

손동작 인식을 통한 컴퓨터 제어

김희성, 이봉환, 이규원
 대전대학교 정보통신공학과
 e-mail:selrer@nate.com

Computer Control through Recognition of the Hand Movement

Hee-Sung Kim, Bong-Hwan Lee, and Kyu-Won Lee
 Dept. of Information and Communications Engineering, Daejeon University

요 약

사람과 기계의 의사소통을 위해 기본적으로 개발된 방식은 사람의 문자를 기계어로 바꿔주어 작성된 이진 코드들을 기계가 인식하는 방법이었다.[1] 이는 중간 처리기만 이용하여 쉽게 언어를 변환하는 게 가능하였고, 현재까지도 가장 많이 이용되는 방법이다. 하지만 문자를 이용하는 방식 또한 평상시 사람들이 사용하는 언어가 아닌 기계어로의 변환이 용이하도록 미리 정의된 단어들을 사용하기 때문에, 일반적인 사람들이 이용하는 데에는 어려움이 있다. 본 논문에서는 사람의 손동작 영상을 이용한 컴퓨터 제어시스템을 제안하고자 한다.

1. 서론

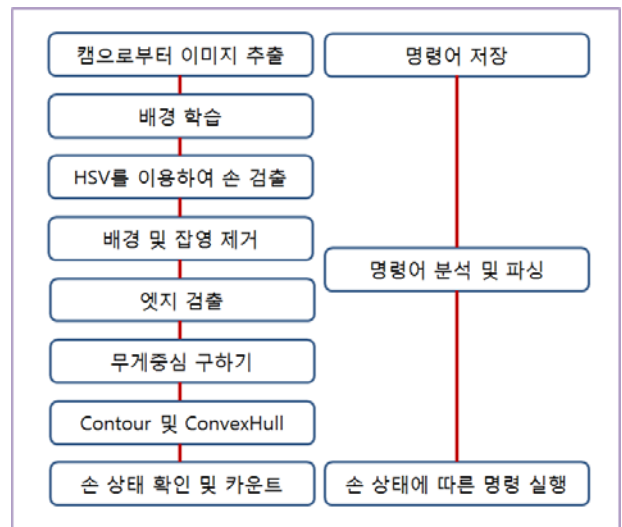
사람 사이의 의사소통은 여러 방식으로 진행 되어질 수 있다. 사람들은 음성을 이용한 대화, 문자, 제스처 등 다양한 방법으로 의사소통을 한다. 현대 사회로 접어들며 기계가 발달됨에 따라 기계 활용도가 높아지고, 사람의 요구나 명령을 기계가 처리하기 위해 사람과 기계사이의 의사소통도 필요하게 되었다. 하지만 초기의 기계는 사람들의 의사소통 방법들을 이해하지 못하였고, 기계어라 불리는 이진화 된 숫자들로 사람의 명령을 수행하였다. 그렇기 때문에 사람들은 자신들이 평상시 사용하는 의사소통 방식으로 기계를 제어하기 원하고, 문자뿐만 아니라 음성과 영상으로도 기계를 제어하는 방식을 필요로 하고 있다.

본 논문에서는 사람의 의사소통 수단 중 영상, 즉 제스처를 이용하는 방식을 사용하고자 하며, 그 중 손동작을 인식하여 그에 따른 컴퓨터의 제어를 수행하고자 한다.

컴퓨터가 사람의 손동작을 인식하기 위해서는 우선 사람이 손동작을 인식하여 처리하는 과정에 대하여 이해해야 한다. 손가락의 길이와 위치, 손바닥 및 팔의 각도 등 사람은 무의식적으로 손모양을 파악하고 분석하여 동작을 인식하게 된다. 컴퓨터에서도 마찬가지로 사용자의 손을 인식하고 손동작을 인식하기 위한 일련의 프로세스를 거쳐 처리 모듈이 실행되게 된다.

사용자는 유동적으로 자신이 원하는 명령을 미리 설정할 수 있고, 손의 상태에 따라 미리 지정된 작업을 수행할 수 있다. 우선 카메라에서 영상을 추출하며, 사용자가 설정한 명령어를 프로그램에 저장한다. 그 후 영상으로부터 일정 프레임동안 배경을 학습한다. 손 영역 검출을 위

해 학습된 배경을 제거한 전경 영상과 HSV의 피부색을 연산하여 손을 검출 한 뒤, 연산 결과에서 잡영을 제거한다. 엣지와 무게 중심을 이용하여 손 경계의 좌표들을 이용하여 손 상태를 확인한다. 전체적인 시스템의 처리도는 (그림 1)과 같다.



