

제스처 인식을 위한 손가락 개수 인식 방법[☆]

A Finger Counting Method for Gesture Recognition

이 도 엽¹ 신 동 규^{1*} 신 동 일¹
DoYeob Lee DongKyo Shin DongIl Shin

요 약

인간은 의사소통을 통해서 상호관계를 유지시키고 발전시켜나간다. 의사소통은 크게 언어적 의사소통과 비언어적 의사소통으로 나뉜다. 언어적 의사소통은 말 또는 글을 사용하는 것이고 비언어적 의사소통은 몸동작으로 의사를 전달하는 것이다. 우리는 일상생활에서 대화를 할 때 말과 더불어 제스처를 함께 사용한다. 제스처는 비언어적 의사소통에 속하며, 다양한 형태와 움직임으로 의사를 전달할 수 있다. 이러한 이유로 제스처는 HCI 분야와 HRI 분야에서 NUI/NUX를 구현하기 위한 수단으로 각광받고 있다. 본 논문에서는 키넥트와 손의 기하학적인 특징을 사용하여 손 영역 검출과 손가락 개수를 인식하는 방법을 제안한다. 키넥트가 제공하는 깊이 영상을 이용하여 영상에서 손 영역을 검출하고 손의 윤곽선과 중점의 거리를 비교하여 손가락 개수를 파악한다. 본 논문에서 제안한 방법에 따른 손가락 개수 인식률은 평균 98.5%이고 수행시간은 0.065ms이다. 이 방법은 기존의 연구와 비교했을 때, 인식 속도가 빠르며, 복잡도가 $O(n)$ 으로써 성능 또한 우수하다. 향후 이를 통해 제스처의 인식 가능한 범위를 증가시켜 보다 컴퓨터와 인간의 상호작용이 수월해지는데 도움이 될 것이다.

☞ 주제어 : 키넥트, 손 제스처, 손 제스처 인식, 깊이 영상

ABSTRACT

Humans develop and maintain relationship through communication. Communication is largely divided into verbal communication and non-verbal communication. Verbal communication involves the use of a language or characters, while non-verbal communication utilizes body language. We use gestures with language together in conversations of everyday life. Gestures belong to non-verbal communication, and can be offered using a variety of shapes and movements to deliver an opinion. For this reason, gestures are spotlighted as a means of implementing an NUI/NUX in the fields of HCI and HRI. In this paper, using Kinect and the geometric features of the hand, we propose a method for recognizing the number of fingers and detecting the hand area. A Kinect depth image can be used to detect the hand region, with the finger number identified by comparing the distance of outline and the central point of a hand. Average recognition rate for recognizing the number of fingers is 98.5%, from the proposed method. The proposed method would help enhancing the functionality of the human computer interaction by increasing the expression range of gestures.

☞ keyword : Kinect, Hand Gesture, Hand Gesture Recognition, Depth Image

1. 서 론

인간은 대인 의사소통을 통해 관계를 시작, 유지, 발전시켜나거나 종료시킨다[1]. 대인 의사소통은 한 사람이 다른 사람에게 메시지를 보내는 상호작용 과정이다. 대인 의사소통은 의사소통의 한 종류이며, 의사 전달을 위해 사용하는 방법에 따라 나뉜다. 이를 나누는 기준은 언어

적 요소를 사용하는지, 비언어적 요소를 사용하는가이다. 언어적 요소는 말 또는 글 같은 것이고, 비언어적 요소는 링크, 악수, 웃음과 같은 몸동작이다.

우리는 일상생활에서 대화를 할 때 언어적 요소인 말과 더불어 비언어적 요소인 제스처를 함께 사용하는 경우가 많다. 이는 제스처를 사용하지 않는 대화보다 훨씬 더 정확히 의사를 전달 할 수 있다. 왜냐하면 인간은 대부분의 정보를 시각과 청각을 통해 수집하는데, 비언어적 요소는 시각과 청각정보를 자극하기 때문이다[2].

오늘 날, 다양한 영상 디스플레이 장치가 발전되고 보급되면서 이전과 달리 비하드웨어적 인터페이스의 사용이 보편화되었다. 하드웨어 인터페이스는 마우스나 키보드와 같이 물리적인 장치를 사용해서 컴퓨터와 상호작용

¹ Department of Computer Engineering, Sejong University, 209, Neungdong-ro, Gwangjin-gu, Seoul, Korea

* Corresponding author (shindk@sejong.ac.kr)

[Received 26 October 2015, Reviewed 29 October 2015(R2 21 December 21), Accepted 19 January 2016]

☆ 이 논문은 2015년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임 (No. 2015R1D1A1A01059253)

하는 것이다. 반면, 비하드웨어 인터페이스는 물리적인 장치가 아닌 사람의 신체기관으로 컴퓨터와 상호작용 하는 것이다. 즉, 비하드웨어 인터페이스는 누구나 배우기 쉽고 사용하기 쉬운 인터페이스다. 이러한 인터페이스를 NUI (Natural User Interface) 또는 NUX(Natural User eXperience) 라고 한다. NUI/NUX에 관한 연구는 HCI(Human-Computer Interaction)와 HRI(Human-Robot Interaction)분야에서 활발히 진행되고 있다[3]. 관련 연구들 중에서도, 특히 제스처를 인식하여 컴퓨터와 상호작용이 가능한 인터페이스를 정의하기 위한 연구가 많이 이루어지고 있다. 이는 제스처가 단순하면서도 직관적이기 때문에 직관적으로 컴퓨터를 제어할 수 있기 때문이다[4].

