사용자 경험)를 구현할 수 있게 만들어져 있다. 주로 벡터 기반의 그래픽과 애니메이션 제작, HD급의 미디어 재생 등에 초점이 맞춰져 있다.

예전 버전의 실버라이트는 C# 프로그래밍 언어 기반이었기 때문에 일부 프로그래머들만 제작할 수 있었다. 하지만, 최근에는 디자이너들을 위한 Expression Blend Tool이 등장함으로써 프로그램을 모르더라도 Tool의 사용법만 익히면 누구나 손쉽게 그래픽, 애니메이션 효과를 빠르고 손쉽게 제작할 수 있다.

6. 마치며

이번 시스템을 구성하며 가슴 깊이 느낀 점이 있다. 무엇을 하든 기술력도 중요하겠지만, 기존 기술을 개선하고 발전시키고자 하는 창의적이며 긍정적인 마음가짐이 더 중요하다는 것이다. 기술적으로 어려운 부분은 인력 또는 자본이 투입되거나, 시간이 지나면 자연스럽게 해결될 가능성이 충분히 있기 때문이다. 스마트폰 시장에서 삼성전자가 애플에 뒤지고 있는 것도 위와 같은 이유가 아닐까 생각된다.

지금까지 소개한 프레젠테이션 시스템은 아직까지는 방송환경에 적용되지 않은 프로토타입 버전(Prototype version)이다. 하지만, 이와 같은 모션 기반 시스템들을 제대로 발전시킨다면, 수 년 안에 진행자의 동작에 따라 움직이는 그래픽들을 방송에서 볼 수 있을 것이다.