(그림 1) 시스템 처리도

2. 손 영역의 검출

본 논문에서는 손 영역을 검출하기 위한 처리 과정으로 두 가지 방법을 병행하였다. 우선 배경학습을 통한 전경 분리로 지정한 일정 프레임동안 영상들의 픽셀 차이를 누적하여 평균을 구한 뒤 실시간 영상과의 비교를 통해 배경과 전경을 분리 하였다. 이 과정에서 손 영역뿐

만 아니라 옷이나 다른 외부 객체들 또한 전경으로 인식하는 오류가 생겼다. 손 영역만 분리하기 위해 피부색을 추출하기 위한 컬러모델로써 HSV를 사용하였다. 그리하여 배경에서 분리된 전경과 HSV로 추출된 피부색 영역의 연산을 통해 손을 검출할 수 있었다.

배경학습과 병행한 컬러모델의 피부색 검출을 통해 손 영역을 검출한 결과, 각각의 방식을 따로 사용하여 손 영역을 검출하였을 때 보다 더 정확한 손 영역 검출이 가능하였다.

2.1 배경학습을 통한 전경 분리

카메라로부터 추출된 영상 이미지는 각 픽셀마다 고유 값들로 이루어져 있다. 프레임의 이미지들은 픽셀들의 변화를 통해 동영상으로 표현된다.[2] 동영상에서 픽셀들은 프레임마다 변화하게 되고, 복수의 프레임동안 변하지 않는 픽셀은 고정된 이미지, 즉 배경으로 인식되게 된다.[3] 배경학습이란 일정시간동안 각 프레임들의 평균을 구하고, 이 평균과 픽셀 값들의 차이를 이용해 특정 임계치를 정한 뒤 임계치 이상의 변화된 픽셀값이 검출되면 전경으로 인식한다. 배경은 프레임의 각 픽셀에 대한 값들의 평균으로 구하며 하나의 픽셀 (x,y) 에서 T프레임동안의 대한 배경은 식 (1)과 같이 계산할 수 있다.

$$P_B(x,y) = \frac{\sum_{t=1}^T P_t(x,y)}{T} \quad (1)$$

차영상은 현재 프레임과 이전프레임의 차이를 말하며, 특정 시간 n에서 임의의 한 픽셀 (x,y) 에 대한 차영상은 식 (2)와 같이 계산할 수 있다. 또한 식 (3)은 정해진 T만큼의 프레임동안 임의의 한 픽셀 (x,y) 에 대한 차이값의 평균을 의미하며, 이는 배경학습 후의 픽셀의 차이가 전경인지 배경의 변화인지를 확인하는 요소가 된다.

$$P_D(x,y) = |P_n(x,y) - P_{n-1}(x,y)| \quad (2)$$

$$P_{Avg}(x,y) = \frac{\sum_{t=1}^T |P_t(x,y) - P_{t-1}(x,y)|}{T} \quad (3)$$

배경학습을 통해 전경을 분리하게 되면, 객체가 영상 영역에 들어왔을 때 객체만 표현하는게 가능해진다. 만약 배경학습이 된 영상 영역에 손이 들어가면 새로 들어온 손 부분만 추출하는게 가능해진다. (그림 2)는 원본 이미지와 배경학습을 통해 전경을 분리하여 이진화한 영상을 나타낸다.



(그림 2) 원본 영상과 전경추출 영상

2.2 HSV 색상 모델을 통한 손 영역 추출

HSV 색상 모델은 색상(Hue), 채도(Saturation), 명도(Value)의 좌표를 써서 특정한 색을 지정하는 표현 방법의 하나이다.[4] 손 영역을 추출하기 위해 HSV를 사용하는 이유는 환경에 따라 변화가 심한 채도와 명도를 가변적으로 지정하고 피부색에 맞는 색상을 기준으로 피부색 영역을 검출하고자 하였기 때문이다.[5]

카메라로부터 추출한 실시간 영상에 대한 정보를 RGB값에서 HSV값으로 변경하기 위해서는 변환함수가 필요하게 되고, 변환하기 위한 방식은 식 (4)와 같다.