제스처 인식을 위한 연구는 다양한 장비들을 사용하여 진행되어왔다. 데이터 글러브와 비디오 카메라가 대표적인 예이다. 데이터 글러브는 컴퓨터와 상호작용하는 멀티미디어 입력 장치이다. 데이터 글러브를 손에 끼면 글러브 안에 장착된 센서를 통하여 컴퓨터에 위치를 입력한다. 즉, 글러브 안의 센서를 통해 손의 모양과 움직임을 정확하게 측정할 수 있다[5]. 하지만 손의 움직임을 제한하고 추가적인 보정을 요구하여 비용이 많이 들어가는 단점이 있다. 비디오 카메라를 이용한 인식방법은 이슈가 되고 있는 방법이다. 하지만 배경과 손을 분리하는 것이 쉽지 않다는 단점을 가지고 있다. 이는 조명의 변화와 물체의 빠른 움직임, 손 주변에 다른 피부색의 개체가 있을 경우 명확한 손 분리가 어렵기 때문이다.

위와 같은 문제들은, 마이크로소프트사의 키넥트가 출시되면서 해결할 수 있게 되었다. 키넥트는 RGB센서와 IR센서를 사용하여 깊이영상과 RGB영상을 제공한다. 깊이영상은 적외선 영사기에서 조사된 패턴을 적외선 카메라로 읽어 패턴 매칭 방식으로 생성한다. 또한 깊이영상은 깊이정보를 제공하는데, 이는 키넥트와 개체간의 거리를 계산한 값이다. 키넥트와 개체간의 거리가 멀수록 깊이정보는 큰 값을 갖고, 거리가 가까울수록 깊이정보의 값은 작은 값을 갖는다. 키넥트와 개체의 거리는 1.2~3.5m가 가장 효율적인 거리다. 키넥트의 이러한 기술은 기존의 비디오 카메라와 달리 조명에 영향을 받지 않음으로써 배경과 사람의 분리가 가능하게 한다. 또한 주변에 다른 피부색의 개체가 있을 경우에도 정확한 배경 분리를 돕는다.

본 논문에서는 조명과 같은 주변 환경에 영향을 받지 않는 키넥트의 깊이 정보와 손의 기하학적인 특징을 이용한 손 인식방법을 제안한다. 제안하는 방법은 깊이정보를 사용하여 손과 배경을 분리하고 손의 중심점과 손 영

역과의 거리를 비교하여 손가락과 손가락의 개수를 인식한다. 2장에서는 기존의 연구들을 소개하며, 3장에서는 본 논문에서 제시한 손가락 개수 인식 알고리즘을 설명한다. 4장에서는 손가락 개수 인식 알고리즘의 성능을 평가하고 5장에서는 결론을 맺는다.

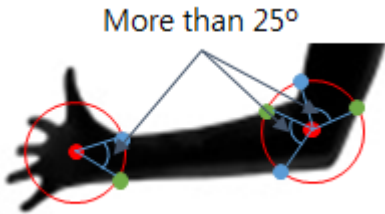
2. 관련연구

제스처는 다양한 디스플레이를 효과적으로 제어할 수 있는 가장 직관적이고 간단한 인터페이스다[6,7]. 이에 따라 제스처를 새로운 인터페이스로 정의하기 위해서 다양한 방법으로 연구가 진행되었다. 그 중, 카메라와 키넥트와 같이 영상입력 장치와 데이터 글러브를 활용한 제스처 인식방법이 각광을 받고 있다. 영상입력 장치를 사용한 연구에서는 깊이정보, 스킨 컬러(Skin Color), 깊이정보와 스킨 컬러를 동시에 사용하는 방법 등이 제시되었다.

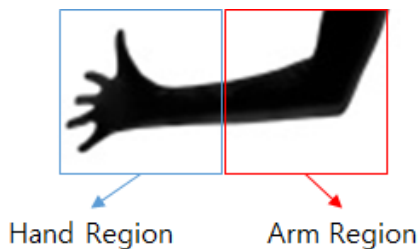
Choi는 데이터 글러브를 활용하여 제스처를 인식하였다[8]. 그는 관성 센서 ‘ADXL330K’ 5개로 구성된 데이터 글러브를 사용했다. ‘ADXL330K’는 운동의 관성력을 검출하여 측정 대상의 가속도, 방향, 거리 등 다양한 항법 관련 정보를 제공한다. 즉, 데이터 글러브의 관성 센서에서 입력되는 X, Y, Z 3축의 G값(-3g~3g)을 분석하여 제스처의 3D방향성을 분석하였다. 분석을 위해서 HMM알고리즘과 다중퍼셉트론(Multilayer perceptron, MLP)을 사용한다. 이는 HMM이 인식 속도가 빠르지만 확률적인 계산 방법으로 결과의 정확도가 떨어지는 단점을 보완한 것이다. 즉, 제스처 데이터 중, 일부를 HMM으로 학습시키고, HMM을 학습하지 않은 데이터로 HMM의 출력확률을 계산한다. 그 후, 계산된 출력 확률 값을 퍼셉트론으로 계산하여 가정 큰 값인 출력확률의 인식 결과를 손가락 포즈를 인식한다.

