

# Real-time Hand Region Detection and Tracking using Depth Information

SungIl Joo<sup>†</sup> · SunHee Weon<sup>††</sup> · HyungIl Choi<sup>†††</sup>

## ABSTRACT

In this paper, we propose a real-time approach for detecting and tracking a hand region by analyzing depth images. We build a hand model in advance. The model has the shape information of a hand. The detecting process extracts out moving areas in an image, which are possibly caused by moving a hand in front of a camera. The moving areas can be identified by analyzing accumulated difference images and applying the region growing technique. The extracted moving areas are compared against a hand model to get justified as a hand region. The tracking process keeps the track of center points of hand regions of successive frames. For this purpose, it involves three steps. The first step is to determine a seed point that is the closest point to the center point of a previous frame. The second step is to perform region growing to form a candidate region of a hand. The third step is to determine the center point of a hand to be tracked. This point is searched by the mean-shift algorithm within a confined area whose size varies adaptively according to the depth information. To verify the effectiveness of our approach, we have evaluated the performance of our approach while changing the shape and position of a hand as well as the velocity of hand movement.

**Keywords :** Hand Region Detection, Accumulated Difference Image, Depth Image, Hand Tracking, Kinect, DAM-Shift

## 깊이정보를 이용한 실시간 손 영역 검출 및 추적

주성일<sup>†</sup> · 원선희<sup>††</sup> · 최형일<sup>†††</sup>

## 요 약

본 논문에서는 실시간 손동작 분석을 위한 깊이정보 기반 손 영역 검출 및 추적 방법을 제안한다. 이를 위해 손 영역 검출단계에서는 깊이 정보만을 이용하여 손 영역의 특징인 형태모델을 생성하고, 검출 시 움직임 정보와 영역 확장(Region Growing)을 통해 객체를 추출한다. 추출된 객체는 사전에 생성된 형태모델과 크기정보를 분석하여 최종 손 영역으로 판정한다. 판정된 손 객체는 추적단계에서 중심점 전이 과정을 통해 이전 중심점과의 최근접점을 획득하고, 최근접점으로부터 영역 확장과 깊이기반 적응적 평균 이동 기법(DAM-Shift)을 통해 새로운 중심점을 검출하여 추적한다. 마지막으로 성능 검증을 위해 다양한 손 모양과 속도 및 위치에 대한 다양한 환경에서 실험하고, 검출속도와 추적된 궤적의 정량적, 정성적 분석을 통해 제안하는 방법의 효율성을 입증한다.

**키워드 :** 손 영역 검출, 누적차영상, 깊이영상, 손 추적, 키넥트, 깊이영상 기반 적응적 평균 이동

## 1. 서 론

지난 10년간 인간의 행동을 자동으로 분석하기 위한 연구가 집중적으로 이루어져왔다. 그 중 HCI 분야에서 주로 인간의 손동작 인식에 흥미로운 관심을 보이며 많은 연구가 진행되어오고 있다. 손동작은 인간이 비언어적으로 표현할 수 있는 효과적인 의사소통 방식으로서 사람들 간의 복잡한 상호

작용을 위해 간단한 동작으로 의사 전달이 가능하다. 기존의 연구들에서는 청각장애인들을 위한 수화 시스템과 같이 정적인 손 인식 연구에서부터 스마트 기기를 동작시키기 위한 효율적인 수단의 손동작 인식을 위한 최신 연구까지 수많은 목적을 위해 다양한 분야에서 적용되어지고 있다.

손동작 인식을 위해서는 많은 다양한 접근법들이 보고되고 있다. 이를 위해서는 손 영역 검출, 손 특징 추출, 그리고 학습 및 인식 방법들이 적용된다. 기존의 연구들은 크게 손 영상을 분석하기 위해 데이터 글로브를 사용하는 방법 [1]과 손의 2차원 패턴 정보를 분석하는 방법 [2-6], 3차원 모델을 사용하거나 스테레오 비전을 이용하여 추출된 손의 3차원 기하학적인 정보를 분석하는 방법 [7-10]으로 분류할 수 있다. 데이터 글로브를 사용하는 방법은 정밀도가 낮고

※ 이 논문은 서울시 산학연 협력사업(SS110013)의 지원을 받아 수행된 연구임.

† 준회원: 숭실대학교 미디어학과 박사과정

†† 정회원: 숭실대학교 미디어학과 Post Doc

††† 종신회원: 숭실대학교 미디어학과 교수

논문접수: 2012년 9월 11일

수정일: 1차 2012년 9월 28일

심사완료: 2012년 9월 28일

\* Corresponding Author: HyungIl Choi(hic@ssu.ac.kr)

전체 시스템과 접속하기 위해 반드시 연결선이 필요하다는 점에서 자연스러운 인터페이스 구축에는 많은 문제점을 안고 있다[11].

카메라의 영상을 이용하는 방법들은 주로 손 피부 색상이나 예지 정보와 같은 영상의 화소 정보들을 이용한다.

색상 정보를 이용하는 방법은 주로 영상에서 초기 영역을 설정하고 영역 내의 피부 색상 모델을 사용하여 피부색상 확률밀도함수를 학습한 후에 손 영역을 분할하여 최종 손 영역을 검출한다[2-4]. Suk[4]은 YIQ모델과 HSV모델을 이용한 피부색상 모델을 생성하고 이를 기반으로 손과 얼굴 영역을 검출 후, 가우스 함수에 의해 얼굴과 손 영역을 추적한다. 이 방법은 손 검출 및 추적을 위한 특징으로 정보의 양이 매우 부족하고, 조명의 변화에 민감하여 효율성이 낮다는 문제점이 있다. 템플릿을 사용하는 방법은 미리 학습한 손 모양 템플릿을 영상에 정합시켜 손의 위치를 검출한다[5][6]. Stenger[6]은 색상과 모션 모델을 이용하여 손을 검출하고 사전 정의된 손 포즈 템플릿과의 거리를 계산하여 실시간 정적 제스처 인식을 수행한다. 그러나 이러한 방법들은 사실상 대부분이 2차원 영상의 화소 데이터를 분석하는 방법들이다. 이 방법들은 오로지 2차원의 영상 공간 안에서만 손동작을 분석할 수 있다. 즉, 이러한 카메라 영상 기반의 손 검출 방법에서는 손의 색상이나 템플릿을 위한 예지 등의 영상 정보가 조명에 따라 영향을 많이 받게 되어 성능에 대한 신뢰성이 낮아진다는 문제점이 존재한다.