$$H = \begin{cases} 0 + (G - B) \times 60 / S, & \text{if } V = R \\ 180 + (B - R) \times 60 / S, & \text{if } V = G \\ 240 + (R - G) \times 60 / S, & \text{if } V = B \end{cases}$$

if $H < 0$ then $H = H + 360$

$$S = \begin{cases} (V - \min(R, G, B)) \times 255 / V, & \text{if } (V \neq 0) \\ 0, & \text{otherwise} \end{cases}$$

$$V = \max(R, G, B)$$

(식 4) RGB와 HSV의 변환

배경학습을 통해 검출한 전경에서 HSV를 이용한 피부색 영역을 추출하면, 손을 검출할 수 있고, 두 영상을 연산하여 손 영역을 추출한 영상은 (그림 3)과 같다.



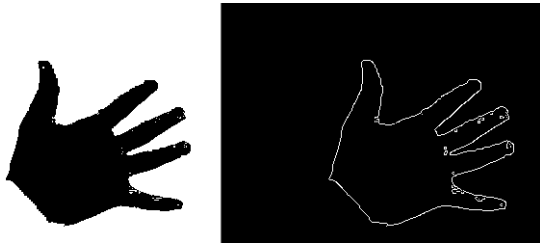
(그림 3) 배경학습과 색상 모델을 이용한 손 영역 검출

3. 손의 경계 및 상태 확인

배경학습과 색상 모델을 이용하여 영상 내에 손 영역을 검출한 뒤, 손의 상태를 확인하기 위해 손 객체의 외곽 점들과 그 좌표들을 이용해야 한다. 우선 검출한 손 영역을 이용해 픽셀이 바뀌는 부분의 경계를 추출하고, 그 경계값들을 좌표로 지정하기 위해 엣지검출과 Contour/ConvexHull을 이용한다.

3.1 엣지 검출

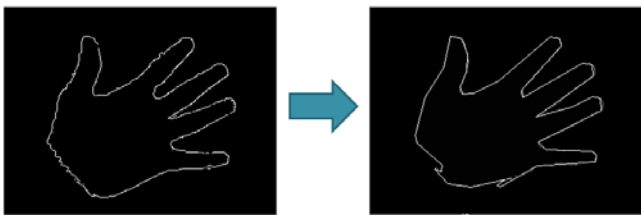
엣지란 가장 자리라는 의미로 윤곽선이라고도 한다. 영상의 밝기가 낮은 값에서 높은 값으로, 또는 그 반대로 변하는 지점에 존재하는 부분을 가리킨다. 영상에서 영역을 검출하는 기법으로는 대부분 칼라 값이 변화하거나 밝기의 변화가 큰 부분이 특정한 물체의 경계를 나타내고, 일정한 분포를 가진 부분은 경계면을 나타낸다고 본다. 따라서 영상 내에서 밝기 변화 값의 차이가 큰 부분을 찾아 객체의 경계를 추출할 수 있다. 엣지 추출은 1차 미분, 즉 밝기 변화율인 기울기를 검출한다. 기울기가 크다는 것은 밝기 값의 변화가 크게 일어난다는 의미로, 영역의 경계면이 존재한다는 것을 나타낸다. 이는 엣지가 상대적으로 다른 명암도를 가진 두 영역간의 경계이기 때문이다.



(그림 4) 손 영역과 엣지 영상

3.2 Contour

검출한 엣지를 이용하여 최외곽 경계들을 좌표값으로 저장하고 처리하기 위해 Contour를 이용한다. Contour는 단어 그대로 윤곽을 표현하며, 윤곽의 좌표들을 저장한다. Contour는 처음으로 나타나는 임의의 경계에 좌표를 정하고 해당 픽셀의 이웃해 있는 픽셀들을 검사하여 윤곽들을 계속하여 검사하는 방식으로 추출할 수 있다. (그림 5)는 원본이미지에서 손 영역의 엣지를 이용하여 Contour를 추출한 영상이다.