Choi와 Han은 스킨 컬러를 사용하여 배경과 손 영역을 분리하고 인식하였다[9]. 손 영역 분리를 위해서 카메라를 통해 입력된 컬러 영상에서 스킨 컬러와 유사한 부분을 추출한다. 스킨 컬러를 사용한 이유는 사람의 피부색과 유사한 픽셀 값을 갖는 픽셀을 추출하여 배경과 손의 분리를 쉽게 하기 위해서다. 이 방법을 통해서 검출된 영역은 손 영역으로 인식한다. 하지만 검출된 영역에 오직 손만 있지 않을 수 있다. 다시 말해서, 손 영역뿐만 아니라 팔 영역까지 포함되어 있을 수 있다. 이러한 경우에는 정확한 손 인식이 불가능하게 된다. 따라서 손 영역을 분리하기 위해 손의 기하학적인 특징을 사용하여 팔 영역

과 손 영역을 분리한다. 손 영역과 팔 영역의 분리는 각 영역에 원을 그리고, 영역과 원의 교점 사이의 각도를 통해서 이루어진다. 만약 교점 사이의 각도가 25° 이상인 부분이 2개 이상이면 팔 부분으로 인식한다. 그림 1.a는 이 방법을 통해서 손 영역과 팔 부분을 인식하는 과정이고, 그림 1.b는 손 영역과 팔 부분을 인식한 결과이다.



(그림 1.a) 손 영역과 팔 영역 인식 과정
(Figure 1.a) Recognition process for hand and arm region



(그림 1.b) 손 영역과 팔 영역 인식 결과
(Figure 1.b) Recognition result for hand and arm region

팔 영역이 제외된 손 영역을 취득하면, 해당 영역에서 중심점을 찾는다. 중심점은 손 영역에서 거리변환을 사용하여 픽셀 값이 가장 큰 부분으로 한다. 손의 중심점을 인식한 후, 손가락 개수의 파악을 위해서 손 중심점을 중심으로 하여 원을 그린다. 원의 반지름은 원의 중심점으로부터 손 영역까지의 거리의 1.5배로 한다. 그 후, 시계 방향으로 탐색하여 원과 손 영역의 교점을 찾는다. 교점 중 연속된 두 점의 각도를 계산하여 10° 미만이면 손가락 개수를 증가시키고, 25° 이상이면 손목의 개수를 증가시킨다. 탐색이 종료되면, 현재 계산된 손가락의 개수를 통해서 최종적으로 손가락 개수를 인식했다.

Choi와 Seo는 깊이정보와 키넥트가 제공하는 스켈레톤 트래킹(Skeleton tracking)을 사용하여 손가락을 인식했

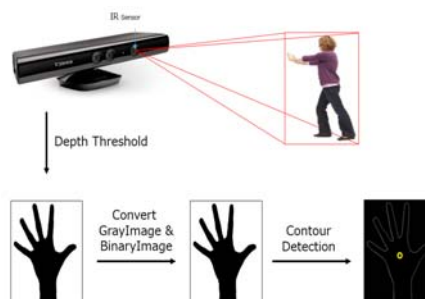
다[10]. 키넥트는 깊이정보를 제공하며, 깊이정보를 사용하여 스켈레톤 트래킹 기능을 제공한다. 스켈레톤 트래킹은 영상에서 인간 신체의 뼈대(골격구조)를 추적하는 기능이다. Choi와 Seo는 이를 활용하여, 손의 스켈레톤을 추적하였고, 손의 스켈레톤 부분의 영역을 손 영역으로 정의하였다. 그리고 거리 변환을 사용하여 손 중심점을 찾았다. 그 후, 손 영역에서 굴곡진 부분을 찾아서 손가락 후보로 설정한다. 하지만 굴곡진 부분은 손 끝만이 아니라 손가락 골도 포함 된다. 이는 후보의 외적 값이 양수라면 후보에서 제외한다. 손가락 골이 후보에서 제거되면, 남은 손가락 후보로 손가락 개수를 인식하였다.

Jagdish는 깊이정보와 openNI모듈을 활용하여 손가락 개수를 파악했다[11]. 그는 우선 키넥트가 제공하는 깊이 영상의 깊이 정보를 사용하여 배경과 손을 분리했다. 일반적으로 제스처를 취하면 키넥트와 손의 거리가 가장 가까워지는 상황이 된다. 즉, 손이 가지는 깊이정보가 가장 작아진다. 따라서 깊이정보에 임계값(Threshold)을 정하면 손이 아닌 배경을 제거할 수 있다. 그리고 베이지안 기반 위치 인식(Bayesian Object Localization)을 사용한 NITE 모듈을 사용하여 손의 포인트를 찾는다. 그 후, 손 영역에서 손의 포인트를 갖는 부분을 제거하여 손가락 영역을 찾는다. 손가락 영역에서 깊이정보가 최소가 되는 부분을 찾아 손가락 개수를 인식하였다.

