

Video see-through HMD 기반 증강현실을 위한 손 인터페이스*

하태진, 우운택
광주과학기술원 U-VR 연구실
{tha,wwoo}@gist.ac.kr

Video see-through HMD based Hand Interface for Augmented Reality

Taejin Ha, Woontack Woo
GIST U-VR Lab

요약

본 논문에서는 Video see-through HMD 에 기반 하여 증강 현실을 위한 손 인터페이스를 제안한다. 착용형 컴퓨터의 입력 장치로써 Video see-through HMD 에 부착된 USB 카메라로부터 영상을 입력 받은 후, HSV 컬러 공간에서 탐색 윈도우 안의 개체를 이중 임계 값을 이용해 손과 팔이 포함된 객체로 분리한다. 그 다음 거리 변형 행렬을 이용하여 손과 팔을 분리하고, 볼록 다각형 외각점 추출을 통해 손 끝의 좌표를 검출한다. 이를 기반으로 한 어플리케이션 “AR-Memo” 은 현실세계에서 손끝에 가상의 펜을 증강하여 메모를 하고 이동중에 손바닥을 통해 메모를 볼 수 있다. 증강 현실 기반 손 인터페이스를 사용함으로써 사용자는 이동중에도 직관적으로 입력을 할 수 있다. 또한 어떠한 물리적인 장치나 마커를 손에 부착하지 않기 때문에 자연스러운 인터페이스이다. 본 시스템은 착용형 컴퓨터와 결합되어 사용자에게 편리한 인터페이스를 제공할 수 있을 것으로 기대된다.

Keyword : Ubiquitous Computing, Augmented Reality, User Interface

1. 서론

컴퓨터 기술의 빠른 발전으로 인하여 사용자는 고성능의 소형화된 컴퓨터를 휴대하며 시간과 장소에 상관없이 언제 어디서나 편리하게 사용할 수 있게 되었다. 이에 따라 착용형 컴퓨팅의 시대가 다가오고 있고, 이동중의 사용자에게 편리하고 휴대성이 좋은 인터페이스를 제공해야 할 필요성이 증가되고 있다.

이러한 이동형 인터페이스로 보편적으로 사용되는 장치는 내장형 키보드, 접이식 키보드 [1] 그리고 미니 키보드등이 있다. 그러나 이러한 장치들 대부분은 이를 사용할 수 있는 일정한 공간과 장소를 필요로 한다는 제약과, 이동하면서 사용하기에 불편하다는 단점이 있다. 이를 개선한

인터페이스로써 트위들러 [3]가 있다. 하지만 이 장치 역시 항상 손에 쥐고 있어야 한다는 단점을 가지고 있다.

본 논문에서는 이러한 단점들을 보완하기 위하여 HMD 를 이용한 증강 현실 기반 손 입력 인터페이스를 제안한다. 이를 위해서 첫 번째로 HMD 에 부착된 USB 카메라로부터 얻어진 영상을 HSV 컬러 공간으로 변환 후, 탐색윈도우 내에서 이중 임계값 처리를 통해 손과 손목이 포함된 객체로 분리한다. 두 번째로는 손과 팔이 포함된 객체에서 거리변형 행렬을 이용해 손과 팔을 분리하고 볼록 다각형의 외각점 추출을 통해 손가락 끝을 검출한다. 마지막으로 이를 응용한 어플리케이션으로써 “AR-Memo” 는 손가락 끝의 좌표를

* 본 연구는 광주과학기술원 문화기술연구소 (CTRC)와 실감콘텐츠연구센터 (ICRC)의 지원에 의한 것임

이용해 가상의 펜을 증강하여 메모를 하고, 저장한 메모는 손바닥에 증강한다. 그림 1은 전체 알고리즘의 흐름을 보인다.