손 검출을 위한 또 다른 방법인 센서 기반 방법들은 3차원 데이터를 이용할 수 있기 때문에 영상 기반 방법의 한계점을 해결할 수 있다. 3차원 데이터는 손이 위치하는 실제 공간에서의 위치 정보와 형태 정보를 표현할 수 있으므로 주변 잡음이나 영상의 밝기정보, 색상정보와 무관하게 손을 검출할 수 있다. Breuer[4]은 IR 카메라로부터 입력된 3차원 포인트 클라우드(point cloud)를 PCA를 이용하여 손의 위치와 방향을 측정한 후, 사전에 정의된 손 모델에 데이터를 정합하여 손동작을 인식한다. 그러나 사전에 정의된 모델에 대해서만 인식이 가능하므로 더 많은 자유도의 손 모델이 필요하다는 문제점이 있다. Oikonomidis[5]는 모든 자유도에 대해 손 모델을 정합하기 위한 연구로서 키넥트(Kinect) 카메라 센서[12]를 이용하여 손 모델을 초기화하고 근접한 거리에서 실시간으로 추적한다. 손 형태와 손 모델의 가정된 예제 3차원 구조와 실제 손 간의 차이를 최소화하기 위한 손 모델 파라미터를 최적화한다. 그러나 근접한 영역에서의 손 인식 방법이기 때문에 거리가 멀리 떨어질수록(8cm) 손의 포즈(pose)가 모호해져서 에러가 발생하는 단점이 있다. 그리고 Park[6]은 깊이영상에서의 누적차영상을 통해 손동작 움직임 영역을 검출하고, 웨이브 움직임으로 초기 손 영역을 검출한 후 칼만필터로 추적한다. 그러나 초기 손 검출을 위해 정확한 포즈로 웨이브 동작을 일정 횟수(3회)를 반복 수행해야만 검출이 가능하다는 문제점이 있다. 이러한 3차원 깊이정보들의 오차로 인한 성능 저하문제를 해결하기 위해 2차원 정보와 혼합하여 사용하는 연구들도

이루어지고 있다. Holte[7]는 모션 기반 손 추적 방법으로 ToF 카메라를 이용하여 시점 불변인 제스처 인식 방법을 제안하였다. 3차원 데이터에 기반하여 누적된 영상으로부터 움직임을 검출하는데 이를 위해 두 장의 깊이영상에서 픽셀 기반 깊이 값 차이를 계산한 2차원 이중 차이값의 3차원 시각 정보를 이용한다.

본 논문에서는 실험 환경의 조명 조건에 영향을 받지 않고, 색상 정보에 무관하게 안정적으로 손을 검출하기 위해 키넥트 카메라 센서로부터 입력되는 깊이정보를 이용하며, 검출된 손 영역의 중심점 추적 및 갱신 방법을 제안한다. 카메라로부터의 거리와 속도에 따른 손 영역 추적 결과를 정량적, 정성적 분석을 통해 깊이영상 기반의 제안하는 방법의 타당성을 입증하고, 평균 손 검출속도 및 연산속도, 추적의 성능을 분석하여 시스템의 효율성에 대해 설명한다.

본 논문의 구성은 2장에서는 실시간 손 검출 및 추적 시스템의 구성에 대해 설명하고, 3장에서는 본 논문에서 제안하는 깊이영상을 이용한 손 영역 검출을 설명하고, 4장에서는 깊이기반 적응적 평균 이동 기법(Depth based Adaptive Mean Shift, DAM-Shift)를 사용하여 검출된 중심점을 추적하는 방법에 대해 설명한다. 그리고 5장에서는 제안한 알고리즘의 검증에 대한 실험 환경과 결과를 기술한다. 마지막으로 6장에서는 결론 및 향후 연구방향에 대해 제시한다.

## 2. 시스템 구성

본 장에서는 Kinect센서를 통해 획득한 조명과 색상에 안정적인 깊이영상만을 이용하여 다양한 형태를 갖는 손 영역을 검출하고, 검출된 손 영역의 중심점을 추적하는 방법을 제안한다. Fig. 1은 전체 시스템 구성을 보여준다. 먼저 깊이영상이 입력되면, 추적객체가 있는지 판단 후 없다면 손 영역 검출 프로세스로 진입하고 추적객체가 있다면 손

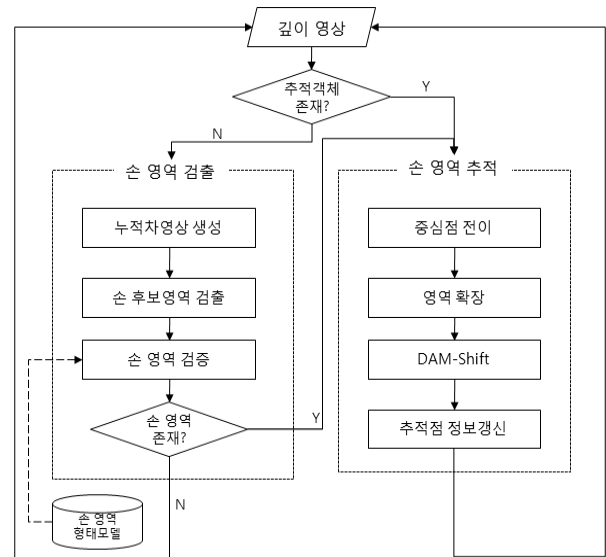


Fig. 1. System Architecture

영역 추적 프로세스를 수행한다. 손 영역 검출 프로세스에서는 누적차영상 기법을 이용하여 움직임을 검출하고 움직임 영역 중 영역 확장을 통해 손 후보영역을 획득한다. 획득된 손 후보영역은 검증단계에서 사전에 학습된 손 영역의 형태모델과 크기를 분석하여 손 영역인지 판단하게 된다. 만약 손 영역이라 판단되면 다음 프레임부터는 추적 프로세스를 진행하여, 이전에 찾은 중심점으로부터 중심점 전이 과정을 통해 최근접점을 검출하고, 최근접점을 기준으로 영역 확장을 통해 손 영역을 분할한다. 분할된 손 영역에서 거리에 따라 적응적으로 변하는 커널을 사용하는 깊이 기반 적응적 평균 이동 기법(DAM-Shift)을 통해 새로운 중심점으로 갱신함으로써 실시간 손 영역 추적을 수행한다.

### 3. 손 영역 검출

본 논문의 목적은 제스처 인식에 사용가능한 손 영역의 레저적 정보 추출을 위해 손 영역을 검출하고 추적하는 방법을 제안한다. 이를 위해서는 먼저 손 영역을 추출해야한다. 손 영역 추출 방법은 크게 2 단계로 구성된다. 먼저 움직임 정보와 깊이정보를 이용하여 관심영역을 검출하는 후보영역 검출단계와 검출된 관심영역 중 사전에 학습된 손 영역 형태모델을 통해 검증을 수행한 후 최종 손 영역을 검출하는 손 영역 검증단계를 수행한다.

Fig. 2는 손 영역을 검출 시 기본적인 자세의 한 예를 나타낸다. 그림과 같이 손바닥은 팔꿈치보다 앞에 있고 손가락은 편안하게 펴진 상태라 가정한다. 이러한 이유는 본 연구의 손 영역 검출과정에서 사용하는 Distance Transform (DT) 알고리즘[13]으로 검출한 손의 중심점이 학습된 모델과 비교 시 매우 중요한 기준점으로 사용되는데, 손바닥이 팔꿈치보다 뒤에 있다면 손의 중심점이 다를 수 있기 때문이다. 하지만 손 영역 검출 시 위치, 회전, 크기에 불변한 서술자를 사용함으로써 너무 정형화된 자세로 가정하지는 않는다.