Tao Hongyoung은 깊이정보와 스킨 컬러를 동시에 사용하여 손가락 개수를 파악했다[12]. 스킨 컬러만 사용했을 경우엔, 손 영역 외에 피부색과 유사한 부분이 검출될 수 있고, 깊이정보만 사용했을 경우엔 잡음(noise)로 인해 정확한 검출이 힘들 수 있다. 하지만 두 가지 방법을 동시에 사용할 경우 한 가지 방법만을 사용했을 때보다 정확히 손 영역을 검출할 수 있다. 우선 손가락 개수를 파악하기 위해서 손 영역을 검출한다. 이는 깊이정보에 임계치를 설정하여 손 영역을 배경과 분리한다. 그리고 YCrCb를 사용하여 영상에서 피부색과 유사한 영역을 검출한다. 그 후, 두 가지 방법을 통해서 얻은 영역에서 공통된 부분을 찾는다. 공통된 부분을 최종적으로 손 영역으로 인식한다. 손가락 개수는 검출된 영역에서 깊이정보가 가장 작은 부분을 찾아 인식했다.

Park은 깊이정보와 컬러영상 및 손의 기하학적인 특징을 사용하여 손 모양을 인식했다[13]. 우선 영상에서 손 영역을 얻기 위해서, 스킨 컬러를 사용하여 얻은 손 영역과 깊이정보를 사용하여 얻은 손 영역의 공통부분을 찾는다. 그 후, 거리변환을 적용하여 가장 큰 값을 갖는 픽셀을 찾아 손바닥 중심점으로 한다. 그리고 손바닥 중심

집을 원의 중심으로 하고, 거리변환의 값을 반지름으로 갖는 원을 그려 손바닥영역을 검출한다. 또한 손가락 끝을 찾기 위해서 손의 외곽선을 검출하고, 기울기가 급격히 변하는 지점을 손가락 끝 후보로 지정한다. 이 때, 후보 중에 손가락 끝이 아닌 손가락 골이 지정될 수 있다. 이런 경우를 제거하기 위해서, 외적의 세 번째 요소가 양수가 되는 곳을 제외함으로 제거시켰다. 이 후, 남은 후보들의 개수를 통해 손가락 개수를 인식하였다. 이외에도 제스처를 인식하기 위해 Choi는 손의 기하학적인 특성을 기반으로 손 영역을 검출하였다[14]. 또한 Cao는 제스처의 특징이 추출된 특징 벡터 등으로 구성된 데이터베이스를 기반으로 손의 포즈와 모양을 인식하였다[15].



(그림 2) 손 영역 검출 알고리즘 순서도
(Figure 2) Flowchart for hand region detection algorithm

3. 손가락 개수 인식 알고리즘

3.1 손 영역 검출

본 논문에서는 키넥트 앞에서 손을 앞으로 쭉 뻗고 손이 위로 향한 상태라고 가정한다. 그림 2는 본 논문에서 제안하는 손가락 개수 인식 알고리즘을 위한 손 영역 검출 알고리즘 순서도이다. 키넥트는 컬러 영상과 깊이 영상을 동시에 제공한다. 깊이 영상은 적외선 센서를 통해 입력되고, 깊이정보를 제공한다. 깊이정보는 키넥트와 개체사이의 거리이며, 거리에 따라 다른 값을 갖는다. 즉, 깊이정보는 키넥트와 개체사이의 거리가 가까울수록 작은 값을 가지고, 멀수록 큰 값을 가진다. 이러한 깊이정보는 3차원형태로 입력되는 제스처를 인식하는데 적합하다.

정확하게 제스처를 인식하기 위한 가장 이상적인 조건은 오직 손 영역만을 가지고 제스처를 분류하는 것이다. 따라서 손 영역만을 얻기 위해서 입력된 영상에서 배경과 손의 분리가 우선적으로 이루어져야 한다. 따라서 깊이정보에 임계값을 설정하여 배경과 손을 분리한다. 일반적으로 제스처를 취하면 손과 키넥트의 거리가 제일 가까기 때문에 손의 깊이정보가 작아지기 때문에 배경과 손의 분리가 용이하기 때문이다. 배경과 손을 분리한 후, 손가락 개수를 인식하기 위해서 영상을 변환한다. 우선, 그레이 영상(Gray Image)으로 변환한 후 이진화 영상(Binary Image)으로 변환한다. 그 후, 손 영역의 윤곽선을 검출한다. 그림 3은 임계값을 설정하여 영상에서 손 영역을 검출한 결과이다.



