



깊이 영상 기반 손 영역 추적 및 손 끝점 검출

Hand Region Tracking and Fingertip Detection based on Depth Image

저자 (Authors)	주성일, 원선희, 최형일 Sung-Il Joo, Sun-Hee Weon, Hyung-Il Choi
출처 (Source)	한국컴퓨터정보학회논문지 18(8) , 2013.8, 65-75 (11 pages) Journal of the Korea Society of Computer and Information 18(8) , 2013.8, 65-75 (11 pages)
발행처 (Publisher)	한국컴퓨터정보학회 The Korean Society Of Computer And Information
URL	http://www.dbpia.co.kr/Article/NODE06529004
APA Style	주성일, 원선희, 최형일 (2013). 깊이 영상 기반 손 영역 추적 및 손 끝점 검출. 한국컴퓨터정보학회논문지, 18(8), 65-75.
이용정보 (Accessed)	광운대학교 128.134.57.*** 2018/05/08 03:00 (KST)

저작권 안내

DBpia에서 제공되는 모든 저작물의 저작권은 원저작자에게 있으며, 누리미디어는 각 저작물의 내용을 보증하거나 책임을 지지 않습니다. 그리고 DBpia에서 제공되는 저작물은 DBpia와 구독계약을 체결한 기관소속 이용자 혹은 해당 저작물의 개별 구매자가 비영리적으로만 이용할 수 있습니다. 그러므로 이에 위반하여 DBpia에서 제공되는 저작물을 복제, 전송 등의 방법으로 무단 이용하는 경우 관련 법령에 따라 민, 형사상의 책임을 질 수 있습니다.

Copyright Information

Copyright of all literary works provided by DBpia belongs to the copyright holder(s) and Nurimedia does not guarantee contents of the literary work or assume responsibility for the same. In addition, the literary works provided by DBpia may only be used by the users affiliated to the institutions which executed a subscription agreement with DBpia or the individual purchasers of the literary work(s) for non-commercial purposes. Therefore, any person who illegally uses the literary works provided by DBpia by means of reproduction or transmission shall assume civil and criminal responsibility according to applicable laws and regulations.

깊이 영상 기반 손 영역 추적 및 손 끝점 검출

주성일*, 원선희*, 최형일*

Hand Region Tracking and Fingertip Detection based on Depth Image

Sung-Il Joo*, Sun-Hee Weon*, Hyung-Il Choi*

요약

본 논문에서는 깊이 영상만을 이용하여 손 영역 추적 및 손 끝점 검출 방법을 제안한다. 조명 조건의 영향을 제거하고 빠르고 안정적인 정보 획득을 위해 깊이 정보만을 이용하는 추적 방법을 제안하고, 영역 확장 방법을 통해 추적 과정 중에 발생할 수 있는 오류에 대한 판단 방법과 다양한 제스처 인식에 응용이 가능한 손 끝점 검출 방법을 제안한다. 먼저 추적점을 찾기 위해 중심점 전이 과정을 통해 최근접점을 찾고 그 점으로부터 영역 확장을 통해 손 영역과 경계선을 검출한다. 그리고 영역 확장을 통해 획득한 무효 경계선의 비율을 이용하여 추적 영역에 대한 신뢰도를 계산함으로써 정상 추적 여부를 판단한다. 정상적인 추적인 경우, 검출된 손 영역으로부터 윤곽선을 추출하고 곡률 및 RANSAC, 컨벡스 헐(Convex-Hull)을 이용하여 손 끝점을 검출한다. 마지막으로 성능 검증을 위해 다양한 상황에 따른 정량적, 정성적 분석을 통해 제안하는 추적 및 손 끝점 검출 알고리즘의 효율성을 입증한다.

▶ Keywords : 손 영역 검출, 깊이 추적, 손 끝점 검출

Abstract

This paper proposes a method of tracking the hand region and detecting the fingertip using only depth images. In order to eliminate the influence of lighting conditions and obtain information quickly and stably, this paper proposes a tracking method that relies only on depth information, as well as a method of using region growing to identify errors that can occur during the tracking process and a method of detecting the fingertip that can be applied for the recognition of various gestures. First, the closest point of approach is identified through the process of transferring the center point in order to locate the tracking point, and the region is grown from that point to detect

•제1저자 : 주성일 •교신저자 : 최형일

•투고일 : 2013. 7. 30, 심사일 : 2013. 8. 7, 게재확정일 : 2013. 8. 19.

* 숭실대학교 글로벌미디어학과(Dept. of Global Media, Soongsil University)

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

the hand region and boundary line. Next, the ratio of the invalid boundary, obtained by means of region growing, is used to calculate the validity of the tracking region and thereby judge whether the tracking is normal. If tracking is normal, the contour line is extracted from the detected hand region and the curvature and RANSAC and Convex-Hull are used to detect the fingertip. Lastly, quantitative and qualitative analyses are performed to verify the performance in various situations and prove the efficiency of the proposed algorithm for tracking and detecting the fingertip.

▶ Keywords : Hand detection, Depth Tracking, Fingertip detection

I. 서 론

최근 스마트폰, 스마트 TV등의 스마트 기기의 급속한 발전과 보급화로 인해 사용자 인터랙션에 대한 사용자들의 관심이 날로 급증하고 있다. 이러한 대중들의 관심을 반영하기 위해 가장 집중적으로 연구되고 있는 분야가 바로 지능형 사용자 인터페이스 분야이다. 사용자 인터페이스는 기술적인 부분도 중요하지만, 무엇보다 사용자가 컴퓨터와 대화하기 편리해야 한다. 이러한 사용자의 편리성을 위해 사용자 관점의 여러 가지 조건이 요구된다. 먼저 어떤 의도와 그 의도에 대한 동작이 자연스러운 방식으로 연결되어 있어야 하는 의도적합성(Intention Compatibility)과 동작이 자연스럽게 유발되는 지에 대한 동작 유발성(Affordance), 또한 사용자가 기능 동작을 위한 명령은 예측 가능하여야 하며 사용자의 신체 부하를 최소화시켜야 한다. 마지막으로 사용자의 의도를 정확히 수행하기 위해 기능들이 의도에 따라 정확하게 수행되어야 한다.