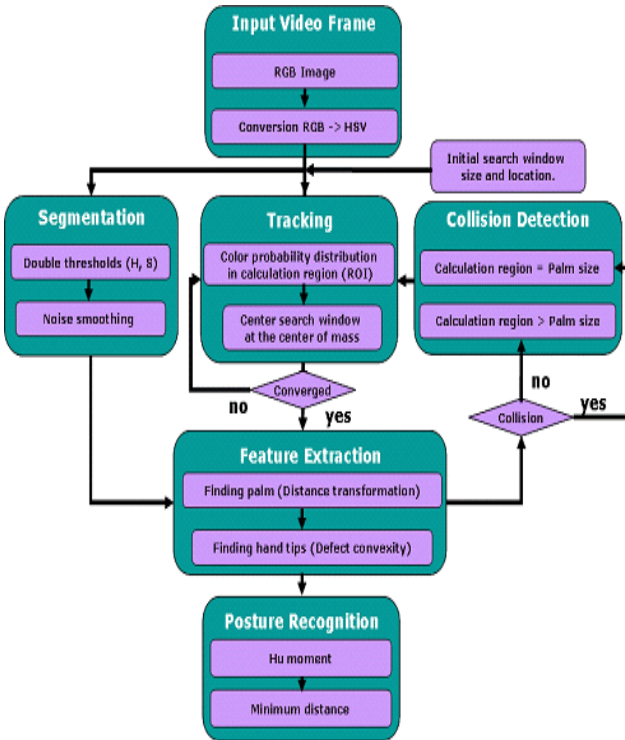


그림 1. 전체 알고리즘 흐름도

제안된 시스템은 다음과 같은 장점을 갖는다. 첫째, 사용자는 직관적으로 손을 입력 장치로 사용할 수 있다. 둘째, 장소에 제약이 없이 이동 중에도 입력이 가능하다. 마지막으로 인식을 위한 특정 마커 [4]가 필요가 없다.

본 논문은 다음과 같이 구성된다. 2 장에서는 전체 알고리즘에 대해 설명하고, 3 장에서는 이를 바탕으로 AR-Memo에 대한 설명을 한다. 4 장에서는 본 시스템의 구현 및 결과에 대해서 다룬다. 그리고 5 장에서 결론을 맺는다.

2. AR Memo 인터페이스

2-1 손 분리

손 분리란 배경영상으로부터 손 객체를 분리하기 위한 과정으로써 손 자세나 제스처 인식을 위한 출발점으로써 가장 중요한 단계이다. 사용자가 이동 중에도 사용할 수 있는 인터페이스를 위해서는 계속 변하는 조명 조건과 복잡한 배경에서도 강인한 손 영역의 분리가

선행되어야 한다. 또 자연스러운 인터페이스를 위해서 특정한 마커를 사용하지 않아야 한다. 이를 위해서 본 논문에서는 첫 번째 단계로 RGB 컬러 영역에서 HSV 컬러 영역으로 변형을 한다. 입력 영상의 컬러 성분은 RGB 인데, 각각의 성분들에 휘도 성분이 포함되어있기 때문에 조명에 따라 특징치의 값이 매우 현저하게 변환한다. 따라서 식 (1)에 따라 RGB 컬러 영역에서 HSV 컬러 영역으로 바꾼다. 휘도 성분을 포함하지 않은 색조 성분인 H 값과 채도 성분인 S 값이 추출된다. 이들 값은 조명의 영향을 덜 받는다 [5].

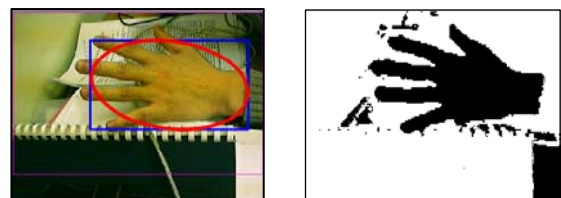
$$H = \cos^{-1} \left(\frac{\frac{1}{2}[(R-G) + (R-B)]}{[(R-G)^2 + (R-B)(G-B)]^{1/2}} \right) \quad (1)$$

$$S = 1 - \frac{3}{(R+G+B)} [\min(R, G, B)]$$

H 성분과 S 성분 각각에 대해 이중 임계 값을 적용한 후, 교집합을 구해 그림 2와 같이 배경 영상으로부터 손 객체를 분리한다.

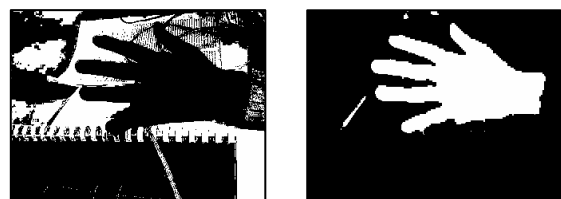
$$\text{If } (H_{\min} < I_{xy} < H_{\max}) \wedge (S_{\min} < I_{xy} < S_{\max}) \quad (2) \\ \text{then skin-region}$$

여기서 $H_{\min}(S_{\min})$ 와 $H_{\max}(S_{\max})$ 는 각각 색조(채도) 성분의 임계값에 대한 최소값과 최대값을 나타내고, I_{xy} 는 해당 좌표의 밝기값이다.