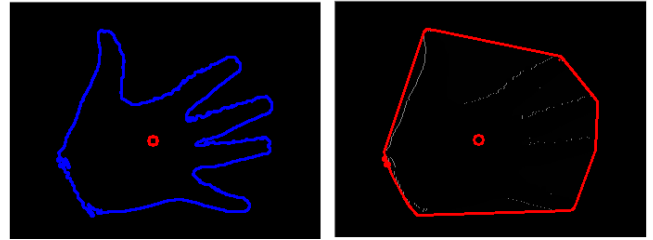


(그림 5) 손 영역과 Contour 검출

3.3 ConvexHull

Contour를 이용하여 윤곽을 검출한 뒤, 윤곽의 점군 중에 가장 각도가 큰 점들을 연결하여 ConvexHull을 구한다. Contour는 개체를 구성하는 점들을 모두 모아놓은 것이고, ConvexHull은 그 중에서도 기울기가 가장 큰 점들을 연결하여 보여준 것이다. Contour와 ConvexHull을 이용하면, 개체의 가장 외곽 좌표들과 개체의 가장 안쪽 좌표들을 구할 수 있으며, 손을 기준으로 Contour와 ConvexHull을 이용하면 손가락들의 끝부분과 손가락의 사이부분을 검출할 수 있게 된다. (그림 6)은 Contour를

이용한 ConvexHull에서 가장 외곽점과 내각점을 표시한 영상이다. 또한 점군들 내에 분포한 픽셀들의 위치 평균을 이용하여 무게 중심을 구할 수 있다.



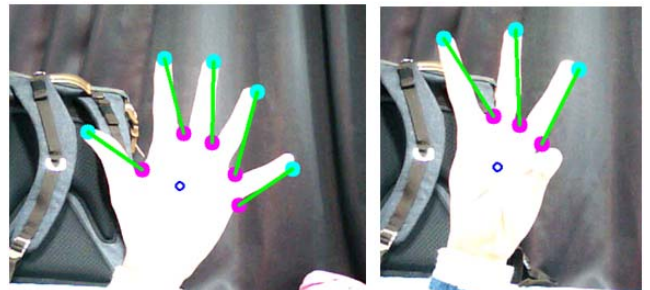
(그림 6) Contour와 ConvexHull

4. 손 정보를 이용하기 위한 전처리과정

손가락의 끝과 손가락 사이의 좌표를 이용하여 손의 상태를 확인할 수 있다. 하나의 손가락을 기준으로 손가락의 시작점과 끝점을 연결하는 선을 긋는다면 손가락이 펴고 굽히는 상태에 따라 선의 길이가 바뀌게 된다. 선들의 길이로 각각의 손가락의 상태를 확인할 수 있고, 선들의 각도와 전체적인 손의 각도를 이용하여 임계값을 지정하면 어떠한 손가락인지 또한 확인할 수 있게 된다.

4.1 손가락의 상태 확인

손바닥의 중심과 손가락의 시작점, 끝점을 이용하여 손가락들의 상태를 확인할 수 있다. 우선적으로 Contour와 ConvexHull 두 윤곽점들의 공통된 점과 차이가 가장 큰 점을 확인해 보면 손가락의 끝과 사이 이외에도 손목 부분에도 좌표들이 확인된다. 이는 손의 중심과 두 개의 점들의 정보(길이, 각도)들을 이용하여 제거할 수 있다. 예를 들어 손가락의 경우 중심에서 손가락의 시작점까지의 길이는 중심에서 손가락의 끝점까지의 길이보다 짧게 된다. 또한 손의 중심에서 손가락 끝까지 길이는 손가락 시작점에서 끝점의 길이보다 일정 비율이상 길어질 수 없다. 이러한 점들의 거리 정보를 이용하여 손목을 제거할 수 있고, 손가락이 펴졌는지 굽혀졌는지 또한 확인할 수 있게 된다. (그림 7)과 같이 이러한 정보들을 이용하여 펴진 손가락만 표시할 수 있게 된다.



(그림 7) 손가락의 상태 확인

5. 실험 및 결과 고찰

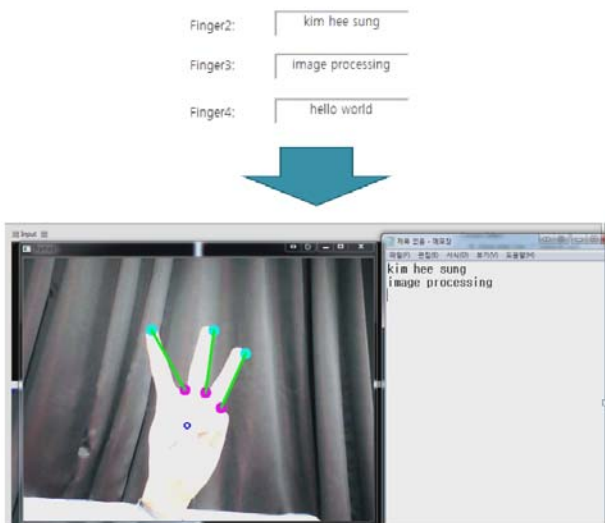
사용자의 요구에 맞게 기능을 구현하기 위해 다양한 방식의 프로세스를 구현해야 한다. 특수키 입력과 어플리