이러한 요구로 인하여 제스처 관련 분야도 손 영역의 제스처 뿐만 아니라 다른 정보들을 이용하기 위한 연구들이 많이 진행되고 있으며, 최근 저렴한 가격의 깊이 카메라(예, 키넥트)가 보급되면서 깊이 영상과 관련된 제스처 인식 분야의 연구도 활발히 진행되고 있다. 제스처 인식을 위한 필수적인 과정은 크게 두 단계로 분류될 수 있다. 먼저 제스처 인식을 위한 손 영역 검출 과정이다. 손의 제스처를 이용하여 인식하는 경우 손 영역의 검출은 필수적이다. 그리고 제스처는 시간정보에 따른 이동 좌표의 나열이므로 추적과정 또한 필수적으로 필요하다. 또한 제스처 인식의 종류는 크게 정적 제스처(Static Pose), 동적 제스처(Dynamic Gesture)로 분류할 수 있다. 정적 제스처의 경우에는 손의 모양, 형태 정보를 이용하여 사용자와 인터랙션을 하는 방법이며, 정적 제스처는 손의 움직임에 따른 제스처 정보의 조합을 통해 인터랙션하는 방법이다.

이를 위한 연구 분야로는 크게 3가지로 구분할 수 있다. 먼저 색상정보를 이용한 연구[1,2,3] 주를 이루고 있으며, 최근 들어 색상과 깊이정보를 결합한 연구와[4,5], 깊이정보만을 이용한 연구가[6,7] 활발히 진행되고 있다.

II. 관련 연구

손 끝점 검출을 위한 연구로 색상 정보만을 이용한 Daeho Lee[1]는 피부색과 손의 움직임 정보를 이용하여 손 영역을 찾고 경계선 검출 후 k-cosines 방법을 통해 손 끝점을 검출하였다. Aditya Ramamoorthy[2]는 손 영역의 형태를 Contour Discriminant Analysis를 통해 인식하고 칼만필터로 추적한다. 그리고 은닉 마르코프 모델을 사용하여 인식을 수행한다.

Van den Bergh[4]는 RGB 영상에서 얼굴을 검출하고, 검출한 얼굴의 거리 값을 임계값으로 적용하여 배경을 제거한 뒤, 나머지 영역에서 피부색으로 손 영역을 검출하는 혼합된 형태를 제안하였다. Trindade[5]는 RGB-D 센서로부터 RGB 색상에 의한 피부 색상 필터링을 먼저 수행하여 몸체 부분과 얼굴, 손 영역 부분을 검출하고, 깊이축(depth axis)에 따라 히스토그램을 배치한 후 임계값으로 필터링한다. 이후 k-평균 클러스터를 이용하여 이상치(outlier)를 제거함으로써 손 영역의 중심점을 추정하며, 추정된 손 중심점을 기반으로 최종 손 영역을 검출한 후 포즈 인식을 수행한다. 이 방법은 손 영역 검출을 위해 색상정보와 깊이정보를 융합한 형태로 필터링 과정의 이상치 제거와 클러스터링 기법을 적용함으로써 검출 단계의 정확성을 높였다.

Poonam Suryanarayan[6]는 깊이정보를 이용하여 손 포즈 인식을 위해 2차원 형태정보와 압축된 3차원 형태 디스크립터, 3차원 볼륨매트릭 형태 디스크립터를 제안하였다. 3차원 형태를 표현하는 디스크립터는 매우 좋은 방법이나, 일

반적인 깊이 영상의 경우 카메라로부터 보이는 부분만 깊이 값을 가지고 있으므로 이에 대한 처리의 어려움이 있다. Xia Liu[7]은 카메라로부터 일정 거리 안에 있는 객체를 사람이라 가정하고, 그 객체만 가로와 세로로 프로젝션하여 얼굴을 검출한다. 손동작 인식을 위해서는 일반적으로 손은 몸체와 떨어져있다고 가정하고 팔 부분과 손 부분을 분리하기 위해 비율적인 상수를 사용하여 손 영역만 추출한다. 그리고 제스처 인식을 위해 손 형태 데이터와 챔퍼 거리(Chamfer Distance)를 사용하여 유사도를 계산하여 인식한다. 그리고 손 영역의 궤적 정보를 구하고, FFT를 사용하여 잡음을 제거 후 사전에 정의한 제스처 이동 궤적 데이터와 최소 제곱 에러를 계산하여 제스처 인식을 수행한다. 이는 사전에 미리 제스처를 관련 데이터를 제공해야 하며, 유사도 측정 방법에 따라 성능이 달라진다.

이러한 다양한 연구는 장단점을 가지고 있다. 먼저 색상 기반의 연구는 조명에 민감하여 조명이 없는 어두운 실내에서는 수행할 수 없으며, 깊이와 컬러가 융합된 연구는 색상 기반 연구와 같은 문제점 외에 연산량이 색상 기반보다 많다는 문제점이 있다. 따라서 본 논문에서는 깊이 영상만을 사용하여 손을 추적하고, 손 영역만을 분리하여 다양한 동적 제스처 및 정적 제스처 인식을 위한 손 궤적 및 손 끝점 검출 방법에 대해서 제안한다. [그림 1]은 시스템 흐름도를 보여준다. 먼저 깊이

영상이 입력되면, 주어진 깊이 영상으로부터 손 영역을 검출하고, 손 영역 추적 단계로 넘어간다. 추적 단계에서는 중심점 전이, 영역 확장, DAM-Shift를 통해 추적을 수행하고, 신뢰도 계산 방법을 통해 예외 사항을 처리한다. 추적 단계가 완료되면, 추적된 영역으로부터 손 끝점을 검출 할 수 있다. 손 끝점 검출을 위해서 RANSAC[8] 알고리즘으로 손 영역의 주방향을 검출하고, 영역의 경계를 검출 후 곡률 분석을 통해 특징점을 검출한다. 마지막으로 검출된 특징점으로부터 Convex-Hull 알고리즘을 사용하여 손 끝점만을 검출한다.

본 논문의 구성은 다음과 같다. 3장에서 손 영역 추적의 세부 알고리즘에 대해서 설명하고, 4장에서 손 끝점 검출 방법에 대해 제안한다. 5장에서는 제안된 방법의 실험 결과를 보이고, 6장에서 결론을 맺는다.

III. 손 영역 추적

손 영역을 추적하기 위해서는 먼저 추적할 객체 영역, 즉 손 영역을 검출해야한다. Joo[9]에서 제안된 방법인 움직임 정보와 형태 정보를 사용하여 손 영역을 검출하였다. 추적 과정은 검출된 손 영역 위의 한 점의 입력으로부터 동작한다. 따라서 손 영역 검출로 인해 손 영역 위의 한점을 입력하였고 가정하고 추적 방법에 대해 설명한다. 전반적인 추적 알고리즘은 Joo의 방법과 유사하나 새로운 영역 확장 방법을 통한 개선된 추적 방법에 대해 제안한다.