(그림 3) 임계값 설정을 통한 손 영역 분리
(Figure 3) Detection of hand region by the threshold

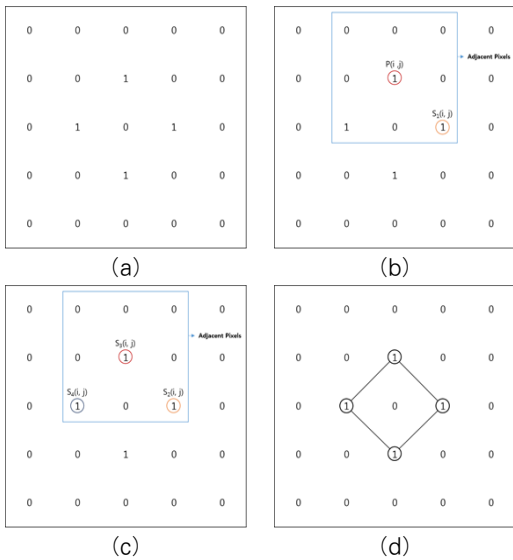
3.2 손 윤곽선 검출

손의 중점과 윤곽선은 컨투어를 활용하여 획득한다. 컨투어는 Suzuki85 알고리즘을 사용하여 검출한다. 컨투어는 인접한 픽셀 값과 동일한 값을 가진 픽셀들의 집합이다. 하지만 키넥트를 통해 입력된 영상의 프레임은 픽셀로 구성되어 있으며 각 픽셀마다 각기 다른 값을 가지고 있다. 이러한 형태로 컨투어를 검출하게 되면 시간이 오래 걸린다. 따라서 이진화를 사용하여 픽셀들의 값을 단순화시킨다. 이진화 된 프레임은 맨 위쪽의 행과 열, 맨 아래쪽의 행과 열로 구성된다. 행은 i 로 표시하고 열은 j 로 표시한다. 또한 i 번째 행과 j 번째 열에 위치한 픽셀은 $P(i,j)$ 로 표시한다. 행 번호 i 는 위에서 아래로 증가하고 열 번호 j 는 왼쪽에서 오른쪽으로 증가한다. 해당 픽셀의 값은 P_{ij} 로 표시한다. 또한 검출된 컨투어를 확인하기 위해서 4개의 픽셀 위치를 저장한다. 저장한 픽셀의 위치는 $S_n(i,j)$ 로 표시한다($1 \leq n \leq 4$).

그림 4.a는 이진화 된 프레임의 예시이다. 1은 프레임에서 임계값 이상에 해당하는 픽셀이고, 0은 임계값 미만의 픽셀이다. 컨투어 검출을 위해서 한 행씩 검사를 하

며, $P(i, j)$ 의 $P_{ij} \neq 0$ 인 경우 아래의 과정을 수행한다. 그림 4.d는 컨투어를 검출한 결과이다.

- (1) $P_{ij} = 1$ 이고 $P_{i,j-1} = 0$ 이라면 경계의 시작점이고, $S_2(i, j) \leftarrow P(i, j-1)$. 만약 없다면 (6)으로 넘어간다.
- (2) $P(i, j)$ 를 중심으로 인접한 픽셀에서 시계방향으로 탐색한다. 탐색은 $S_2(i, j)$ 에서 시작하며, 처음으로 $P_{ij} = 1$ 인 지점을 찾고 해당 위치를 $S_1(i, j)$ 에 저장한다(그림4.b). 만약 없다면 (6)으로 넘어간다.
- (3) $S_2(i, j) \leftarrow S_1(i, j)$, $S_3(i, j) \leftarrow P(i, j)$.
- (4) $S_3(i, j)$ 를 중심으로 인접한 픽셀에서 반 시계방향으로 탐색한다. 탐색은 $S_2(i, j)$ 에서 시작하며, 처음으로 $P_{ij} = 1$ 인 지점을 찾고 해당 위치를 $S_4(i, j)$ 에 저장한다(그림4.c).
- (5) $S_4(i, j) = P(i, j)$ 이고 $S_3(i, j) = S_1(i, j)$ 라면 (6)으로 넘어간다. 아닐 경우 $S_2(i, j) \leftarrow S_3(i, j)$, $S_3(i, j) \leftarrow S_4(i, j)$ 하고 (3)으로 돌아간다.
- (6) $P(i, j+1)$ 부터 이 과정을 다시 시작한다. 만약 $P(i, j)$ 가 프레임의 오른쪽 하단이면 종료한다.



(그림 4) 컨투어 예시
(Figure 4) Contour example

손의 중심점은 검출된 컨투어 좌표의 평균값을 계산하여 찾으며, 윤곽선은 컨투어 좌표들을 연결한다. 그림 5는 검출된 손 영역에서 손 중심점과 윤곽선을 검출한 결과이다.