(a)

(b)



(c)

(d)

그림 2. (a) 입력 영상 (b) 색조에 대해 이중 임계 값 처리 (c) 채도에 대해 이중 임계 값 처리 (d) AND 연산 후의 결과 이미지

2-2 손 영역 추적

안정적인 손 객체를 검출하기 위해 탐색 윈도우를 이용해 손 객체를 추적하고 이 영역 내에서만 영상처리를 한다. 이렇게 함으로써 탐색 영역 밖의 얼굴이나 다른 손 객체는 무시할 수 있다. 본 논문에서는 Continuously Adaptive Mean Shift Algorithm [3]을 이용하여 손 객체를 추적한다.

그러나 이 알고리즘을 손 영역 추적에 적용할 경우 두 가지 문제점을 가진다. 첫 번째로 유사한 색상을 지닌 객체를 포함하도록 탐색 윈도우가 계속 확장된다는 점이다. 따라서 손 뿐만 아니라 팔목과 팔까지 탐색 윈도우가 확장된다. 이것을 해결하기 위해서 다음 절에서 구한 손바닥의 중심점으로 탐색윈도우의 중심점을 이동시키고, 탐색 윈도우크기를 손바닥의 반지름의 일정 배수 크기로 제한한다. 이렇게 함으로써 탐색윈도우가 무한히 확장되는 것을 방지하고, 손바닥을 중심으로 탐색윈도우를 설정한다.

$$Sc = Hc \quad (3)$$

$$Sr = c * Hr$$

여기서 Sc 와 Hc 은 각각 탐색윈도우와 손바닥의 중심 좌표이며, Sr 과 Hr 은 각각 탐색윈도우의 크기와 손바닥의 반지름이다.

두 번째 문제점은 이 알고리즘에서 쓰이는 역투영 이미지에서는 객체의 윤곽을 구하기 어렵다. 따라서 본 논문에서는 윤곽을 구하는 문제는 탐색윈도우내의 이중 임계 값 처리를 한 후의 이진이미지 결과 값을 이용한다. 그림 3 은 탐색 알고리즘을 처리한 후의 역 투영 이미지와 이중 임계 값 처리된 이미지를 통해 손 영역을 구한 결과이다. 또한 그림 4 와 같이 이중 임계 값 사이의 범위를 크게 정한 경우, 탐색 윈도우 내에서만 영상처리를 하기 때문에 이 영역 밖에

얼굴이나 다른 손이나 객체가 존재해도 원하는 손 영역만을 적절하게 검출한다.

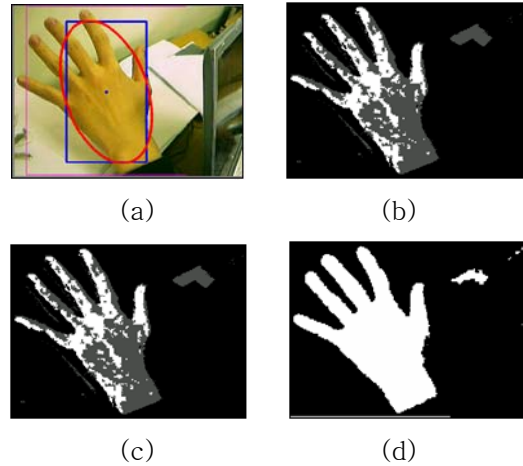


그림 3. (a) 입력 영상 (b) 역 투영 이미지 (c) 역 투영 이진 이미지 (d) 이중 임계 값 처리를 통한 이진 이미지

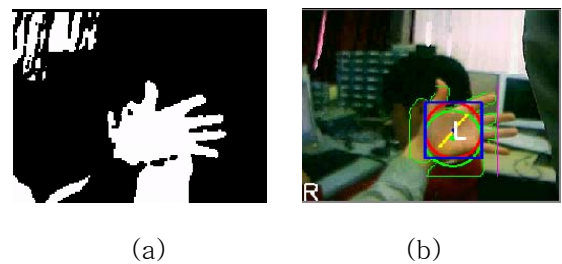


그림 4. (a) 임계값의 범위를 크게 한 경우의 이진 이미지 (b) 탐색 영역 내에서 영상처리를 통해 손 영역 검출

2-3 손과 팔의 분리 및 손 끝 검출

손과 팔은 유사한 색이기 때문에 손 객체만 검출하기 위해서는 이 두 영역을 분리 하는 방법이 고려가 되어야 한다. 이를 위해서 표 1 과 같은 거리 변형 행렬을 손 영역이미지에 적용한다. 이를 통해 나온 결과 값을 중에서 가장 큰 값의 좌표를 찾으면 손바닥의 중심점이 된다.