1. 중심점 전이 및 영역 확장

손 영역이 검출되면 추적을 수행한다. 먼저 이전 시점에 검출된 손 영역의 3차원 좌표를 이용하여 3D 공간상의 가장 가까운 좌표를 선택한다. [수식 1]은 최근접점을 찾는 수식으로 TP_t^{pre} 는 이전 프레임의 추적점을 나타내며, p 는 비교하고자하는 임의의 점이다. [수식2]는 x 축과 y 축의 경우에는 유클리디안 거리를 사용하고 거리 정보인 z 축으로 차이는 [수식3]과 같이 계산한다. [수식3]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.

$$TP_t^{seed} = \arg \min d(p, TP_t^{pre}) \quad (1)$$

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

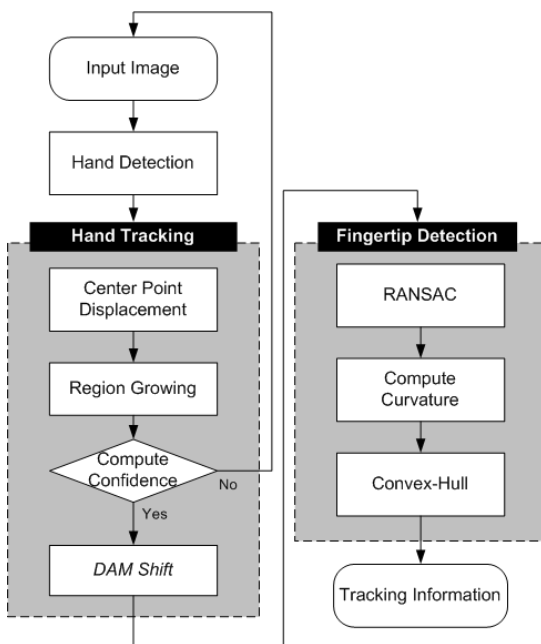


그림 1. 시스템 흐름도
Fig. 1. System flow chart

$$zd(a,b) = \left(\frac{|a_z - (b_z - T_f)|}{T_{dc}} \right)^2 \quad (3)$$

이전 시점의 추적점으로부터 최근접점(TP_t^{seed})을 찾으면, 손 영역을 유추할 수 있다. 본 논문에서는 최근접점으로부터 손 영역을 검출하기 위해 영역 확장 방법을 사용한다. 먼저 영역 확장의 기준은 깊이 값이다. 즉 깊이 값이 유사하다면 영역 확장을 수행한다. 하지만 깊이 값이 조금씩 변화하는 객체의 경우 영역 확장을 수행하면 너무 많은 영역 확장이 일어날 수 있다. 따라서 이러한 문제점을 막기 위해 전역 임계값을 함께 사용한다.

$$Cond_{all} = \begin{cases} \sum_i^n cond_i \geq n & True \\ otherwise & False \end{cases} \quad (4)$$

$$cond_1 = \begin{cases} |D_t(n,m) - D_{t-1}(c,r)| < Th_D & 1 \\ otherwise & 0 \end{cases} \quad (5)$$

$$cond_2 = \begin{cases} D_t(n,m) - C_{dep} < Th_{GD} & 1 \\ otherwise & 0 \end{cases} \quad (6)$$

$$cond_3 = \begin{cases} Grow_{cnt} < Th_c & 1 \\ otherwise & 0 \end{cases} \quad (7)$$

$$cond_4 = \begin{cases} Dist < Th_c & 1 \\ otherwise & 0 \end{cases} \quad (8)$$

[수식 4~8]은 영역 확장 여부를 결정하는 수식이다. 먼저, [수식 5]의 $D_t(n,m)$ 은 현재 확장 여부를 검사하기 위한 화소의 깊이 값이며, $D_{t-1}(c,r)$ 는 이전 시점에 확장된 기준 화소의 깊이 값이다. 즉 기준 좌표의 깊이 값을 기반으로 임계값(Th_D) 이하라면 조건을 만족할 것이다. 임계값 Th_D 는 객체의 연결성을 정의하기 위한 값으로 손 영역이라면 이웃한 화소의 깊이 값의 차이가 적고 이웃한 화소와의 깊이 차이가 크다면 배경일 가능성이 크기 때문이다. [수식 6]은 영역 확장의 범위를 제한하기 위한 조건이다. C_{dep} 은 TP_t^{seed} 의 깊이 값으로 전역 임계값(Th_{GD})을 사용하여 깊이 값 허용 범위를 제한하였다. 또한 [수식 7]은 영역의 크기를 제한하기 위해 확장 횟수($Grow_{cnt}$)를 누적하여 임계값(Th_c) 이하까지만 반복하도록 조절하며, [수식 8]의 $Dist$ 는 최근접점(TP_t^{seed})

과 확장하고자하는 화소의 유클리디언 거리를 의미하는 것으로 거리를 한정하는 역할을 수행한다. 결과적으로 [수식 4]에 의해 모든 조건(수식5~8)을 만족할 경우 참으로 결정한다.

[표 1]은 영역 확장 알고리즘을 보여준다. 먼저 입력은 [수식 1]로부터 검출한 최근접점(TP_t^{seed})와 깊이 임계값(Th_D), 전역 깊이 임계값(Th_{GD})이 입력된다. 입력된 기준 좌표인 TP_t^{seed} 의 깊이 값을 이용하여 사전에 생성한 2차 선형 모델[9]로 거리 및 영역 확장의 임계값 Th_c 를 적응적으로 획득한다. 그리고 좌표 배열 B 에 기준 좌표 TP_t^{seed} 를 추가하여 초기화를 수행한다. 결과적으로 B 로부터 좌표점들을 획득하여 주위 8방향에 대해 위에서 설명한 [수식 4] 조건을 검사하고 만족한 화소에 대해 반복적으로 영역 확장을 수행한다. 영역 확장의 완료 시점은 [수식 4]의 4가지 조건에 의해 상황에 따라서 적응적으로 변화한다. 또한 조건($Cond_{all}$)을 만족하지 못한 경우, 두 가지 경우에 따라 분리되어 저장된다. 만약 $cond_1$ 을 만족하지 못한 경우에는 유효경계선 집합(VB)에 추가되며, 나머지 경우에는 무효 경계선 집합(UVB)에 추가

표 1. 영역 확장 알고리즘
Table 1. The algorithm of region growing

```

• Input :  $TP_t^{seed}$ ,  $Th_d$ ,  $Th_{GD}$ 
• Initialize :  $B = \emptyset$ ,  $tmpB = \emptyset$ ,  $VB = \emptyset$ ,  $UVB = \emptyset$ 
•  $Th_c \leftarrow SecondOrderLinearModel(Dep_{TP})$ 
•  $B \leftarrow Add\ TP_t^{seed}$ 
• While  $B_{size} > 0$ 
  for  $i = 0, i < B_{size}$ 
     $c = B[i]_x$ ,  $r = B[i]_y$ 
    for  $m = r - 1, m \leq r + 1$ 
      for  $n = c - 1, n \leq c + 1$ 
        if ( $Cond_{all}$ )
           $tmpB \leftarrow Add\ Point(n, m)$ 
        else
          if ( $cond_1 \neq 1$ )
             $VB \leftarrow Add\ Point(n, m)$ 
          else
             $UVB \leftarrow Add\ Point(n, m)$ 
        endif
      end
    end
  end
   $Swap(B, tmpB)$ 
   $tmpB \leftarrow \emptyset$ 
   $Grow_{cnt} \leftarrow Grow_{cnt} + 1$ 
• Output :  $VB, UVB$ 

```

된다. 영역 확장의 결과로 획득한 경계선 집합은 이 후에 DAM-Shift 알고리즘에 사용된다.