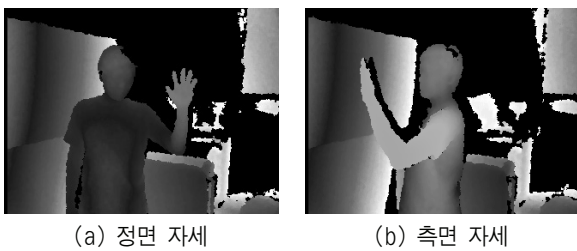


Fig. 2. Fundamental position for hand detection

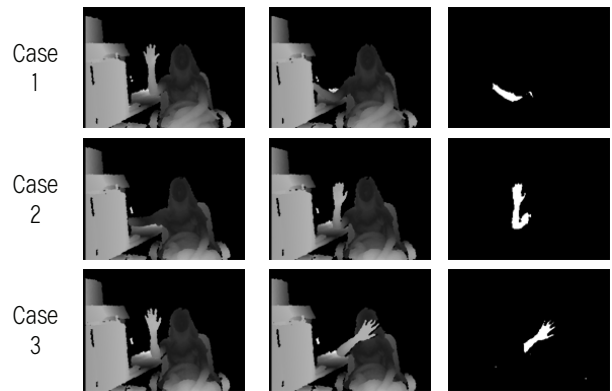
#### 3.1 후보영역 검출

후보영역 검출은 손 영역 검증을 위한 입력을 생성하는 단계로서 움직임 정보를 이용한다. 움직임 정보를 좀 더 안정적으로 획득하기 위해 다음과 같은 조건을 사용한다.

[수식 1]은 차영상 생성 조건을 나타낸다. 먼저  $d_t(x,y)$ 는  $(x,y)$ 좌표에서의 깊이 값을 나타내며 카메라에 근접한

경우 작은 값을 갖는다. 또한  $T_s$ 는 임계값으로서 본 논문에서는 실험을 통하여 50으로 설정하였다. 일반적으로 차영상은  $t$ 프레임과  $t-1$ 프레임 차이의 절대값을 사용하여 임계값 이상 차이가 날 경우 255로 설정하여 영상을 생성한다. 하지만 본 논문에서는 깊이영상을 이용하기 때문에 깊이영상의 특성을 적용하여 절대값을 취하지 않았다.

$$S(x,y) = \begin{cases} d_{t-1}(x,y) - d_t(x,y) > T_s & 255 \\ otherwise & 0 \end{cases} \quad (1)$$



(a)  $t-1$  프레임 (b)  $t$  프레임 (c) 차영상  
Fig. 3. Difference image using depth information

Fig. 3은 [수식 1]을 사용하여 획득한 세 가지 케이스에서의 차영상 결과를 나타낸다.

Case 1 : 손이 있다가 사라진 경우

$t-1$ 시점의 손바닥 위치의 깊이 값이  $t$ 시점에는 더 큰 수를 갖기 때문에 [수식1]에 대입할 경우 음수 값이 나온다. 따라서 손바닥이 차영상에 적용되지 않은 결과를 나타낸다.

Case 2 : 손이 나타난 경우

이 경우는 첫 번째 경우와 반대 경우로 검출된 손바닥 위치의 깊이 값이 이전 프레임의 깊이 값보다 작아졌으므로 양수를 갖고 임계값 이상이므로 255로 검출된 결과를 볼 수 있다.

Case 3 : 손이 움직인 경우

절대값을 이용한 경우에는  $t-1$ 시점의 손 영역과  $t$ 시점의 손 영역이 함께 검출되지만 절대값을 사용하지 않아  $t$ 시점의 손 영역 위치에서만 손 영역이 검출된 결과를 볼 수 있다. 하지만 단순 차영상만 이용할 경우 객체가 천천히 이동할 경우 문제가 된다. 정확한 객체의 영역을 추출할 수 없기 때문이다. 이러한 문제 해결을 위해 차영상을 일정 시간 누적하여 사용한다.

[수식 2]는 누적차영상을 생성하는 방법이다.  $A$ 는 누적차영상이며,  $S(x,y)$ 는 [수식 1]로부터 얻은 차영상이다.  $n$ 은 누적할 프레임 개수이다. 이렇게 구한 누적차영상의 전경화소 중 일부는 움직인 객체의 영역을 포함한다.

$$A(x,y) = \begin{cases} \sum_{i=t-n+1}^t S_i(x,y) > 0 & 255 \\ otherwise & 0 \end{cases} \quad (2)$$

[수식 1]과 [수식 2]에 의해서 전경 화소는 근접한 거리의 객체의 움직임을 표현한다. 따라서 먼저 누적차영상에서 전경 화소들을 연결성 분석을 통해 객체 단위로 그룹화하고, 객체 영역 내부에서 가장 근접한 위치의 있는 화소를 선택한다. 선택된 화소의 위치를 기준으로 주위에 급격하게 깊이 변화가 일어나지 않는 화소들을 하나의 영역으로 그룹화하여 후보영역을 검출하게 된다.



Fig. 4. Candidate Hand Region Detection Result

Fig. 4는 손 후보영역 검출 결과를 보여준다. (a)는  $t$ 시점의 깊이영상이며, (b)는  $t-n+1$ 부터  $t$ 프레임까지 누적하여 얻은 누적차영상이며, 점  $P$ 는 누적차영상의 전경 화소들 중 현재 시점의 깊이영상에서 가장 앞에 있는 화소의 위치를 나타낸다. 또한 점  $P$ 는 누적차영상의 각 영역별로 각각 검출된다. (c)는 (b)에서 선택된 위치  $P$ 를 기준으로 주위 8방향을 유사한 화소일 경우 연결하여 확장 생성된 객체 영역이다.

[수식 3]과 [수식 4]는 영역 확장 시에 적용되는 검사 조건을 나타낸다. [수식 3]의  $P$ 는 이전에 구한 누적차영상의 전경영역 중 카메라로부터 가장 근거리에서 있는 화소의  $x, y$  좌표를 나타내며, 점  $p$ 는 영역 확장을 할 것인지 검사될 화소의 좌표이다. [수식 4]의  $p_x$ 와  $p_y$ 는 마지막으로 확장된 영역의 좌표이며  $d(p)$ 는  $p$ 좌표의 깊이 값을 나타낸다. [수식 3]과 [수식 4]는 주위 8방향에 대해 확장 검사 시 깊이 값 차이가 임계값 이하의 조건을 만족함과 동시에 처음에 찾은  $P$  위치로부터 임계값  $T_{gd}$  거리 안에 포함되어야 함을 의미한다. 이렇게 얻은 객체 후보영역들은 손 영역 검증 과정을 통해 추적 여부를 결정하게 된다.

$$\| P-p \| < T_{gd} \quad (3)$$

$$d(p_{x,y}) - d(p_{x+i,y+j}) < T_{dep} \quad -1 \leq i, j \leq 1 \quad (4)$$

### 3.2 손 영역 검증

#### 1) 손 영역 형태모델 생성

손 후보영역을 검출하였다면, 해당 후보영역이 손 영역인지 판단해야한다. 이를 위해 본 논문에서는 손 영역의 학습

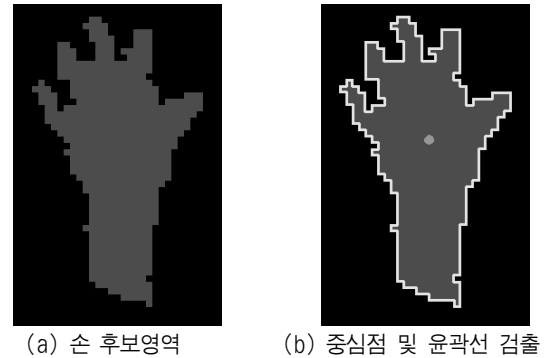


Fig. 5. Feature Extraction

영상을 수집하여 손 영역의 형태를 모델링함으로써 손 영역 검증과정을 위해 사용한다.