(그림 5) 손 중심점과 윤곽선 검출
(Figure 5) Detection of the center point and outline of the hand

3.3 손가락 개수 인식

손가락 개수는 손 중심점과 손의 윤곽선의 거리를 이용하여 인식한다. 윤곽선은 컨투어로 구성되어 있기 때문에 컨투어 각각의 좌표를 이용할 수 있다. 따라서 윤곽선을 구성하고 있는 좌표들과 손 중심점과의 거리를 계산할 수 있다. 우선, 손 중심에 손 스�কে레톤과 손목 스�কে레톤의 거리의 0.8배의 길이를 지름으로 갖는 원을 그린다. 그 후, 좌표 탐색을 진행하며 탐색방향은 반 시계방향으로 진행한다. 손가락 개수의 파악을 위해서 연속된 3개의 좌표와 중심점의 거리를 비교한다. 연속된 3개의 값 중 두 번째의 값이 가장 크다면, 그 지점이 손가락 후보가 된다. 이 때, 손의 중심점보다 아래에 있는 컨투어들은 연산 대상에서 제외한다. 탐색이 끝나면, 손가락 후보의 개수를 손가락 개수로 인식한다. 그림 6은 손가락 개수 인식 알고리즘의 유사코드이며, 알고리즘의 복잡도는 $O(n)$ 이다. 그림 7.a는 컨투어를 검출한 결과이고, 그림 7.b는 제시한 방법을 통해서 손가락 개수 후보를 탐색하는 과정이다. 그림 8은 본 논문에서 제안한 방법을 통해서 손가락 개수를 인식한 결과이다.

```

for i = 0 to contourArrLength
    fipoint = contourArr[i]
    if i = 0
        firFinger = fipoint

    if firFingerLength = 0
        firtmpFinger = fipoint
        firLength = sqrt(tmpLength)
/*tmpLength is the distance between fipoint and center
point in hand*/

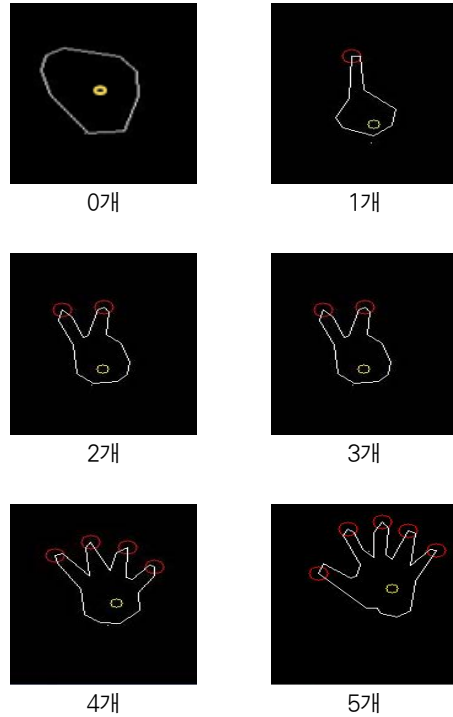
    if secFingerLength = 0
        sectmpFinger = fipoint
        secLength = sqrt(tmpLength)
    
```

```

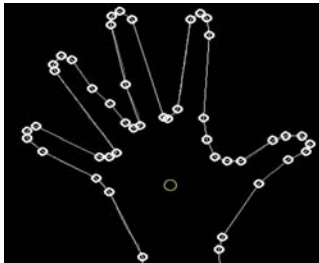
if thiFingerLength = 0
  thitmpFinger = fipoint
  thiLength = sqrt(tmpLength)

if firLength>=secLength or secLength<thiLength
  firLength = secLength
  firtmpFinger = sectmpFinger
  secLength = thiLength
  sectmpFinger = thitmpFinger
  thiFingerLength = 0
elseif secLength>=firLength
  and secLength>=thiLength
  if cy >= sectmpFinger
    /*cy is the y point in center point of hand*/
    fingerCount++
  else
    firLength = secLength
    firtmpFinger = sectmpFinger
    secLength = thiLength
    sectmpFinger = thitmpFinger
    thiFingerLength = 0
    
```

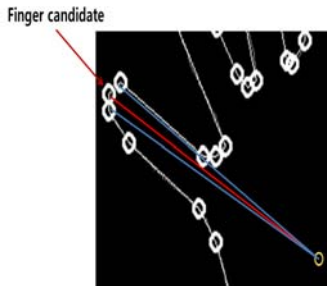
(그림 6) 손가락 개수 인식 알고리즘 유사코드
 (Figure 6) Pseudo-code of finger count recognition algorithm



(그림 8) 거리를 사용한 손가락 개수 인식
 (Figure 8) Recognition of finger count using the distance



(a)



(b)

(그림 7) 컨투어 검출과 손가락 탐색 과정
 (Figure 7) Contour detection and finger navigation process

4. 실험

제시한 알고리즘을 평가하기 위해서 손가락 개수의 인식률을 평가하였다. 실험은 AMD FM(tm)-8150 Eight-Core Processor(3.60 GHz), 메모리 32GB, Window 7 Professional K Service Pack 1(64bit), 윈도우 용 키넥트 v1 카메라로 진행되었다. 실험인원은 20명을 대상으로 진행되었으며 그림 7에서 인식한 서로 다른 6가지의 손가락 개수의 인식률을 비교하였다. 실험은 특수한 상황을 가정하여 환경을 조성하면 정확도의 예측이 가능하기 때문에 일반적인 환경에서 진행되었다. 일반적인 환경이란 실생활에서 웹캠이나 카메라 등을 사용하는 환경으로써, 조명의 변화, 손 주변에 다른 피부색의 개체가 존재 할 수 있는 환경이다. 또한 깊이 영상은 640x480의 해상도를 갖고, 초당 30프레임으로 출력된다. 표 1은 본 논문에서 제안한 방법을 사용한 손가락 개수에 따른 인식률이다.