[그림 2]는 [표 1]의 알고리즘 구현 결과이다. 설명한 수식 4 조건을 검사하고 만족한 화소에 대해 반복적으로 영역 확장한 결과를 보여준다. 그림의 예에서는 입력된 TP_t^{seed} 의 깊이 값으로 계산한 결과 Th_c 는 48이었다. 따라서 48번 반복하여 영역 확장이 완료되었다. (f)를 보면 경계선 색상이 녹색과 노란색을 볼 수 있는데, 이는 $Cond_{all}$ 을 통과하지 못하였을 경우에 해당하는 좌표로서 녹색은 위의 알고리즘에서 설명한 유효경계선 VB 를 의미하며, 노란색은 무효경계선 UVB 를 나타낸다.

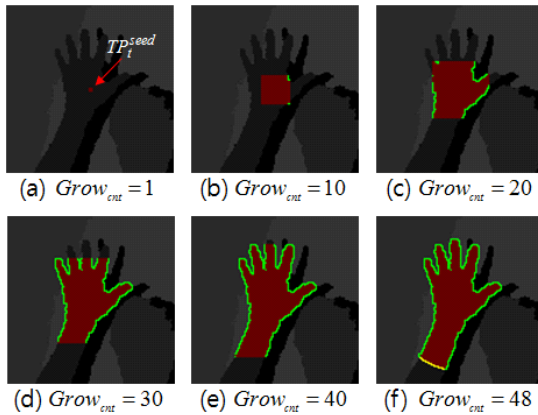


그림 2. 영역 확장 결과
Fig. 2. The result of region growing

2. 무효 추적 결정 방법

추적 과정은 항상 정상적으로 수행될 수 있다고 가정할 수 없다. 예상치 못한 문제로 인해 추적을 실패할 수 있기 때문에 시스템은 추적 성공 여부를 판단할 수 있어야 한다. 추적 성공 여부는 중심점 전이 과정과 영역 확장이 완료되면 판단할 수 있기 때문에 먼저 설명한다.

많은 실험을 수행한 결과, 손을 매우 빠르게 움직이거나, 얼굴 뒤로 손이 움직이는 경우, 손이 다른 물체에 붙은 경우 추적점이 다른 객체로 이동하는 오작동을 보였다. 이러한 문제점을 해결하고자, 무효 추적 결정 방법을 제안한다.

[그림 3]의 (a)~(c)의 경우와 같이 일반적으로 손 영역이 정상적으로 검출된 경우에는 무효경계선(노란선)의 길이가 매우 짧은 것을 알 수 있다. 반대로 (d)와 (e)의 경우와 같이 손 영역이 다른 영역과 겹친 경우와 (f)처럼 손 영역 추적을 실패하여 얼굴로 전이된 경우, 정상적인 손 영역의

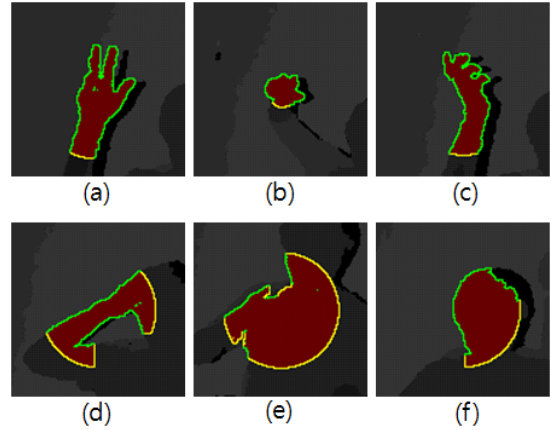


그림 3. 영역 확장과 경계선 검출 결과
Fig. 3. The result of region growing and boundary detection

무효경계선보다 상대적으로 많은 무효경계선이 포함됨을 확인할 수 있다. 따라서 이러한 특성을 이용하여 현재 추적중인 손 영역이 정상적으로 추적되고 있는지 판단한다.

[수식 9]는 원의 둘레 공식을 이용한 신뢰도 임계값 계산 수식이다. Th_c 는 [수식 7]과 [수식 8]에서 사용된 임계값으로 최근점점의 깊이 값에 따라 적응적인 값을 갖는다. 0.35는 경험적으로 결정하였다. 이렇게 영역 확장 시 경계선 검출까지 같이 수행하고, 유효경계선과 무효경계선을 구함으로써 추적의 성공 여부를 판단할 수 있다. 최종적으로 무효경계선에 포함되는 화소의 수가 $Th_{confidence}$ 보다 적으면 정상적인 추적으로 판단하며 반대의 경우에는 비정상적인 추적으로 판단한다.

$$Th_{confidence} = 2\pi \times Th_c \times 0.35 \quad (9)$$

3. DAM-Shift를 이용한 추적

이전 단계에서 중심점 전이 과정과 영역 확장을 통하여 최근점점을 정하고, 추적되는 손 영역을 검출하였다. 또한 무효 추적 결정 방법을 통해 추적의 지속 여부를 결정하였다. 하지만, 중심점 전이의 경우에는 항상 유사한 위치의 점을 검출할 수 없다. 추적을 위해서는 일반적으로 항상 같은 기준의 점이 검출되어야만 한다. 이를 위한 방법으로 DAM-Shift를 사용한다. [9] [그림 2]의 (f)를 보면 손 영역 중 팔보다 손바닥 근방에 많은 경계가 존재함을 알 수 있다. 이러한 특성을 이용하여 DAM-Shift로 최종 추적점을 결정한다.