Fig. 5은 특징 추출을 하기 위한 전처리 과정의 결과를 보여준다. (a)는 이전 단계에서 구한 후보영역이며, (b)의 손 바닥 중심에 있는 점은 DT 알고리즘을 사용하여 가장 큰 값을 갖는 좌표이며, 황색 경계선은 (a)의 적색 영역을 전경으로 하여 얻은 윤곽선 검출 결과이다.[14]

손 영역의 윤곽선은 손의 위치, 카메라로부터 손이 위치한 거리, 손의 회전에 따라서 윤곽선의 크기와 길이, 순서가 달라진다. 따라서 손 영역 형태모델을 만들기 위해 윤곽선의 크기, 이동, 회전에 불변하도록 특징을 생성해야한다. 먼저 윤곽선의 길이가 손 영역의 크기에 따라서 달라질 수 있다. 특징 간 비교를 위한 편의성을 위하여 특징벡터의 길이는  $n$ 개로 정규화한다. [수식 5]에서  $p=(x,y)$ 로서 좌표를 의미하며,  $F_M$ 은 윤곽선의 길이에 상관없이 같은 길이의 특징벡터를 생성하기 위해  $n$ 개로 샘플링 된 특징벡터이다.  $n$ 은 64개로 실험적으로 정의하였다.

$$F_M = \{p^0, p^1, \dots, p^{n-1}\} \quad (5)$$

다음은 이동에 불변인 특징벡터 생성을 위해 이전에 DT에 의해 획득한 극점을 중심이 되도록 각 윤곽선의 좌표를 이동한다. [수식 6]은 이동 불변을 위한 수식이며,  $D_m$ 은 DT의 극점을 나타낸다.

$$p^i = p^j - D^m \quad (6)$$

또한 크기에 불변하기 위해 [수식 7]과 같이 기준점  $D_m$ 으로부터 윤곽선을 이루는 모든 점들의 거리 합이 1이 되도록 정규화시키고, 중심점으로부터 가장 멀리 있는 윤곽선상의 한 점을 기준으로 회전시켜 시계 방향으로 특징벡터를 생성한다. [수식 8]은 기준점을 선택하는 수식으로 중심점  $D_m$ 으로부터 거리가 가장 먼 점을 기준점으로 선택하며, [수식 9]는 선택된 기준점의 회전각을 구하는 수식이다. 이렇게 구한  $\theta$ 를 이용하여 모든 점들을  $-\theta$ 만큼 회전하여 회전에 불변하도록 한다. 또한 시작점이 다를 수 있으므로 시작점

또한  $p^{\max}$ 를 기준으로 시계방향으로 재배치하여 최종 특징 벡터를 생성하게 된다.

$$p^i = \frac{p^i}{\sum_{k=0}^{n-1} \|D_m - p^k\|} \quad (7)$$

$$p^{\max} = \operatorname{argmax}_i (\|D^m - p^i\|) \quad (8)$$

$$\theta = \operatorname{atan}\left(\frac{p_y^{\max} - D_y^m}{p_x^{\max} - D_x^m}\right) \quad (9)$$

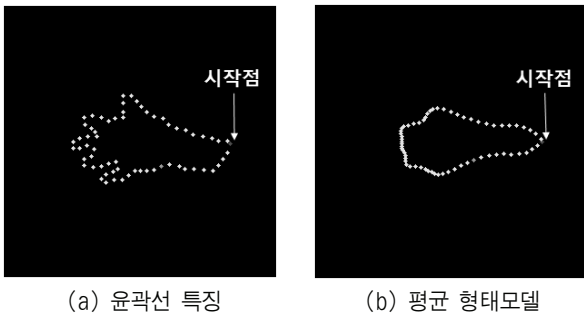


Fig. 6. Hand shape model

Fig. 6의 (a)는 위의 방법대로 얻은 하나의 학습영상에 대한 특징벡터를 확대한 그림이며, (b)는 모든 학습영상으로부터 구한 특징벡터를 평균하여 구한 모델이다. (a)와 (b)의 시작점으로 표기한 점은 특징벡터의 시작점을 나타내며, 시계방향의 순서로 조합된 좌표들이 최종 특징벡터이다. 평균 형태모델에서 손가락 부분이 단순화되어 표현되어있는데, 이는 오른손과 왼손, 다양한 회전, 모양에 대한 학습영상을 사용하였기 때문이다.

### 2) 손 영역 검증

손 영역을 판단하기 위한 모델을 생성하였으므로, 검출된 후보영역이 손 영역인지 검증할 수 있다. 검증을 위한 척도로는 위에서 구한 특징벡터를 사용한 형태 정보와 손 영역의 일반적인 크기가 존재한다는 가정을 이용하여 검증을 수행한다.

먼저 후보영역이 손 영역인지 판단하기 위해서 크기 정보를 사용한다. 손의 경우 거리에 따라서 일정 범위의 손 영역의 크기 정보를 갖는다. 예를 들어 손 영역이 카메라로부터 원거리에 위치해 있는 경우 영역의 크기는 상당히 작을 것이며, 카메라 근거리에 있는 경우 영역의 크기는 원거리에 있는 경우보다 클 것이다. 또한 사람의 손 영역은 일정 범위의 크기를 갖는다. 이러한 사전 정보를 이용하여 본 논문에서는 2차 선형 모델[9]을 사용하여 깊이, 즉 거리에 따른 손 영역의 크기를 정의한다. 이렇게 정의된 거리에 따른 손 영역의 크기를 기준으로 일정 범위에 안에 속하는 영역

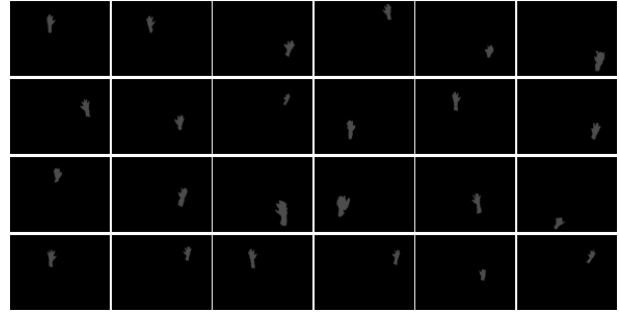


Fig. 7. Training data for 2nd linear model

만을 2차적으로 형태 정보를 분석하여 최종 손 영역이라 판단하게 된다.

Fig. 7는 크기 정보를 이용하여 모델링하기 위한 학습 데이터의 예를 보여준다. 그림과 같이 다양한 형태, 거리에 위치해 있는 손 영역을 검출하여 학습 데이터에 포함시킨다. 이렇게 수집된 각 학습 데이터에서 DT를 이용하여 최고 극값을 갖는 손바닥 중심 위치를 찾고, 찾은 중심 위치의 깊이 값을 획득하고, 영역의 크기 정보를 획득한다.