케이션 실행, 문자열 입력 등 다양한 처리를 하기 위해 사용자는 기능들을 선택할 수 있으며, 구현된 프로그램에서는 리스트 형식으로 기능들을 나열하여 선택할 수 있게 하였다. 기본적으로 구현된 리스트들은 (표 1)과 같다.

ESC	특수키 입력
Enter	
Space	
BackSpace	
Ctrl	
Alt	
Page Up	
Page Down	
Insert	
Delete	
Home	
End	
Print Screen	
Application	
Text	텍스트 입력

(표 1) 사용자 선택 기능

사용자가 어플리케이션 실행과 텍스트 입력을 선택하게 되면 FilePath나 문자열을 입력하기 위한 컨트롤이 활성화 되고, 선택된 기능에 따라 프로그램 내부에서 다른 프로세스를 실행시키게 된다. 어플리케이션 실행은 지정한 Path에 응용프로그램을 실행하도록 ShellExecute를 실행하고, 텍스트 입력을 하게 되면, 별도의 파싱 함수를 거쳐 특수키와 마찬가지로 KeyPress를 실행하게 된다. 문자열을 Char형으로 변환하여 하나의 문자마다 파싱을 실행하여 Key Code값으로 변환하고, 절차에 따라 키를 입력하도록 수행한다. 사용자가 입력한 텍스트와 손의 상태에 따른 프로세스 처리 방식은 (그림 8)에서 보이는 바와 같이 실행된다.



(그림 8) 손동작 인식에 따른 프로그램 실행화면

사용자는 손 상태에 따른 명령을 미리 입력할 수 있고, 사용자가 입력한 텍스트나 특수키를 인식 프로세스를 거쳐 타이핑 형태로 실행하게 된다.

실행 결과 잡영 제거를 한 뒤에도 영상에 잡음이 생기거나 외곽선 추출에 오류가 생기기도 하고 떨림 현상으로 인해 손 영역의 추출에 어려움이 생기기도 한다. 이는 외곽선을 선명하고 부드럽게 해주기 위해 엡지 영상에 smooth() 함수를 적용하여 보완할 수 있다.

6. 결론

본 논문에서는 배경학습과 HSV의 피부색 추출을 이용하여, 배경과 피부색이 같은 경우에 해결할 수 있고, 배경학습 과정에서 평균을 이용하고 임계값을 설정하기 때문에 형광등이나 햇빛의 영향을 줄일 수 있다.

또한 잘못된 손동작 인식을 방지하기 위해 손가락 개수만 계산하여 일정 프레임동안 손가락 개수가 변동이 없을 시에 해당 프로세스를 실행시키게 된다. 또한 프로세스 실행에 있어 인식 및 분석 프로세스와 별개의 쓰레드를 생성하여 처리하였기 때문에, 실행 프로그램이 활성화 되지 않은 상태에서도 처리 프로세스를 백그라운드로 실행할 수 있다.

Acknowledgement

본 논문은 미래창조과학부의 고용계약형 SW석사과정 지원 사업으로 수행한 결과임.

참고문헌

[1] 박한훈, 최준영, 박종일, 문광석, “Kinect 기반 손 모양 인식을 위한 손 영역 검출에 관한 연구”, 방송공학회논문지 제 18권 제3호, 2013.
 [2] 주은지, “손동작 인식에 의한 핸드 마우스 구현”, 정보통신전문대학논문집/한밭대학교 제8권 제1호, 2010
 [3] 양선욱, “칼만필터를 이용한 적응적 손영역 획득 시스템”, 국립안성산업대학교 제29집 제2호, 1997
 [4] 김대철, 이철희, 최명희, 하영호, “색 공간 기반의 피부색 검출 방법과 관심 영역을 이용한 피부색 검출 기법”, 한국화상학회지, 18권, 4호, Startpage 16, Endpage 22, Totalpage 7, 2012
 [5] 최봉희, “걸러 영상에서 인체 검사를 위한 피부색 검출”, 대덕대학 제19집, 2001