(표 1) 손가락 개수에 따른 인식률
(Table 1) Recognition rate according to the number of finger

손가락 개수에 따른 인식률						평균 인식률
0	1	2	3	4	5	
100%	100%	98%	96%	98%	99%	98.5%

표 2는 본 논문의 방법과 이전의 연구들을 통해 제시된 방법의 결과를 비교한 것이다. 기존의 방법들 중 데이터 글러브를 사용한 손가락 개수 인식률은 평균 63.4%이고, 깊이정보와 스킨컬러를 동시에 사용한 방법은 평균 98%이다.

(표 2) 기존의 방법들과 제안한 방법의 인식률 비교
(Table 2) Comparison recognition compared to conventional methods and proposed method

개수 \ 방법	Data Glove [8]	Depth & Skin Color [9]	Proposed Method
1개	66%	100%	100%
2개	62%	100%	98%
3개	60%	93%	96%
4개	68%	100%	98%
5개	61%	97%	99%
평균	63.4%	98%	98.2%

표 3은 표 2에서 비교한 방법들과 본 논문에서 제안한 방법의 수행시간을 비교한 결과이다. 데이터 글러브를 사용한 방법은 0.60ms, 깊이영상과 스킨컬러를 사용한 방법은 25ms, 본 논문에서 제안한 방법은 0.065ms의 수행시간이 걸렸다.

(표 3) 기존의 방법들과 제안한 방법의 알고리즘 수행 시간 비교
(Table 3) Algorithm execution time compared to conventional methods and proposed method

수행시간 \ 방법	Data Glove [8]	Depth & Skin Color [9]	Proposed Method
수행시간	0.60ms	25ms	0.065ms

본 논문에서 제안한 방법의 정확도는 평균 98.2%이고 수행시간 0.065ms이다. 다른 방법에 비해 정확도가 높은

것은 주변의 환경의 영향을 받지 않는 깊이정보를 사용하고 컨투어 및 손의 기하학적인 특징을 사용하였기 때문이다. 또한 깊이정보를 사용하였기 때문에 입력된 데이터에 대한 추가적인 연산이나 정제하는 과정을 거치지 않아도 된다. 이에 더불어, 손가락 개수를 인식하기 위한 연산과정 또한 손의 기하학적인 특징을 사용함으로써 복잡하지 않아서 수행시간이 기존의 방법에 비해 빠르다.

5. 결 론

제스처는 음성을 제외하고 가장 직관적인 인간의 의사 표현 수단이다. 이는 제스처가 비언어적인 요소이지만 링크, 악수와 같은 몸짓이 큰 의미를 전달한다는 의미다. 또한 제스처는 다양한 형태를 표현할 수 있기 때문에 인간과 컴퓨터의 상호작용을 위한 수단으로써 각광받고 있다. 이점에 착안하여 본 논문에서는, 제스처를 인식하기 위해서 영상입력장치를 통해 들어오는 영상에서 손가락 개수를 인식하는 방법을 제안했다. 손 영역은 키넥트가 제공하는 깊이 영상의 깊이정보를 활용하여 검출하였고, 손가락 개수는 손 윤곽선을 구성하는 컨투어의 좌표들과 손 중심점의 거리를 비교하여 인식하였다. 또한 제안한 손가락 개수 인식 방법은 평균 인식률이 98.6%이고 수행시간은 0.065ms이다. 기존의 연구에 비해 인식 속도가 빠르고 복잡도가 O(n)으로 성능 또한 우수하다.

향후 연구에서는 손가락 개수와 손의 모양이 가진 특징을 통해서 제스처를 인식할 수 있을 것이다. 또한 이는 제스처의 표현 가능한 범위를 증가시켜 보다 컴퓨터와 인간의 상호작용이 수월해지는데 도움이 될 것이다.

참 고 문 헌 (Reference)

- [1] J.L. Applegate, G.B Leichty, "Managing interpersonal relationships : Social cognitive and strategic determinants of competence," In R. N. Bostrom(Ed.), Competence in communication : A multidisciplinary approach, 1984, pp. 33-56.
- [2] Su jin Chae, "The Importance of Nonverbal Communication Skills," Korean J Med Educ 2010 Jun, 2010, pp. 149-150.
<http://dx.doi.org/10.3946/kjme.2010.22.2.149>
- [3] J.P. Wachs, M. Kolsch, H. Stern and Y. Edan, "Vision-based hand gesture applications," Communications