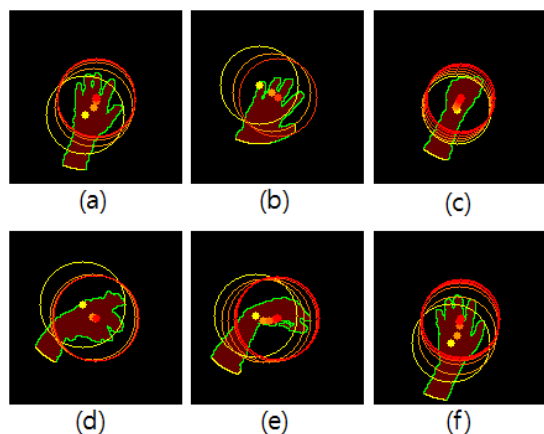


그림 4. DAM-Shift 결과
Fig. 4. The result of DAM-Shift

DAM-Shift는 평균이동기법(Mean-Shift)과 유사하나 깊이 값에 따라 커널 크기를 적응적으로 변화시켜 개선한 알고리즘이다. DAM-Shift를 수행하기 위해서는 초기 위치와 조건이 필요하다. 초기 위치는 이전 단계에서 구한 최근접점이 되며, 입력 경계선은 영역 확장 과정에서 획득한 유효경계선만을 이용한다. 또한 커널 크기는 사진에 깊이에 따른 커널 크기를 2차 선형 모델로 학습하여 사용한다. [그림 4]는 DAM-Shift 수행 결과를 보여준다. 초기 위치는 노란점이며, DAM-Shift를 통해 이동된 최종점은 적색점이다. 또한 원은 커널의 크기를 보여준다. 다양한 손 영역의 형태에서도 모두 손바닥 내부에 추적점이 최종 결정된 것을 볼 수 있다.

IV. 손 끝점 검출

3장에서 손 영역 추적에 대해 설명하였다. 손 영역 추적을 통해 얻은 추적점만으로도 손동작 인식이 가능하다. 하지만 좀 더 다양하고 사용자 편의를 위한 손동작 인식을 위해서는 좀 더 많은 정보가 필요하다. 이러한 이유로 본 장에서는 손 끝점을 검출하는 방법에 대해 설명하고자 한다.

손 끝점 검출 방법은 크게 2단계로 구성된다. 먼저 RANSAC을 이용하여 주축을 찾는다. 이는 이후 단계인 Convex-Hull을 정상적으로 만들기 위해 임의의 한 점을 추가하기 위함이다. 마지막으로 3.1절에서 획득한 손 영역으로 윤곽선을 검출하여 Peak점과 Valley점을 검출하고 최종적으로 Convex-Hull을 통해 손 끝점만 검출한다.

1. RANSAC을 이용한 주축 검출

3.1절에서 경계선뿐만 아니라 손 영역도 검출하였다. 검출한 손 영역을 입력으로 RANSAC을 통하여 주축을 찾는다. [8] RANSAC은 잡음이 많은 데이터를 이용하여 모델링하는 응용에 자주 사용된다. 간단하게 설명하자면 검출된 손 영역에 속하는 화소들 중 랜덤하게 2개의 점을 선택한다. 선택된 두 점으로 직선의 방정식을 구하고, 구해진 직선의 방정식으로 모든 손 영역 화소들과의 거리를 계산한다. 거리가 일정 임계값 이하인 점들만 참 집합(Inlier)에 포함하고, 포함된 점들이 표현하는 가장 근사한 직선의 방정식을 계산한다. 직선의 방정식에 포함된 화소들의 수를 이용하여 적합도로 계산하고, 위의 단계들을 일정 횟수 반복하여 가장 좋은 적합도를 갖는 직선의 방정식을 손 영역의 주축으로 결정한다.

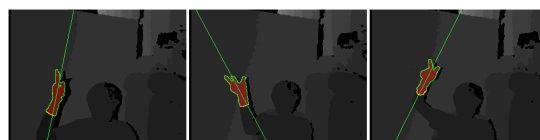


그림 5. RANSAC 결과
Fig. 5. The result of RANSAC algorithm

[그림 5]는 3.1절에서 획득한 손 영역을 이용하여 RANSAC을 수행한 결과이다. 그림과 같이 손가락의 다양한 변화에도 주축을 잘 찾음을 볼 수 있다.

2. Convex-Hull을 이용한 손 끝점 검출

손 끝점을 검출하는 방법은 3단계로 구성된다. 먼저, 손 영역으로부터 윤곽선을 검출하고, 검출된 윤곽선 대해 스무딩을 수행한다. 두 번째로 스무딩 된 윤곽선을 이용하여 관심점을 추출하고, 마지막으로 추출된 관심점들을 이용하여 Convex-Hull을 구성하여 손 끝점만을 검출한다.

2.1 곡률을 이용한 관심점 추출

영역 확장 과정에서 경계선을 검출하였으나, Square Tracing[10] 알고리즘으로 윤곽선을 다시 검출한다. 이러한 이유는 영역 확장 알고리즘은 내부에 구멍(Hole)이 있는 경우에도 구멍에 해당하는 윤곽선 또한 검출되는 특성이 있기 때문이다. 본 논문에서 사용한 깊이 영상의 경우 매우 잡음이 많았으며, 잡음이 많은 경우 곡률을 계산할 때 오류가 나타날 수 있기 때문에 윤곽선 좌표들에 대해 스무딩을 수행한다. [그림 6]의 (a)는 검출된 영역에서 윤곽선을 검출한 결과이며, (b)는 스무딩을 수행한 결과이다. 손바닥 중심으로부터

좌측의 윤곽선, 검지 손가락의 좌측 윤곽선을 포함하여 전체적으로 윤곽선이 매우 부드러운 것을 알 수 있다.

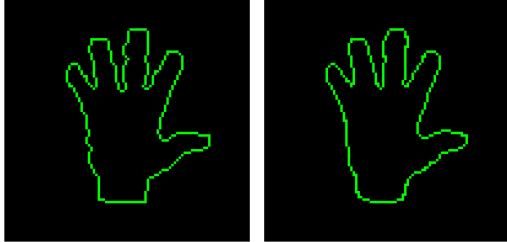


그림 6. 스무딩
Fig. 6. Smoothing

전처리된 윤곽선을 획득하면 중요한 관심점을 추출해야한다. 여기서 관심점의 정의는 변화가 큰 점을 의미한다. 즉 손 끝 또는 손가락 사이의 점과 같이 곡률이 큰 점을 찾고자한다. [그림 7]은 관심점을 찾는 방법과 결과를 보여준다. 먼저 현재 점인 P_i 가 관심점인지를 판단하기 위해 경계선의 집합에서 이전 d 만큼 떨어진 점 P_{i-d} 와 다음 d 만큼 떨어진 P_{i+d} 점을 이용하여 두 개의 벡터(V_p, V_n)를 만든다. 그리고 각 벡터의 방향을 구하고 두 벡터의 방향 차이를 현재 포인트의 곡률로 정의한다. 이렇게 하는 경우 손 끝점 근처에서 방향에 대해 큰 차이를 갖는 점들이 많이 분포될 수도 있는데, 하나의 관심점을 선택하기 위해 지역최대치를 만족하고, 임계값 이상의 값을 갖는 점을 최종 관심점으로 결정한다. [그림 7]의 (c)와 (d)는 최종적으로 선택된 관심점들을 보여준다. 그림과 같이 손 끝점뿐만 아니라 손가락 사이의 점들도 검출된 것을 볼 수 있다. 또한 이 점들 외에 팔목 부분에서도 검출될 수 있는데, 이러한 문제는 다음 단계에서 해결한다.