[수식 10]은 깊이에 따른 영역의 크기를 2차 선형 모델을 사용하여 나타낸 수식이다.  $y$ 는 [수식 12]와 같이 각 학습 데이터로부터 얻는 영역의 크기( $r$ )로 이루어진 열벡터이며,  $P$ 는 손 영역의 중심으로부터 추출된 깊이 값( $x$ )을 이용하여 만든  $n \times 3$  행렬이다. 또한  $\alpha = [\alpha_1 \ \alpha_2 \ \alpha_3]^T$ 이다. 위에서 손 영역이 카메라로부터 거리에 따라서 영역의 크기가 일정한 범위에 속한다는 사전 정보를 이용하여 거리에 따른 영역의 크기를 유추하기 위해 2차 선형 모델을 사용하였다. 따라서 [수식 11]을 이용하여  $\alpha$ 를 구하면 깊이 값에 대응하는 영역의 크기 정보를 산출할 수 있다.

$$y = P\alpha \quad (10)$$

$$\alpha = (P^T P)^{-1} P^T y \quad (11)$$

$$P = \begin{bmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 \end{bmatrix}, \quad y = \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_n \end{bmatrix} \quad (12)$$

후보영역이 검출되면 중심점을 찾고, 중심점의 깊이 값을 이용하여 [수식 10]에 대입하면  $y$ 를 구할 수 있다.

$$h(x) = \begin{cases} True & |y - Region| < T_r \\ False & otherwise \end{cases} \quad (13)$$

[수식 13]은 [수식 10]에 대입하여 얻은  $y$ 값과 현재 영역의 크기인  $Region$ 의 차이가  $T_r$ 보다 작다면 1차 검증에 성공한 것을 의미한다. 1차 검증에 성공한 후보영역은 2차적

으로 3.2.1절에서 생성한 평균 형태모델을 이용하여 형태학적인 검증 작업을 수행한다.

[수식 14]는 평균 형태모델과 1차 검증에 성공한 후보영역의 형태학적인 차이값을 나타내는 수식으로  $F_M(i)$ 는 평균 형태모델의  $i$ 번째 특징을 나타내며,  $f(i)$ 는 후보영역의  $i$ 번째 특징을 나타낸다. 이렇게 구한  $Diff$ 가 임계값  $T_m$ 보다 작을 경우 최종적으로 손 영역이라 판단한다.

$$Diff = \sum_{i=0}^{n-1} \| F_M(i) - f(i) \| \quad (14)$$

### 4. 손 영역 추적

제스처 인식을 하기 위해서는 손의 형태를 인식하거나 손 영역의 궤적을 검출하는 과정이 필요하다. 본 논문에서는 이러한 궤적을 검출하기 위해 손 영역의 중심을 추적하는 방법을 제안한다. 추적을 수행하기 위해서는 시작점이 있어야 하며, 다음 프레임에서의 이동된 위치를 찾아 추적을 수행하기 위해서는 이전 프레임과의 연관성을 분석하여 정합 또는 판단하는 방법이 필요하다. 따라서 이러한 요건을 충족하기 위해 추적 단계에서는 2단계 과정을 수행한다. 첫 번째로 중심점에 대한 검색 영역을 줄이기 위해 이전에 찾은 중심점을 기준으로 최근접점을 검출한다. 두 번째 단계는 추적점을 항상 같은 위치로 하기 위해 깊이기반 적응적 평균 이동 기법(DAM-Shift)을 사용하여 새로운 중심점을 구한다. 이렇게 2단계 과정을 반복적으로 수행함으로써 손 영역을 추적한다.

#### 4.1 중심점 전이

이전 단계에서 손 영역을 검출하고, 손 영역의 중심점을 검출하였다. 본 논문에서는 이러한 검출된 손 영역의 중심점을 기준으로 최근접점을 검출하고[15], 이를 바탕으로 손 영역 객체를 검출한다.

Fig. 8은 중심점 전이 방법을 나타낸 그림이다. 먼저 (a)는 손 영역 검출 단계에서 찾은 손 영역의 중심점을 포함한 깊이영상이다. (b)는 다음 프레임 영상으로 손 위치가 이동되었음을 알 수 있다. 본 논문에서는 손을 추적하기 위해서 매우 간단하고 효과적인 방법을 제안한다. (b)에서와 같이 이전 중심점으로부터 가장 가까운 점을 찾는다. (b)의 최근접점이 이전 중심점으로부터 가장 가까운 점이다. (c)의 영상은 검출된 최근접점으로부터 3.1에서 설명한 영역 확장 방법을 통해 분할된 깊이영상이며, (d)는 영역 확장 결과의 2D 영상이다.

[수식 15]는 최근접점을 찾는 수식으로  $TP_t^{pre}$ 는 이전 프레임의 중심점을 나타내며,  $p$ 는 비교하고자하는 임의의 점이다. [수식 16]은 일반적으로 사용하는 유클리디안 거리이다. 하지만  $x$ 축과  $y$ 축의 거리는 단순히 두 점간의 좌표 차이를 계산하지만 거리 정보인  $z$ 축으로 차이는 [수식 17]과

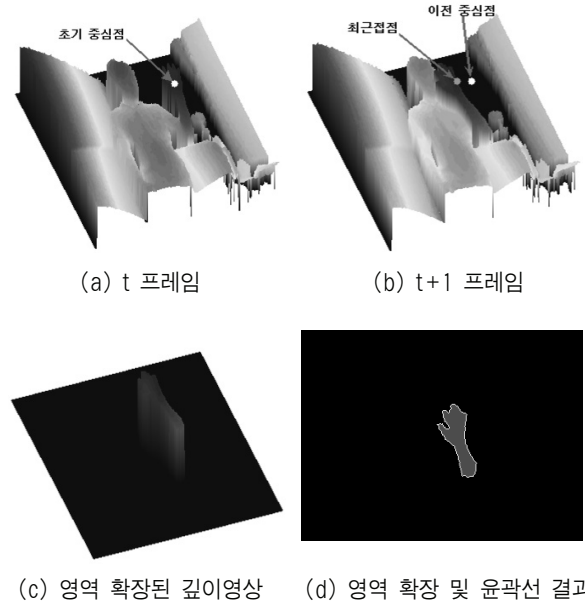


Fig. 8. Translation of center point

같이 계산한다. [수식 17]에서  $T_f$ 는 상수로서 카메라로부터 근접한 거리에 있는 점에 가중치를 주는 역할을 한다. 또한  $T_{dc}$ 는  $z$ 축의 단위와  $x$ 축,  $y$ 축에서의 단위가 서로 다르기 때문에 이를 보정해주는 역할을 한다. 이 값들은 실험을 통해 경험적으로  $T_f = 50$ ,  $T_{dc} = 4$ 로 적용하였다.

