

윈도우 어플리케이션을 위한 사용자 제스처-명령 (User Gesture-command) 인식 시스템의 구현 Implement of the User Gesture-Command Recognition System for Windows Application

장성원, 심우섭, 박병호, 최용석, 성현경
상지대학교 컴퓨터정보공학부

Jang Sung Won, Sim Woo Sub, Park Byeong Ho,
Choi Yong Seok, Seong Hyeong Kyeong
School of Computer, Information and Communication
Engineering Sangji University

요약

본 논문에서는 윈도우 어플리케이션을 위한 사용자의 손동작 제스처 인식 시스템을 구현하였다. 실시간 제스처 인식 시스템은 사용자의 손 움직임을 웹캠으로 입력받아 제스처 명령으로 구현하였다. MFC Visual C++언어를 기반으로 개발된 인텔사의 OpenCV 라이브러리를 사용하였다. 인식시스템의 유효성을 검사하기 위하여 사용자의 입력을 받아 진행되는 사격게임을 개발 하였다.

I. 서론

컴퓨터 및 주변기기의 성능이 발전함에 따라 영상처리에 대한 관심이 높아지고, 영상으로부터 원하는 정보를 얻기 위한 연구가 활발히 진행되고 있다. 우리는 영상 처리에서 손 추적과 손 동작 인식을 이용하여 윈도우 어플리케이션을 위한 사용자 인터페이스로서 손동작 제스처 인식 시스템을 구현한다. 개발된 인식시스템을 구현하고, 과녁판 사격게임에서 사용자 명령 입력기로 적용하여 그 유효성을 보인다.

II. 시스템 개발 환경

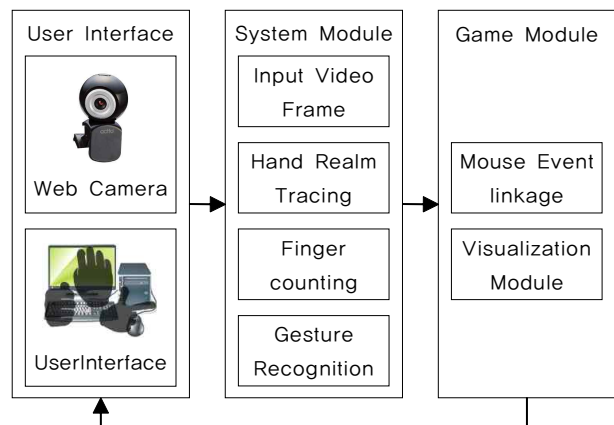
2.1 OpenCV (Open Source Computer Vision Library)

OpenCV는 Open Source Computer Vision Library의 약자로 영상 처리 분야를 위하여 인텔사에서 1999년부터 개발한 C/C++로 개발된 오픈 소스 컴퓨터 비전 라이브러리로서 최적화된 관계로 실시간 어플리케이션에 적합하며 OS, 하드웨어, 창 관리(window-manager)에 독립적인 라이브러리이다. 영상 처리에 있어서 자유로운 영상/비디오 불러들이기, 저장하기, 가져오기, 영상 데이터의 간편한 조작, 영상과 비디오 입출력 지원, 행렬과 벡터 조작, 선형대수(linearalgebra)루틴 지원, 그리고 기본 영상처리 관련 함수를 제공한다. 제공되는 함수 중 컴퓨터 비전 관련 함수에는 구조 분석(structural analysis)함수, 카메라 교정(camera calibration)함수, 움직임 분석(motion analysis)함수, 객체 인식(object recognition)함수, 기계 학습(machine learning)관련함수, 기본 GUI(영상/비디오 출력, 키보드와 마우스, 트랙바 제어)등이 있다.

2.2 어플리케이션

윈도우에서 동작하는 어플리케이션의 개발은 Visual Studio 2008서 OpenCV 2.0Library를 연동하여 MFC 다이얼로그 박스기반으로 개발하였다. 웹캠으로 입력받아 제스처인식을 하기 위해 USB 기반의 LEBECA PRO [PD6000]을 사용하였다.

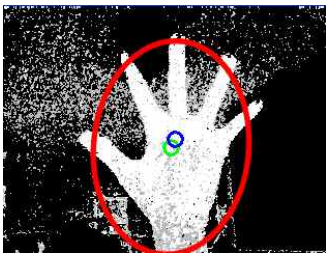
웹캠으로부터 실영상을 입력받아 영상처리를 담당하는 System Module에서 입력받은 Video frame을 영상처리(Image Processing)과정을 거쳐 피부색검출을 통해 손영역을 추출하여, 추적하고, 손가락의 수를 세어 손가락 개수에 따라 제스처를 인식한다. 인식한 제스처를 마우스 이벤트와 연동시켜 과녁판 사격게임인 가시화 모듈에 적용한다. 가위바위보 제스처를 사용하는데, 보는 과녁판 사격 게임모듈을 시작하고, 가위는 과녁조준(Mouse Move), 바위는 발사(Mouse Click)로 연동하였다.