4.5	4	3.5	3	3.5	4	4.5
4	3	2.5	2	2.5	3	4
3.5	2.5	1.5	1	1.5	2.5	3.5
3	2	1	0	1	2	3
3.5	2.5	1.5	1	1.5	2.5	3.5
4	3	2.5	2	2.5	3	4
4.5	4	3.5	3	3.5	4	4.5

표 1. 거리 변형 행렬

손바닥의 크기는 손바닥의 중심점 좌표로부터 픽셀 값이 0 이 될 때까지 원의 반지름을 늘려가며 검사한다.

While ($I_{xy} \neq 0$) (3)

$$x = r * \cos(\theta) + C_x$$

$$y = r * \sin(\theta) + C_y$$

I_{xy} 은 x, y 좌표의 밝기값이고, r 은 반지름을 나타내며 증가 값은 1 이다. θ 의 범위는 $[0,360]$ 이고 증가 값은 1 이다. C_x 와 C_y 는 각각 손바닥 중심 좌표 x 값과 y 값이다. 그림 5 은 손바닥의 영역을 검출 한 결과이다.

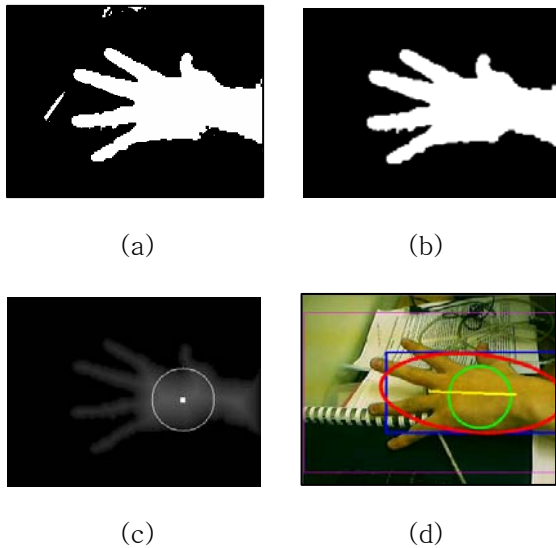


그림 5. (a) 이진 이미지 (b) 탐색 윈도우 내에서 최대 질량을 갖는 객체 추출 (c) 손바닥의 중심 좌표와 반지름 구함 (d) 입력 이미지에 표시

손끝의 좌표를 검출 하기 위해 3 coins algorithm” [6]을 사용한다. 그러나 손가락 끝 마디는 수많은 볼록 외각 점으로 구성되게 되기 때문에 손가락의 끝 한 개의 좌표를 얻기 위한 방법이 필요하다. 이를 위해서 볼록 다각형의 외각 좌표들 간의 거리에 제한을 둔다.

$$\text{If } ((H_x^k - H_x^{k-1})^2 + (H_y^k - H_y^{k-1})^2 > d) \quad (4)$$

Then fingertip

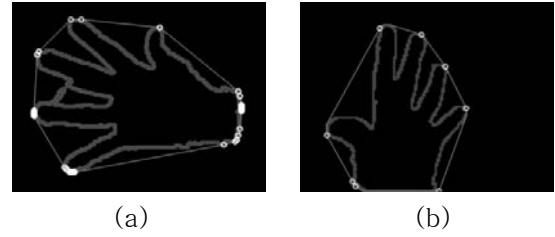


그림 6. (a) 다수의 볼록 외각 점 (b) 점들 거리에 제한을 둔 결과

2-4 양 손 추적 및 충돌 검출

양 손 추적을 위하여 위에서 제안한 추적알고리즘과 탐색윈도우를 두 번 적용하고, 또 탐색 윈도우를 손바닥의 크기와 위치에 따라 동적으로 변화도록 하였다. 만약에 양 손이 겹치게 된다면 한 쪽의 탐색윈도우가 다른 한 쪽으로 탐색 윈도우 쪽으로 이동할 위험이 있다. 본 논문에서는 경계 사각형을 이용하여 두 번의 충돌 검출을 시행한다.