$$TP_t^{secd} = \operatorname{argmin} d(p, TP_t^{pre}) \quad (15)$$

$$d(a,b) = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2 + zd(a,b)} \quad (16)$$

$$zd(a,b) = \left( \frac{|a_z - (b_z - T_f)|}{T_{dc}} \right)^2 \quad (17)$$

#### 4.2 깊이기반 적응적 평균 이동(DAM-Shift)

중심점 전이 단계에서 이전 중심점으로부터 최근접점을 검출하고 영역을 확장하여 손 영역을 검출하였다. 안정적인 추적을 위해서는 특정한 조건을 만족하는 하나의 점을 추적해야 한다. 따라서 본 논문에서는 이러한 점의 기준을 DAM-Shift 를 이용하여 윤곽선의 무게중심이 수렴하는 점을 추적점으로 정의한다.

$$TP_t^{Mean} = \frac{\sum_{p \in \Omega} p \cdot K(p, TP_t^{secd}, TD_{t-1}^{Mean})}{\sum_{p \in \Omega} K(p, TP_t^{secd}, TD_{t-1}^{Mean})} \quad (18)$$

$$K(p, s, d) = \begin{cases} 1 & \|p - s\| < \alpha_1 + \alpha_2 d + \alpha_3 d^2 \\ 0 & otherwise \end{cases} \quad (19)$$

[수식 18]은 DAM-Shift 방법을 나타낸다. 여기서  $K$ 는 커널함수이며,  $TD_{t-1}^{mean}$ 은 적응적으로 커널의 크기를 변화시키는 파라미터로 이전( $t-1$ ) 프레임에서 추적점 위치의 깊이 값이다. 또한  $p$ 는 윤곽선 집합  $\Omega$ 에 속하는 점이다. 초기 시작 위치는 이전 절에서 구한 최근접점  $TP_t^{seed}$ 로부터 평균이동을 시작하며  $TP_t^{seed}$ 를 새로 구해진  $TP_t^{Mean}$ 으로 교체하여 수렴할 때까지 반복한다. 커널함수는 깊이 값에 따라서 적응적으로 반경이 변화한다. [수식 19]의  $\alpha_1, \alpha_2, \alpha_3$ 는 3.2.2절에서 설명한 2차 선형 모델과 동일한 방법으로 구한다. 먼저 다양한 거리에서 모든 손가락이 퍼진 상태의 손 영역이 포함된 영상을 획득하고, 수동으로 손바닥의 중심을 선정한다. 선정된 중심으로부터 가장 멀리 떨어진 중지의 끝 점까지 거리와 손바닥 중심의 깊이 값을 수집한다. 수집한 깊이 값을 [수식 12]의  $x$ 에 대입하고, 손바닥 중심으로부터 가장 멀리 떨어진 중지까지의 거리를  $r$ 에 대입하여 [수식 11]을 계산하면  $\alpha_1, \alpha_2, \alpha_3$ 를 구할 수 있다.

Fig. 9는 DAM-Shift를 이용하여 추적점 결정 과정을 보여준다. 단순한 DAM-Shift만으로는 원활한 추적 프로세스가 동작되지 않는다. 그 이유는 DAM-Shift로 수렴한 위치가 항상 손 영역( $\Psi$ ) 내부에 존재함을 보장하지 못하기 때문이다. 따라서 이를 보완하기 위해 스택 구조를 사용한다. Fig. 9의 (a)는 #1을 초기위치로 정하고 #1위치의 깊이 값을 대입한 커널함수로 DAM-Shift를 수행한 결과이다. 최종 수렴 위치(#9)를 보면 손 영역 외부에 위치함을 알 수 있다. 따라서 다음 프레임의 중심점 전이 과정에서 오류를 발생시킨다.

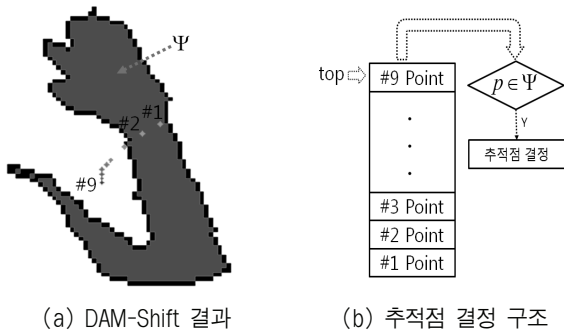


Fig. 9. Decision process for tracking point

이를 해결하기 위해 DAM-Shift의 각 반복 과정마다 획득된 좌표를 (b)와 같이 스택구조의 메모리 공간에 저장한다. 그리고 DAM-Shift가 수렴하게 되면 top위치부터 pop을 수행하여 해당 좌표가 손 영역 내부에 있는지 검사를 수행한다. 마지막으로 검사를 가장 처음으로 통과한 좌표를 최종 추적점으로 결정한다.

5. 실험결과

본 연구에서는 입력장치로 MS사의 Kinect를 사용하고 Kinect for Windows SDK를 이용하여 획득한 320 \* 240 크

기의 깊이영상을 입력으로 Intel(R) Core(TM) Quad CPU 2.66Ghz와 3GB 메모리에서 손 영역 검출 및 추적에 대하여 실험하였다.

Table 1. Operation time (단위 : ms)

	검출단계	추적단계
연산속도	14.85	2.96

Table 1은 검출 프로세스와 추적 프로세스의 평균 연산 속도를 나타낸다. 10번에 걸쳐 약 3분 동안 비정형화된 동작으로 검출과 추적의 프로세스가 동작하도록 실험을 하여 평균속도를 측정한 결과이다. 검출 프로세스의 속도는 움직임이 많을 경우에는 상대적으로 속도가 느렸으며, 움직임이 없는 경우에는 약 4.5ms 정도의 속도를 보였다. 또한 추적 프로세스는 알고리즘 특성상 상당히 연산량이 적으므로 매우 빠른 속도를 보였다.

Table 2. The time of hand detection in several distance

(단위 : sec)

거리	OpenNI	제안한 방법
1m	1.35	1.32
1.5m	1.25	1.27
2m	2.13	2.02

Table 2은 PrimeSense사의 OpenNI 라이브러리에서 제공하는 Sample-PointViewer 프로그램과 제안한 방법과의 검출속도 비교한 결과를 보여준다. OpenNI 라이브러리의 경우 손 검출 시 특정 자세에서만 적용된다. 다양한 형태로 각 거리에 따라 50회씩 손을 올리는 동작을 통해 손 영역 검출 여부와 검출시간을 측정하였다. 제안한 방법의 속도가 OpenNI 라이브러리를 이용한 경우보다 비교적 빠른 것을 확인할 수 있다. 이는 OpenNI의 경우 정해진 자세가 존재하므로 손바닥의 위치가 팔꿈치보다 위에 있는 경우에는 잘 검출하였으나, 그 외의 경우에는 검출하지 못하는 문제점이 있기 때문이다. 반면, 제안한 방법은 회전 불변의 특성을 갖기 때문에 정자세가 아니더라도 검출에 성공하여 평균적으로 좀 더 빠른 성능을 보였다.

Fig. 10은 1m의 거리에서 수행한 제안한 방법의 실험 중 하나의 예를 보여준다. 그림과 같이 17프레임에서 움직임을 시작하여 손을 올리는 동작이 마무리되는 35프레임에서 손 영역을 검출하였음을 알 수 있다. 25프레임에서 영역 확장 결과가 없는 경우는 상단의 누적차영상 영역 중 가장 근접한 화소를 기준으로 영역을 확장을 수행하였을 때 획득한 영역의 크기(화소의 수)가 너무 작아서 검출에 실패한 경우이다. 그림과 같이 1m의 거리에서는 매우 빠른 속도로 검출함을 알 수 있다.