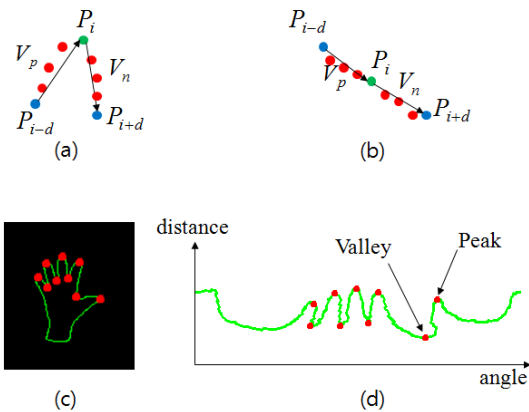


그림 7. 관심점 추출
Fig. 7. Interest point extract

2.2 Convex-Hull

이전 단계에서 손 끝점이 포함된 관심점들을 추출하였다. 하지만 손 끝점뿐만 아니라 다른 점들도 포함되어 검출되었다. 따라서 Convex-Hull을 이용하여 최종적으로 손 끝점만 검출하는 방법에 대해 제안한다.

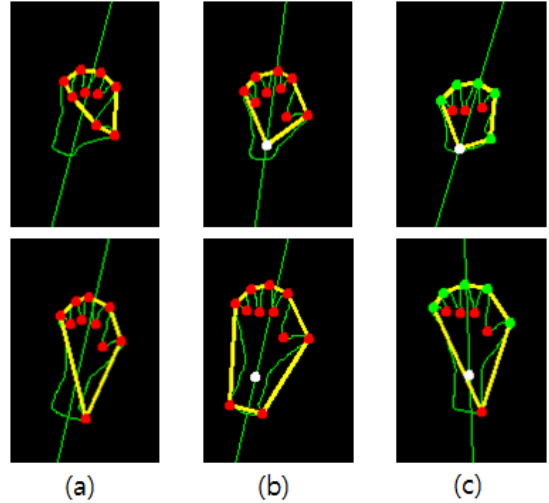


그림 8. Convex-Hull의 예
Fig. 8. The example of Convex-Hull

[그림 8]에서 (a)는 관심점만으로 Convex-Hull을 구한 경우를 보여준다. (a)의 하단 영상의 경우에는 팔목 부분에서 관심점이 검출되어 손가락 사이의 점(Valley)들은 모두 제거되었으나 (a)의 상단의 경우 손 끝점뿐만 아니라 손가락 사이의 점들도 Convex-hull을 이루는 점들에 포함 된 것을 볼 수 있다. 이러한 문제점을 해결하기 위해 이전 단계에서 추출된 관심점 외에 RANSAC으로 구한 주축을 의미하는 직선상에서 추적점을 기준으로 팔목 방향으로 일정거리 떨어진 위치에 강제적으로 관심점을 추가한다. [그림 8]의 (b)와 (c)에서 백색점이 강제적으로 추가된 점이다. (b)의 경우 강제적으로 추가된 백색점으로 인해 모든 손 끝점은 컨벡스 헐을 이루는 점들에 포함되고, 손가락 사이의 점들은 제거된 것을 볼 수 있다. 하지만 (b)의 하단 영상과 같이 팔목 부분에서 관심점이 나온 경우, 손 끝점이 아닌 점들도 포함됨을 알 수 있다. 이를 해결하기 위해, 손가락은 항상 손바닥 위에 있다는 사전 정보를 활용하여 추적점을 기준으로 주축 방향으로 상단의 점들만 손 끝점으로 인식하고 나머지 점들은 제거된다. (c)의 녹색점들은 이러한 방법으로 구한 손 끝점을 보여준다.

V. 실험 결과

본 연구에서는 입력장치로 MS사의 Kinect를 사용하여 얻은 320*240 크기의 깊이 영상을 사용하였다. 또한 모든 실험은 Intel(R) Core(TM) Quad CPU 2.66Ghz와 4GB 메모리에서 수행하였다.

표 5. 거리 별 궤적 평균 오차

Table 4. The average error of tracking with various distance condition

	1.0m	1.5m	2.0m
'a'	5.7883	4.2890	2.1928
'b'	4.3927	3.9835	2.3920
spring	4.6908	3.2395	2.9803

(단위 : pixels)

[표 4]는 손 영역 추적의 정확도를 측정하기 위해 얻은 결과이다. [그림 9]와 같이 'a', 'b', 나선형 형태를 손을 이용하여 그리면서 궤적 정보를 저장하고, 검증 자료(ground truth)와 카메라로부터의 거리를 달리하여 궤적의 평균 오차를 조사하였다. 또한 실험은 각각 10번씩 수행하여 평균을 측정하였다. 상대적으로 가까이 있는 경우에 오차가 높은 것을 볼 수 있는데, 이는 가까이 있는 경우가 멀리 있는 경우보다 같은 속도로 움직여도 더 변위가 크고 손 영역이 크기 때문에 오차가 좀 더 높게 측정되었다. 하지만 대부분 5픽셀 이하의 결과로 좋은 성능을 보여주었다.

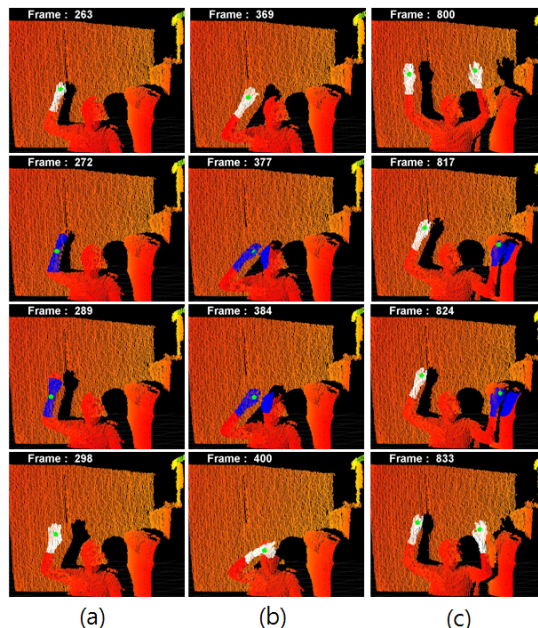


그림 10. 추적 신뢰도 예
Fig. 10 Example of tracking confidence

[그림 9]는 [표 4]의 측정 데이터 중 하나로 손 영역 추적의 궤적 결과를 보여준다. 녹색 점은 추적점을 의미하며, 녹색 선은 추적점을 이전 40프레임동안 이은 결과이다. 또한 영상 좌측 상단의 숫자는 프레임 번호를 의미한다. 상단의 결과는 'a'를 손으로 쓴 결과이며, 중간은 'b', 마지막 하단은 나선형을 각각 약 1.5초 동안 그린 궤적을 보여준다. 각 프레임 별 추적점을 보면 대부분 유사한 위치에 추적점이 존재하는