첫 번째 충돌 검출	확장된 탐색 윈도우 (손 포함 크기)
두 번째 충돌 검출	축소된 탐색 윈도우 (손바닥 포함 크기)

표 2. 두 번의 충돌 검출

첫 번째 충돌이 일어나면 확장된 탐색 윈도우를 축소된 탐색 윈도우 형태로 만든다. 이 경우 그림 7 처럼 손가락은 겹쳐도 손바닥은 겹치지 않는 상태가 된다. 거리가 멀어져 충돌이 일어나지 않는다면 양손은 분리가 가능하다. 두 번째 충돌처리는 손바닥간의 충돌처리로 이 경우에는 탐색 윈도우가 사라지고 탐색 윈도우를 다시 설정해야 한다.

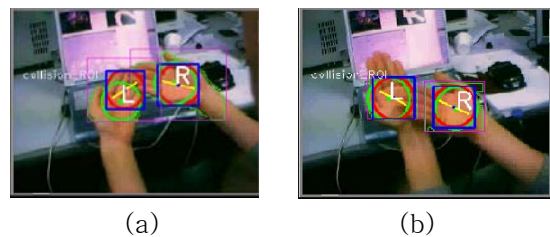


그림 7. (a) 확장된 탐색 윈도우 간의 충돌 (b) 축소된 탐색 윈도우

3. AR-Memo

사용자는 HMD 을 착용한 상태에서 손끝에 펜을 증강시켜 현실세계의 영상에 직접 메모할 수 있다. 펜이 증강되기 위해서는 그림 8 와 같이 세 단계의 절차가 필요하다. 처음에는 보통의 "virtual stick" 이 증강된다. 만약 엄지손가락과 검지손가락 끝이 만난다면 "simple pen" 이 되고, 2 초 동안 이런 상태를 유지하면 "magic pen" 이 된다. 이렇게 임계시간을 도입함으로써 펜이 증강되자마자 선을 그려져서 원하는 위치에 적절하게 메모를 하지 못하는 현상을 제거한다.

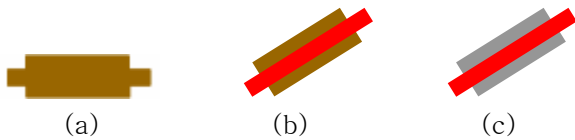


그림 8. (a) virtual stick (b) simple pen (c) magic pen

자유 곡선을 그리는 방법은 펜 끝의 위치를 검출하여 그 점의 좌표를 배열 구조 체에 저장한 후, 처음 좌표부터 두 좌표씩 추출하여 연결한다. 그러나 특정한 어느 한 위치에서 메모를 하다가 다른 위치에 메모를 하게 될 경우, 선이 연결되어 버리는 현상이 발생한다. 이런 현상을 제거하기 위해서 점 사이의 거리를 계산하여 임계 거리 이상일 경우 선을 연결하지 않는다.

4. 구현 및 결과

본 장에서는 제안된 시스템에 의한 실험 결과를 나타낸다. 본 시스템의 입출력 장치로써 저가의 15 FPS 나오는 USB 카메라가 부착된 대양 ENC 의 Cy-glassed 를 사용하였다. 이 HMD 는 640*480 pixel 의 해상도를 가진다. 카메라의 출력은 Pentium 3, Mobile CPU 1.13Ghz, 256 RAM 에 연결하고, 컴퓨터의 비디오 출력은 다시 HMD 로 연결하였다. 컴퓨터는 HMD 에 부착된 카메라로부터의 비디오 영상 처리와 HMD 상에 디스플레이 하기 위한 가상객체의 생성이 그 목적이다. 컴퓨터 비전 라이브러리는 OpenCV 4. beta 을 참조하였다.

실험 환경은 보통의 실내에서 복잡한 배경에서 하였다.

사용자는 그림 9 과 같이 HMD 을 착용한 상태에서 "Magic pen" 을 이용하여 현실세계의 영상에 직접 메모할 수 있다. 또한 AR-Memo 에서는 그림 10 과 같이 펜의 컬러와 두께를 바꿀 수 있다. 손끝의 좌표와 메뉴상자의 경계가 충돌이 일어났을 경우 메뉴가 선택되게 된다. 메모를 마친 후에는 그림 11 과 같이 왼손 바닥 위에 메모를 했던 이미지 또는 문서들도 증강이 된다. 추가적으로 사용자에게 오락적 요소를 제공하기 위해서 그림 12 과 같이 손끝을 이용한 테트리스 게임을 구현하였다. 손끝을 이용하여 블록을 움직인다.