Fig. 11은 손 영역 검출 후 추적 프로세스에서 알파벳 ‘a’, ‘b’를 추적한 궤적을 2차원 평면에 표현한 그림이다. 그림의 좌측 (a), (d)는 느리게 동작한 경우를 보여주며, 중간인 (b), (e)는 보통 속도로 수행한 경우, 마지막으로 우측 (c), (f)는

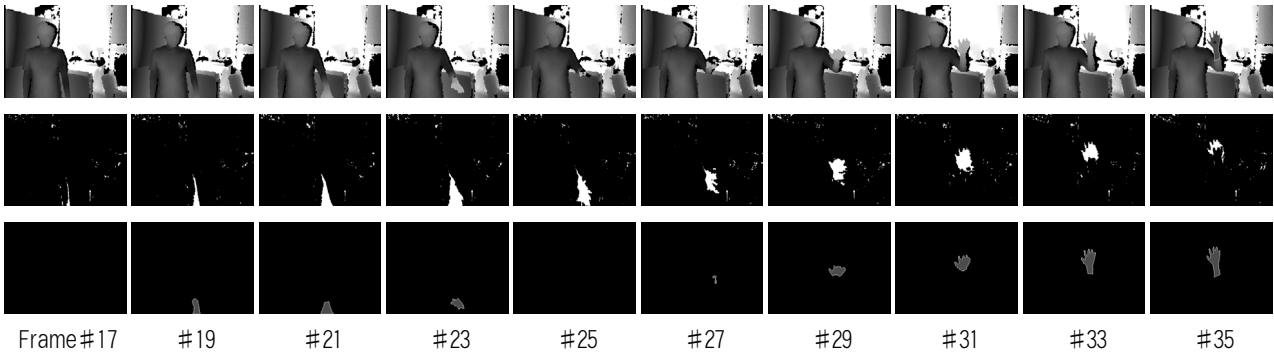


Fig. 10. The result of hand region detection  
(top : depth image, middle : accumulated difference image, bottom : region growing image)

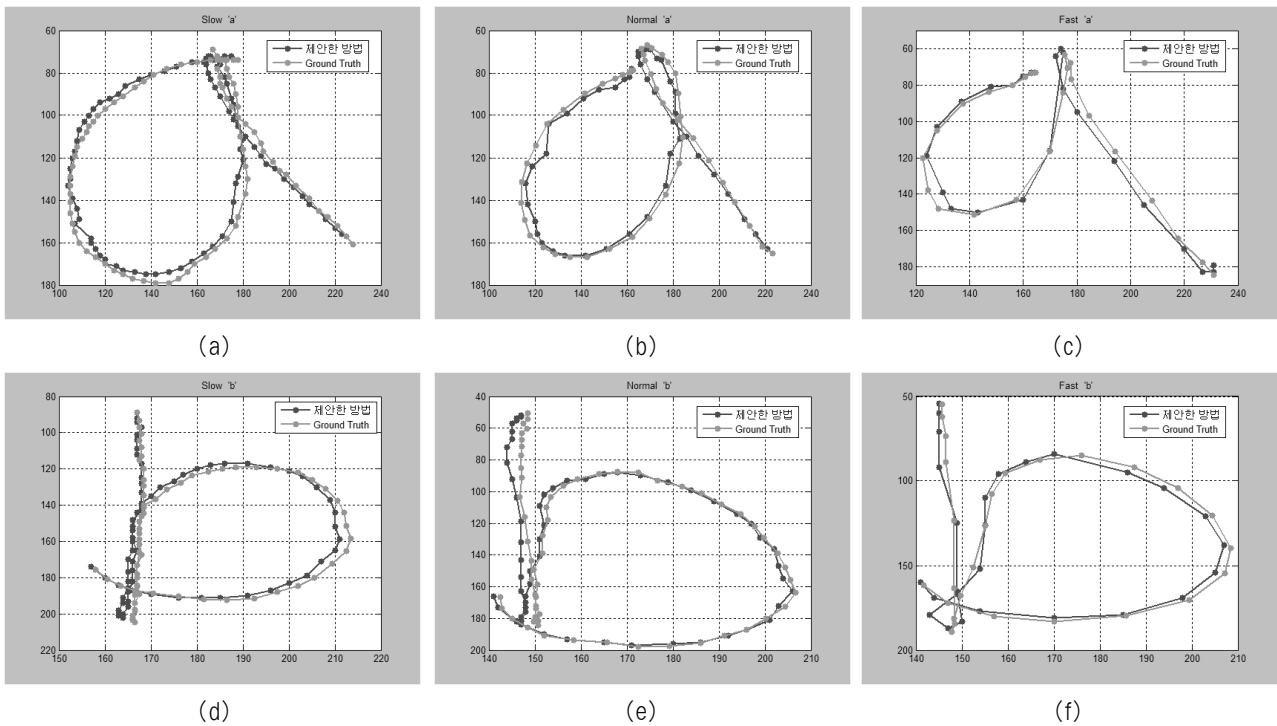


Fig. 11. The tracking result of gesture 'a' and 'b' with various velocity condition

빠르게 수행한 경우를 나타낸다. 적색 궤적은 제안한 방법으로 추적한 궤적을 의미하며, 녹색 궤적은 수동으로 손의 중심점을 획득한 실측 궤적(ground-truth)이다. 대부분의 경우에서 추적된 궤적이 실측 궤적에 대해 형태적으로 큰 오차가 없음을 확인할 수 있다.

Table 3는 제스처 동작 'a'와 'b'를 속도 별로 각 10번씩

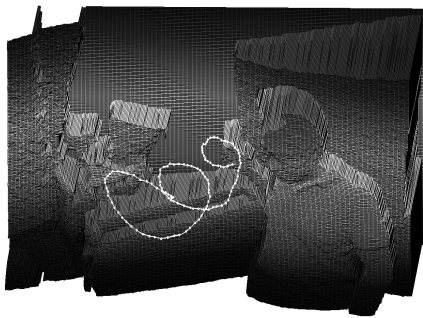
Table 3. The average error of tracking with various velocity condition (단위 : pixels)

	Slow	Normal	Fast
'a'	3.7559	4.1664	8.4290
'b'	2.3446	3.4423	4.3796

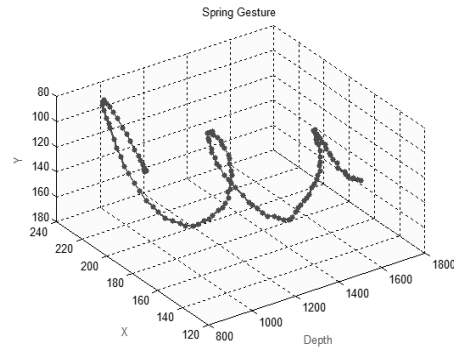
수행하여 실측 궤적과 비교한 평균 궤적 오차이다. 대부분 5화소 이하의 오차가 발생하였다. 'a'를 빠르게 한 경우에 다른 동작에 비해 큰 오차가 발생한 이유는 문자 특성상 마지막 획을 상당히 빠른 속도로 그리면서 손 영역의 깊이영상이 뭉개지는 현상이 발생하여 손의 중심점을 부정확하게 추적하였기 때문이다.