- of the ACM, vol. 55, 2011, pp. 60-71.
<http://dx.doi.org/10.4016/26819.01>
- [4] Sang Yun Park, Eung Joo Lee, "Hand Gesture Recognition Algorithm Robust to Complex Image," Journal of Korea Multimedia Society, Vol.13, No.7, July, 2010, pp. 1000-1015.
http://www.koreascience.or.kr/article/ArticleFullRecord.jsp?cn=MTMDCW_2010_v13n7_1000
- [5] Lauri Connelly, Yicheng Jia, Maria L. Taro, Mary Ellen Stoykov, Robert V. Kenyon, Derek G. Kamper, "A Pneumatic Glove and Immersive Virtual Reality Environment for Hand Rehabilitative Training After Stroke," IEEE Trans. Neural Systems and Rehabilitation Engineering, vol.18, issue 5, 2010, pp. 551-559.
<http://dx.doi.org/10.1109/tnsre.2010.2047588>
- [6] M. Chen, L. Mummert, P. Pillai, A. Hauptmann, and R. Sukthankar, "Controlling your TV with gestures," Proc. Int'l. Conf. Multimedia Information Retrieval, 2010, pp. 405-408.
<http://dx.doi.org/10.1145/1743384.1743453>
- [7] H. P. Jain and A. Subramanian, "Real-time upper-body human pose estimation using a depth camera," Technical Report, HPL-2010-190, HP Laboratories, 2010.
http://dx.doi.org/10.1007/978-3-642-24136-9_20
- [8] Woon-Soo Choi, Yong-Hwan Cho, Joon-suk Lee, "3-D Gesture Recognition Research using CUDA and Improved Hybrid Neural Networks," Korea Entertainment Industry Association 2011 Annual Spring Conference on Computational Intelligence, 2011, pp. 172-180.
<http://www.dbpia.co.kr/Article/NODE01789961>
- [9] Junyeong Choi, Seiheui Han, Hanhoon Park and Jong-Il Park, "A Study on Providing Natural Two-handed Interaction Using a Hybrid Camera," The Third Interaction Conference on Digital Information Processing and Communications(ICDIPC 2013), 2013, pp. 484-484.
<http://sdiwc.net/digital-library/a-study-on-providing-natural-two-handed-interaction-using-a-hybrid-camera.html>
- [10] Junyeong Choi, Byung-Kuk Seo, Daeseon Lee, Hanhoon Park and Jong-Il Park, "RGB-D Camera-based Hand Shape Recognition for Human-robot Interaction," Robotics(ISR), 2013 44th International Symposium on IEEE, 2013, pp.1-2.
<http://dx.doi.org/10.1109/isr.2013.6695627>
- [11] L. Raheja Jadish , Chaudhary Ankit, Singal Kunal, "Tracking of Fingertips and Centers of Palm using KINECT," 2011 Third International Conference on Computational Intelligence, Modelling and Simulation(CIMSIM), 2011, pp. 248-252.
<http://dx.doi.org/10.1109/cimsim.2011.51>
- [12] Hongyong Tao, Youling Yu, "Finger Tracking and Gesture Interaction with Kinect," IEEE 12th International Conference on Computer and Information(CIT), 2012, pp. 214-218.
<http://dx.doi.org/10.1109/cit.2012.62>
- [13] Hanhoon Park, Junyeong Choi, Jong-Il Park and Kwang-Seok Moon, "A Study on Hand Region Detection for Kinect-Based Hand Shape Recognition," Journal of Broadcast Engineering, Volume 18 , Issue 3, 2013, pp.393-400.
<http://dx.doi.org/10.5909/jbe.2013.18.3.393>
- [14] J. Choi, H. Park, and J.-I. Park, "Hand shape recognition using distance transform and shape decomposition," Proc. of ICIP'11, 2011, pp. 3666-3669.
<http://dx.doi.org/10.1109/icip.2011.6116497>
- [15] C.Cao, Y. Sun, R. Li, and L. Chen, "hand posture recognition via joint feature sparse representation," Optical Engineering, vol. 50, no. 12, 2011, pp. 127210.
<http://dx.doi.org/10.1117/1.3662884>

● 저 자 소 개 ●



이 도 엽(DoYeob Lee)

2014년 학점은행제 컴퓨터공학 공학사

2015년 ~ 현재 세종대학교 대학원 컴퓨터 공학과 석사과정

관심분야 : 데이터베이스, 데이터 마이닝, HCI

E-mail : dy03615@gmail.com



신 동 규(Dong Kyoo Shin)

1986년 서울대학교 계산통계학과 졸업

1982년 Illinois Institute of Technology 컴퓨터과학과 석사

1997년 Texas A&M University 컴퓨터공학과 박사

1997년 ~ 현재 세종대학교 컴퓨터공학과 정교수

관심분야 : 상황인식 미들웨어, 웹기반 멀티미디어, 데이터베이스, 데이터마이닝

E-mail : shindk@sejong.ac.kr



신 동 일(Dong Il Shin)

1988년 연세대학교 전산과학과 졸업

1993년 Washington State University 컴퓨터과학과 석사

1997년 University of North Texas 컴퓨터과학과 박사

1997년 ~ 1998년 시스템공학연구소 선임연구원

1998년 ~ 현재 세종대학교 컴퓨터공학과 정교수

관심분야 : HCI, 데이터 마이닝, CSCW

E-mail : dshin@sejong.ac.kr