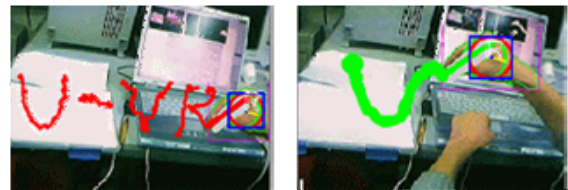


그림 9. (a) 증강된 가상 펜 및 메모 (b) 펜의 색깔과 굵기를 바꾼 후 메모

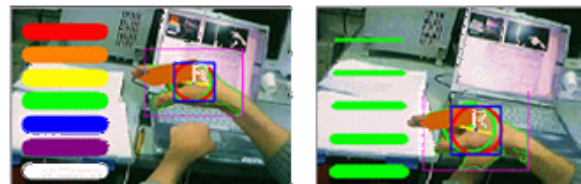


그림 10. (a) 펜의 색깔 선택 메뉴 (b) 펜의 굵기 선택 메뉴



그림 11. (a) 손바닥에 메모 증강 (b) 손바닥에 문서 증강



그림 12. 양손을 이용한 테트리스 게임

PDA 에서 스타일러스 펜을 이용해 메모하는 방법과 본 손 인터페이스의 어플리케이션인 “AR-Memo“ 의 성능에 관한 실험을 했다. “U-VR” 이라는 글자를 쓰는데 걸리는 속도와 글자 크기의 최소값과 최대값을 알아보았다. 메모공간의 크기는 320 X 240 Pixel 이다.

	메모 속도 (평균 글자 수/초)	글자 크기 (Pixel)
PDA 에서 메모	1	10~320
AR-Memo	0.3	80~300

표 3. 메모 성능 비교

다음으로 메모하는 것, 펜의 색깔과 굵기를 선택하는 것, 저장한 영상을 보는 것 그리고 테트리스 게임을 하는 것에 대한 직관성과 만족도에 대한 비교를 하였다. 본 논문에서 제안한 손 인터페이스를 이용한 방법과 PDA 스타일러스 펜을 이용한 방법 중 상대적으로 어느 것에 더 만족하는지에 따라 “불편하다” 또는 “편하다” 나타내었다.

	PDA	AR-Memo
메모	편하다	불편하다
색깔 선택	불편하다	편하다
굵기 선택	불편하다	편하다
영상 보기	불편하다	편하다
테트리스 게임	불편하다	편하다

표 4. 상대적인 만족도 비교

5. 결론 및 향후 연구

본 논문에서는 시각 기반 증강 현실 기술을 활용한 사용자 인터페이스를 제안하였다. 추적을 통한 탐색 윈도우 내에서 손 객체를 분리, 손바닥 영역 추출, 손끝 추출 그리고 두 손 추적과 손 사이의 충돌 처리를 하였다. 제안된 인터페이스를

AR-Memo, Viewer 그리고 테트리스 게임등과 같은 응용 어플리케이션에 적용하였다. 제안된 손 인터페이스는 착용 형 컴퓨터와 결합되어 사용자에게 직관적인 인터페이스가 될 수 있다.

그러나 메모를 할 때 시선이 고정되어야 한다는 점, 불안정한 펜의 증강으로 매끄럽지 못한 글자 그리고 그로 인해 메모하는 데 걸리는 시간이 늘어나는 단점과 불편하다는 결론을 얻었다. 그러나 메뉴를 선택하거나 제어를 위한 인터페이스로는 기존 인터페이스에 비해 상대적으로 더욱 직관적이며 편하다는 것을 알 수 있다.

향후에는 펜의 안정적인 증강을 위해 필터를 적용하는 것과 손바닥의 영역이 겹쳐졌을 경우 안정된 손 영역 분리를 위해 3 차원 시각 정보를 이용하는 것이다. 또한 실내 환경뿐만 아니라 실외환경의 조명에도 강인한 색깔 분리 모델을 고안해야 한다.

6. 참고 문헌

- [1] Targus Universal folding keyboard, <http://www.targusmall.co.kr>
- [2] HandkeyCorporation, <http://www.handykey.com>
- [3] ARToolKit, <http://www.hitl.washington.edu/artoolkit>
- [4] Godart Greory, Localization and Segmentation of Annotating Object for Region-based Video Coding, M.Sc Thesis No. LiTH-ISY-EX-1848, Image Coding Group, Linköping University, Sweden, 1997.
- [5] G. Bradski. Computer vision face tracking for use in a perceptual user interface. Intel Technology Journal, 1998.
- [6] Avis, D. and Toussaint. An optimal algorithm for determining the visibility of a polygon from an edge. IEEE Trans. Comput. 30, 910-914