Fig. 12은 나선형 제스처 추적 결과를 나타낸다. (a)는 나선형 제스처가 완료되는 순간의 깊이영상과 나선형 궤적을 동시에 3차원 공간에서 표현한 것으로서 백색의 궤적이 이전 프레임동안 추적된 나선형 제스처의 궤적을 나타내며, (b)는 나선형 궤적만을 별도로 나타낸 것이다. 결과에서 보듯이  $x$ ,  $y$ 축뿐만 아니라  $z$ 축으로의 변화에도 불구하고 손 영역 추적이 강건하게 수행되는 것을 확인할 수 있다.





(a) 깊이영상과 나선형 궤적



(b) 나선형 궤적

Fig. 12. The tracking result of 'Spring' gesture

### 6. 결 론

본 논문에서는 깊이영상만을 이용하여 최소한의 환경적 제약 조건으로 손 영역을 검출하고, 손 모양, 조명 및 색상 정보의 변화에 강건한 실시간 손 영역 추적 방법에 대해 제안하였다. 움직임 정보에 기반한 누적차영상과 손 영역의 형태모델을 이용한 손 검증 방법을 통해 손 영역을 검출함으로써, 기존의 컬러 영상 정보를 이용하는 방법과 공개된 라이브러리인 OpenNI를 이용하는 것보다 빠르고 정확한 손 검출이 가능함을 확인하였다. 또한, 제스처의 궤적을 분석하기 위한 손 영역의 중심점 추적을 위해 깊이정보를 고려한 최근접점 검출 및 DAM-Shift를 이용한 추적 방법은 깊이 영상만으로 효율적인 제스처 인터페이스 기술로의 확장이 가능함을 입증하였다. 또한 깊이영상만을 이용하기 때문에 센서에 독립적으로 여러 분야에 응용될 수 있다. 본 논문에서는 제스처 인식을 위한 선행연구로 검출 및 추적을 수행하였으므로 향후 추적된 궤적을 이용한 제스처 인식에 대한 연구가 추가적으로 필요할 것이다.

### 참 고 문 헌

[1] V. Pavlovic, R. Sharma, and T. Huang, "Visual Interpretation of Hand Gestures for Human-Computer Interaction: A Review", *IEEE Trans. on PAMI*, Vol.19, No.7, pp.677-695, 1997.

[2] C. Manresa, J. Varona, R. Mas, Francisco, and J. Perales, "Hand Tracking and Gesture Recognition for Human-Computer Interaction", *Electronic Letters on Computer Vision and Image Analysis*, Vol.5, No.3, pp.96-104, 2005.

[3] M. G. Hwang, H. R. Kim, S. B. Kang, and T. K. Ynag, "Vision Based Real-time Hand Shape Recognition Using Fuzzy Inference", *Journal of Korean Institute of Information Technology*, Vol.6, No.2, pp.53-59, 2008.

[4] H. I. Suk, and B. H. Sin, "Dynamic Bayesian Network based Two-Hand Gesture Recognition", *Journal of KIISE : Software*

and Applications, Vol.35, No.4, 2008.

[5] A. Yilmaz, O. Javed, and M. Shah, "Object tracing: A survey", *ACM Comput. Surv.*, Vol.38, No.4, Dec., 2006.

[6] B. Stenger, "Template-Based Hand Pose Recognition Using Multiple Cues", *ACCV 2006, LNCS*. 3852, pp.551-560, 2006.

[7] P. Breuer, C. Eckes, and S. Muller, "Hand Gesture Recognition with a novel IR Time-of-Flight Range Camera-A pilot study", *LNCS*.4418, 247, 2007.

[8] I. Oikonomidis, N. Kyriazis, and A. A. Argyros, "Efficient model-based 3D tracking of hand articulations using Kinect", in *British Machine Vision Conference*, Dundee, UK, pp.101.1-101.11, 2011.

[9] S. H. Park, S. J. Yu, J. R. Kim, S. J. Kim, and S. Y. Lee, "3D hand tracking using Kalman filter in depth space", *EURASIP Journal on Advances in Signal Processing*, Vol.2012, No.1, pp.1-18, 2012.

[10] M. B. Holte, T. B. Moeslund, and P. Fihl, "Fusion of range and intensity information for view invariant gesture recognition", in *IEEE Computer Society Conference on Computer Vision & Pattern Recognition Workshops*, Anchorage, AK, U.S.A, pp.1-7, 2008.

[11] H. S. Yang, and H. W. Jung, "A study on hand recognition in image for multimedia system", *The Journal of the Korea Contents Association*, Vol.5, No.2, pp.267-274, 2005.

[12] PrimeSensor <http://www.primesense.com>

[13] H. Brey, J. Gil, D. Kirkpatrick, M. Werman, "Linear Time Euclidean Distance Transform Algorithms", *Pattern Analysis and Machine Intelligence, IEEE Transactions*, Vol.17, No.5, pp.529-533, 1995.

[14] [http://www.imageprocessingplace.com/downloads\\_V3/root\\_downloads/tutorials/contour\\_tracing\\_Abeer\\_George\\_Ghuneim/index.html](http://www.imageprocessingplace.com/downloads_V3/root_downloads/tutorials/contour_tracing_Abeer_George_Ghuneim/index.html)

[15] C. P. Chen, Y. T. Chen, P. H. Lee, Y. P. Tsai, and S. Lei, "Real-time Hand Tracking on Depth Images", *Visual Communications and Image Processing (VCIP)*, IEEE, pp.1-4, 2011.



**주 성 일**

e-mail : sijoo82@ssu.ac.kr  
2008년 한국산업기술대학교(공학사)  
2010년 숭실대학교 미디어학과(공학석사)  
2010년~현 재 숭실대학교 미디어학과  
박사과정  
관심분야: 영상처리, 컴퓨터비전, 패턴  
인식 등



**원 선 희**

e-mail : nifty12@ssu.ac.kr  
2005년 한경대학교 컴퓨터공학과(공학사)  
2007년 숭실대학교 컴퓨터학과(공학석사)  
2012년 숭실대학교 미디어학과(공학박사)  
2012년~현 재 숭실대학교 미디어학과  
Post Doc  
관심분야: 영상처리, 컴퓨터비전, 3D 모델링, 패턴인식 등



**최 형 일**

e-mail : hic@ssu.ac.kr  
1972년 연세대학교 전자공학과(공학사)  
1982년 미시간대학교 전자공학과  
(공학석사)  
1987년 미시간대학교 전자공학과  
(공학박사)  
1995년~1997년 퍼지 및 지능시스템학회 이사  
1996년~1998년 정보과학회 컴퓨터비전 및 패턴인식 연구회  
위원장  
1997년 IBM Waston Lab 방문연구원  
2005년~2006년 한국정보과학회 이사  
1987년~현 재 숭실대학교 미디어학과 교수  
관심분야: 컴퓨터비전, 퍼지 및 신경망 이론, 패턴인식, 지식기반  
시스템 등