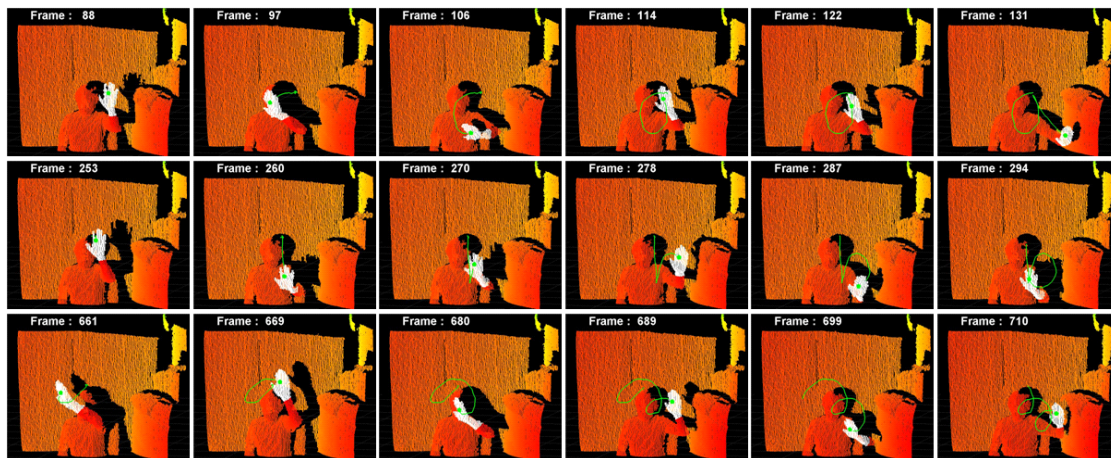


그림 9. 추적 결과

Fig. 9. The result of Tracking

것을 볼 수 있다.

[그림 10]은 3.2절에서 설명한 무효 추적 결정 방법에 의해 추적중 발생한 무효 추적 결과를 보여준다. 백색의 영역은 정상적으로 추적한 손 영역을 보여주고, 파란색의 영역은 신뢰도 평가를 통과하지 못한 경우의 손 영역을 보여준다. (a)는 손이 뒤의 벽면에 가까이 접근했을 경우에 신뢰도 평가를 통과하지 못한 경우이며, (b)는 손이 얼굴이 접촉했다가 떨어지는 경우이다. 또한 (c)는 오른손이 근처 물체와 붙었다가 떨어진 경우를 보여준다. 본 실험에서는 신뢰도 평가를 통과하지 못하더라도 일정 시간동안은 추적을 지속하도록 설정하여 정해진 시간 내에 정상적인 추적이 된 경우 지속적으로 추적을 수행하도록 하였다. 따라서 가장 하단의 영상을 보면 다시 정상적인 추적이 이루어짐을 알 수 있다.

[그림 11]은 거리 별 손 끝점 검출 결과를 보여준다. 그림의 녹색점은 추적점을 나타내고, 적색점은 검출된 손 끝점을 나타낸다. (a)는 카메라로부터 1m 떨어진 지점에서 손 끝점 검출 결과를 보여준다. 양손을 모두 편 상태이므로 10개의 손 끝점을 모두 검출한 것을 볼 수 있다. (b)와 (c)는 1.5m, 2m 거리에서 실험한 결과이다. (d)는 1.5m 거리에서 양손을 다 편 상태이나, 왼손의 약지와 소지 손가락을 검출하지 못한 것을 볼 수 있다. 이는 카메라로부터 멀리 떨어진 경우 손가락의 영역이 매우 작고 깊이 영상 자체에 잡음이 많은 경우 관심점 검출이 정상적으로 이루어지지 않아 발생하였다.

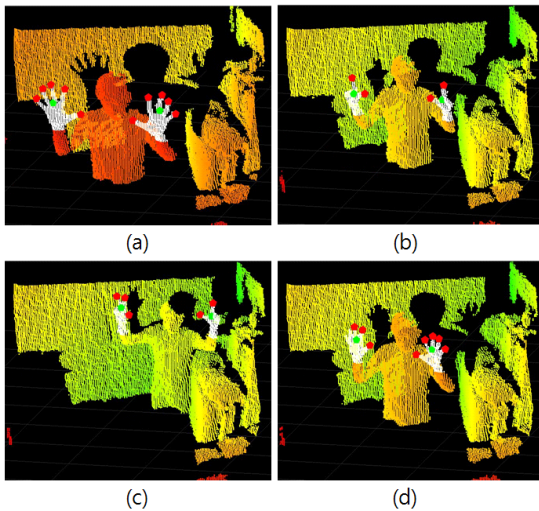


그림 11. 손 끝점 검출 (a : 1m, b-d : 1.5m, c : 2m)
Fig. 11. Fingertip Detection

[표 5]는 거리별로 500프레임 동안 다양한 거리와 손가락에 수에 따라 손 끝점의 검출 결과를 측정한 데이터이다. 1m에서는 손가락 수와 상관없이 대부분 정상적으로 검출하였으며, 1.5m에서는 약지와 소지를 편 경우에 인식률이 다소 감소하였다. 2m에서도 전체적인 검출률은 감소하였으나, 1.5m와 마찬가지로 약지와 소지를 함께 편 경우에 검출률이 현저히 감소하였다. 이는 깊이 영상의 해상도가 낮은 이유로 멀리 있는 경우 깊이 영상에 매우 많은 잡음이 생겼기 때문이다.

표 6. 거리 별 손 끝점 검출 결과
Table 5. The result of fingertip detection with various distance condition

(단위 : %)

손가락 수 \ 거리	1.0m	1.5m	2.0m
1	98.0	95.0	90.3
2	98.0	93.7	88.3
3	97.7	95.0	88.7
4	95.0	87.3	79.7
5	93.0	86.7	78.7

본 논문에서는 실제 사용자 관점에서의 측정을 위해 제스처 프로그램을 구현하여 사용자의 의도와 시스템의 인식 결과가 얼마나 일치하는지 실험하였다. [표 6]은 제안하는 방법을 통해 얻은 정보를 활용하여 얻은 5가지 제스처를 보여준다. 표와 같이 '선택', '이전', '다음'의 명령은 손가락 수와 손 영역의 주축을 사용하여 정의하였으며, '확대'와 '축소'는 손가락 수와 추적점의 이동궤적의 변화를 이용하여 정의하였다.