▶▶ 그림 1. 어플리케이션 구조

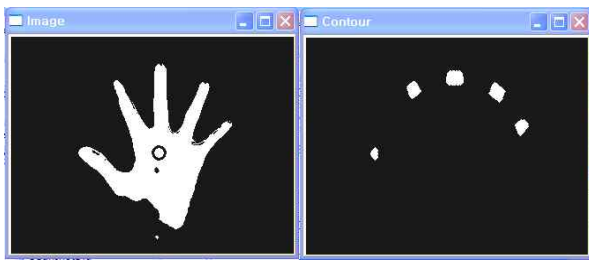
2.3 영상처리(Image Processing) 과정

사용자의 제스처를 인식하기 위해서는 영상처리(Image Processing)과정이 필요하다. 우선 웹으로부터 RGB영상으로 Video frame을 얻어온다. 다음 과정으로는 사용자가 추적할 손 객체 영역을 마우스로 설정한다. 객체의 영역이 선택되면 HSV 컬러 공간으로 변환하고, HSV 컬러공간을 분할하여, Hue(색상)채널만 가져온다. 손 객체 영역 정보를 가지고 관심영역을 가지고 추출하여, 관심영역에 따른 히스토그램을 계산한다. Hue 채널의 이미지와 히스토그램 값을 넣고 backproject(그림2)한 영상을 반환하여, Hue채널 영상과 AND연산을 통해 갱신한다. 후에 CAMShift알고리즘을 이용해 손의 위치를 추적한다.



▶▶ 그림 2. Backproject 영상

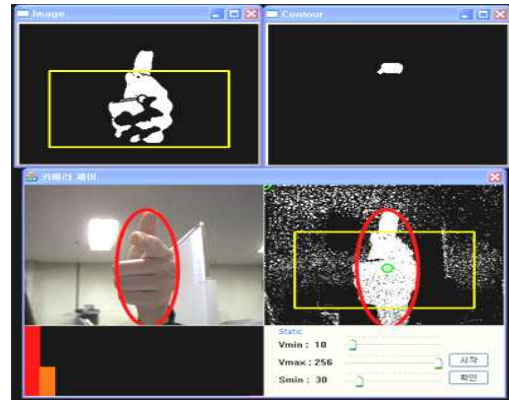
피부색을 선명하게 검출하기 위해 YCbCr 공간으로 변환하여, Y, Cb, Cr 이미지로 분할한다. 피부색으로 여겨지는 부분을 색출하여 노이즈를 줄이기 위해 영상처리를 하고, 팽창과 침식을 통해 구멍난 곳을 메꾼다. 그 다음과정으로 손 중심의 평균값을 계산하고(그림3), 손가락의 개수를 세기위해 부분을 추출하여 팽창을 시킨후 손가락의 개수를 측정한다(그림4). 손가락의 개수는 Palm Center& Palm Circle을 이용하여, 손가락의 수를 센다. 손가락의 수에 따라서 제스처를 인식한다.



▶▶ 그림 3. 손중심 평균값 계산 그림 4. 손가락의 개수 측정

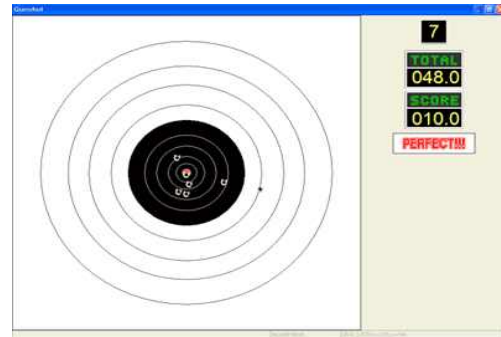
Ⅲ. 구현

본 작품에서 윈도우 어플리케이션을 위한 사용자 인터페이스로서 손동작 제스처 인식 시스템을 구현했다. 구현된 실시간 제스처 인식 시스템은 사용자의 손동작 제스처 명령을 웹캠으로 입력받아서 Intel사의 OpenCV 라이브러리를 기반으로 MFC Visual C++언어로 어플리케이션을 개발하였다. 제스처 인식 System Module을 통해 제스처가 인식되고 있음을 알 수 있다(그림5).



▶▶ 그림 5. 구현된 시스템모듈 애플리케이션

개발된 인식 시스템은 마우스 이벤트와 연동시켜 가시화 모듈 과녁판 사격 게임(그림6)에서 사용자 명령 입력기로 적용해 그 유효성을 보인다.



▶▶ 그림 6. 구현된 시스템 모듈 애플리케이션

IV. 결론 및 향후 연구 방향

본 연구에서는 CAMShift알고리즘과 Palm Center& Palm Circle을 통해 제스처를 인식했다. 제스처 인식 시스템을 구현하여 개발된 인식 시스템은 과녁판 사격 게임에서 사용자 명령 입력기로 적용해 그 유효성을 보였다. 본 연구를 바탕으로 흰배경에서만 인식률이 높았던 점과 거리가 가깝거나 멀면 인식률이 낮았던 점을 보완하고, 신경망을 통한 제스처를 인식을 적용하도록 해 볼 것이다.

■ 참고 문헌 ■

- [1] 정성환, 이문호, 오픈소스 OpenCV를 이용한 컴퓨터 비전 실무 프로그래밍-기본편, 홍릉과학출판사, 2007.
- [2] 황선규, IT EXPERT 영상 처리 프로그래밍 by Visual C++. 한빛미디어, 2007.
- [3] 김건우 외, "제스처 인식 시스템 설계 및 구현". 한국 컴퓨터정보학회 2008 제 38차 하계학술발표논문집, 2008, 06.
- [4] 가상 홈 시스템(VirtualHomeSystem) 구현, 2008.
- [5] OpenCV코리아. www.opencv.co.kr/