표 7. 제스처 정의
Fig. 6. the define of gesture

선택 (Select)	
이전 (Previous)	
다음 (Next)	
확대 (Zoom In)	
축소 (Zoom Out)	

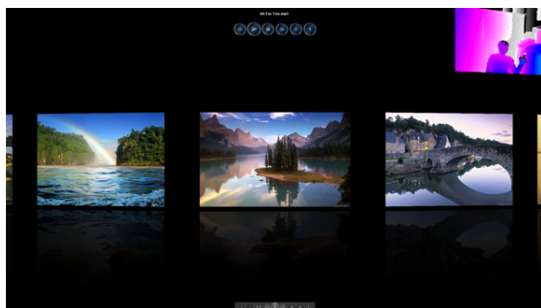


그림 12. 제스처 인식 실험 프로그램
Fig. 12. Test program for gesture recognition

[그림 12]는 정의한 제스처를 이용하여 동작하는 제스처 인식 프로그램이다. 본 프로그램을 사용하여 제스처 인식을 위해 제안한 손 끝점 검출과 추적에 대한 성능을 평가하기 위해 10명의 사용자에게 20번씩 5개의 기능을 수행하도록 실험하였다.

[표 7]은 정확도를 보여주는 것으로 각 기능 별 실험자의 의도와 얼마나 일치하였는가를 나타낸다. 먼저 실험자에게 사전에 각 명령에 대해 숙지시키고, 실제 측정은 실험자가 원하는 명령을 정하고 이를 위한 제스처를 수행하였을 경우 원하는 동작이 이루어졌는지를 세어 측정하였다. 표와 같이 “이전”, “다음”, “선택” 기능에 대해서는 대부분의 실험자의 의도에 맞게 동작하였으나, 확대 축소의 경우에는 다른 기능에 비해 오동작이 존재하였다. 빠르게 양손을 움직이는 과정에서 깊이 영상이 뭉개져 손 끝점 검출 과정에서 정확성이 떨어짐을 확인하였다. 하지만 조명을 없애고 실험한 결과로서 깊이 영상만을 이용하기 때문에 조명이 없는 환경에서도 좋은 성능을 나타냄을 확인하였다.

표 8. 제스처 정확도
Table 7. Accuracy of gesture

(단위 : %)

	Prev	Next	Select	Zoom In	Zoom out
Accuracy	98.8	99.2	97.8	90.3	89.7

VI. 결 론

본 논문에서는 깊이 영상만을 이용하여 손 영역을 추적하고, 손 영역을 분리하여 손 끝점과 손의 궤적 정보를 획득할 수 있는 방법에 대해 제안하였다. 또한 추적 과정에서 나타날 수 있는 문제점을 해결 할 수 있는 신뢰도 측정 방법과 손 궤

적 획득을 위한 안정적인 추적점 정의 방법을 제안함으로써 안정적인 추적이 가능함을 보였다. 마지막으로 손 끝점과 궤적 정보, 손 영역의 주축등의 정보를 획득함으로써 다양한 동적, 정적 제스처로 응용 가능할 것으로 생각된다. 본 논문에서는 제스처 인식을 위한 선행연구로 손 영역 추적 궤적 및 끝점 검출 연구를 수행하였으므로 향후 획득한 정보를 바탕으로 효율적인 제스처 인식에 대한 연구가 추가적으로 필요할 것이다.

참고문헌

- [1] D.H Lee and S.G Lee, “Vision-Based Finger Action Recognition by Angle Detection and Contour Analysis”, ETRI Journal, vol 33, no 3, pp. 415-422, June 2011.
- [2] A. Ramamoorthy, N. Vaswani, S. Chaudhury and S. Banerjee, “Recongition of dynamic hand gestures”, Pattern Recognition, vol. 36, no. 9, pp. 2069-2081, September 2003.
- [3] B.M. Kim, J.W. Kim, K.H. Lee, “An Application of Adaboost Learning Algorithm and Kalman Filter to Hand Detection and Tracking”, The journal of KSCI, vol 10, no 4, pp. 47-56, September 2005.
- [4] M. Van den Bergh, and L. Van Gool, “Combining RGB and ToF Cameras for Real-time 3D Hand Gesture Interaction”, 2011 IEEE Workshop on Application of Computer Vision (WACV), pp. 66-72, January 2011.
- [5] P. Trindade, J. Lobo and J. P. Barreto, “Hand gesture recognition using color and depth images enhanced with hand angular pose data”, IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), pp. 71-76, September 2012.
- [6] P. Suryanarayan, A. Subramanian, and D. Mandalapu, “Dynamic Hand Pose Recognition using Depth Data”, In 2010 International Conference on Pattern Recognition, pp. 3105-3108, August 2010.
- [7] X. Liu and K. Fujimura, “Hand gesture recognition using depth data”, Proc. 6th.

- International Conf. on Automatic Face and Gesture Recognition, pp. 529 - 534, May 2004.
- [8] M. A. Fischler, R. C. Bolles, "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography". Comm. of the ACM, Vol 24, pp 381-395, March 1980.
- [9] S.I Joo, S.H Weon, H.I Choi, "Real-time Hand Region Detection and Tracking using Depth Information", KIPS Transactions on Software and Data Engineering, vol 1, no 3, pp. 177-186, December 2012.
- [10] Square Tracing Algorithm : http://www.imageprocessingplace.com/downloads_V3/root_downloads/tutorials/contour_tracing_Abeer_George_Ghuneim/square.html

저 자 소 개



주 성 일

2008 : 한국산업기술대학교
컴퓨터공학과(공학사)

2010 : 숭실대학교
미디어학과(공학석사)

현 재 : 숭실대학교
미디어학과 박사과정

관심분야: Image Processing,
Computer Vision,
Pattern Recognition,
Machine Learning.

Email : sijoo82@ssu.ac.kr



원 선 희

2005 : 환경대학교
컴퓨터공학과(공학사)

2007 : 숭실대학교
컴퓨터학과(공학석사)

2012 : 숭실대학교
미디어학과(공학박사)

현 재 : 숭실대학교
미디어학과 Post Doc.

관심분야: Image Processing,
Computer Vision,
Pattern Recognition,
Machine Learning.

Email : nifty12@ssu.ac.kr



최 형 일

1972 : 연세대학교 전자공학과(공학사)

1982 : 미시간대학교
전자공학과(공학석사)

1987 : 미시간대학교
전자공학과(공학박사)

1995 : 퍼지 및 지능시스템학회 이사

1996 : 정보과학회 컴퓨터비전 및
패턴인식 연구회 위원장

1997 : IBM Watson Lab 방문연구원

2005 : 한국정보과학회 이사

현 재 : 숭실대학교 미디어학과 교수

관심분야: 컴퓨터공학

Email : hic@ssu.ac.kr