

모바일 폰 터치스크린에서 허프변환 기반의 반자동식 정점 검출 알고리즘 (Hough Transform-based Semi-automatic Vertex Detection Algorithm on a Touch Screen Mobile Phone)

장 영 균 * 우 운 택 **
(Youngkyoon Jang) (Woontack Woo)

요약 본 논문에서는 모바일 폰 터치스크린에서 허프 변환(hough transform) 방법을 이용한 반자동식 정점(vertex) 검출 알고리즘을 제안한다. 제안하는 방법은 사용자의 입력 위치를 기준으로 작은 범위의 ROI(region of interest) 영역 내에서 한정된 매개변수 범위를 탐색하는 허프 변환을 수행함으로써 빠른 처리속도를 보인다. 또한 허프 변환을 통해 검출된 후보직선 중에서 사용자 입력 위치와 가까운 두 직선을 선택함으로써 영상의 잡음으로 인해 잘 못 검출된 후보 직선을 제거할 수 있다. 그리고 검출된 두 직선의 교점을 정점으로 검출함으로써 초기 정점 선택을 위한 입력 이외의 추가적인 상호작용 과정 없이 정확한 정점을 검출할 수 있다. 결과적으로 본 논문에서는 실험을 통해서 제안하는 알고리즘이 센싱 정밀도가 낮은 터치스크린(평균 센싱 오류: 6.3픽셀)에서의 부정확한 입력(5.7 픽셀)에도 1.4픽셀 정도의 정확한 정점 검출 성능을 보이는 것을 확인할 수 있었다.

키워드: 정점선택, 허프변환, 모바일 폰 영상처리

Abstract This paper proposes hough transform-based semi-automatic vertex detection algorithm for object modeling on a mobile phone supporting touch-screens. The proposed algorithm shows fast processing time by searching the limited range of parameters for computing hough transform with a small range of ROI image. Moreover, the proposed algorithm removes bad candidates among the detected lines by selecting the two closest candidate lines from the position of user's input. After that, it accurately detects an interesting vertex without additionally required interactions by detecting an intersection point of the two lines. As a result, we believe that the proposed algorithm shows a 1.4 pixel distance error on average as a vertex detection accuracy under such conditions as a 5.7 pixel distance error on average as an inaccurate input.

Key words: Vertex Detection, Hough Transform, Image Processing on Mobile Phone

1. 서론

최근 신기술을 바탕으로 모바일 컴퓨팅 환경의 발전에 따라 U-VR[1-4] 분야에서는 현실과 가상 세계의 이음매 없는 연결을 실현하기 위한 일환으로 현장에서 모바일 폰을 이용하여 직접 실세계의 객체를 가상의 객체로 모델링하려는 연구를 진행 중에 있다. 이를 위한 기존 방법 중에서는 미리 촬영한 실세계의 객체 영상을 배경에 두고 객체의 정점 좌표를 선택, 조정하는 과정을 거쳐서 객체를 모델링 하는 기술이 많이 요구 된다.

실세계의 객체 모델링을 위해 Hengel은 비디오 시퀀스를 입력으로 하여 장면(scene)을 모델링하는 방법 제안하였다[5]. 이 방법은 객체의 정점 선택을 위해 비디오의 각 프레임 영상을 분할(segmentation)하고 "superpixels" 기반으로 에지(edge) 탐색[6]을 수행한다. 하지만 연산 능력이 낮은 모바일 폰에서 이 과정은 많은 처리 시간을 소요하는 문제가 있다. 또한 Hengel의 방법은 정확한 위치의 정점 선택을 위해서 마우스 클릭을 통한 초기 정점위치 선택과 끌기(drag) 상호작용을 통한 정점 위치 조정 과정을 반복적으로 수행하는데, 이도 역시 낮은 센싱 정밀도를 가지는 터치스크린을 탑재한 모바일 폰에서는 정확한 선택과 세밀한 조정에 문제가 있다. 또 다른 응용에서는 현실 세계의 건물을 3차원 구조로 복원하기 위해서 제안한 방법으로, Gibson은 정점을 바로 선택하지 않고 기본이 되는 뼈대(frame) 상자 모델을 영상에 겹쳐 놓고 제시된 모델의 정점을 영상에 있는 건물의 정점으로 반복 조정하여 맞추도록 하는 방법을 제안하였다[7]. 하지만 이 방법도 Hengel의 방법과 마찬가지로 모바일 폰 환경에서 세밀한 조정을 하기에는 어려움이 있다.

* 본 연구는 문화체육관광부 및 한국콘텐츠진흥원의 2010년도 문화콘텐츠 산업기술지원사업의 연구결과로 수행되었음

· 이 논문은 제36회 추계학술발표회에서 '터치스크린 모바일 폰에서 객체 모델링을 위한 허프 변환 기반 반자동식 정점 검출 알고리즘'의 제목으로 발표된 논문을 확장한 것임

* 학생회원 : 광주과학기술원 정보통신공학과
yjang@gist.ac.kr

** 종신회원 : 광주과학기술원 정보통신공학과 교수
woo@gist.ac.kr

논문접수 : 2009년 12월 28일

심사완료 : 2010년 2월 12일

Copyright©2010 한국정보과학회: 개인 목적이나 교육 목적인 경우, 이 저작물의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다. 이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.

정보과학회논문지: 컴퓨팅의 실제 및 레터 제16권 제5호(2010.5)

이처럼 모바일 폰 터치스크린에서 실세계 객체 모델링을 수행할 때 발생하는 기존 연구의 문제를 해결하기 위해 본 논문에서는 히프 변환 방법을 이용한 반자동식 정점 검출 알고리즘을 제안한다. 제안하는 방법은 사용자의 입력 위치를 기준으로 작은 범위의 ROI(Region of Interests)를 설정하고 해당 영역 내에서만 한정된 매개 변수 범위를 탐색하는 히프 변환을 수행함으로써 연산 능력이 낮은 모바일 폰에서도 빠른 처리속도로 후보 직선을 검출할 수 있다. 또한 얻어진 후보 직선들로부터 얻어진 교점들 중에서 사용자 입력 위치와 가장 가까운 교점을 자동 검출함으로써 세밀한 정점 조정 과정 없이도 원하는 정점을 정확하게 검출할 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 제안하는 알고리즘을 상세히 설명하고, 3장에서는 실제로 제안하는 알고리즘을 이용하여 구현된 내용을 소개하고 4장에서는 모바일 폰 터치스크린을 이용하여 실험한 결과를 통해 제안하는 알고리즘의 성능을 평가하고 5장에서는 본 논문의 결론과 향후 연구방향에 대하여 언급한다.

2. 제안하는 알고리즘

2.1 히프 변환 기반 반자동식 정점 검출의 개요

제안하는 히프변환 기반 반자동식 정점 검출 알고리즘의 블록 다이어그램은 그림 1과 같다. 먼저 사용자는 모바일 폰의 터치 기반 입력으로 영상에서 검출하고자 하는 정점의 위치를 선택한다. 그리고 이어지는 히프 변환 기반 후보직선 검출 단계에서 잡음으로 작용하는 영상의 텍스처 정보를 감소화하기 위해 선택 좌표 기반 ROI를 설정하고 설정된 영역 내에서 블러 함수 적용 후 에지를 검출한다. 검출된 에지를 입력으로 하는 히프 변환을 통해 후보 직선을 검출 하고 검출된 후보 직선들 간의 교점 좌표와 사용자 선택좌표의 거리를 계산함으로써 최적의 교점 좌표를 결정하고 최종적으로 정점으로 검출한다.

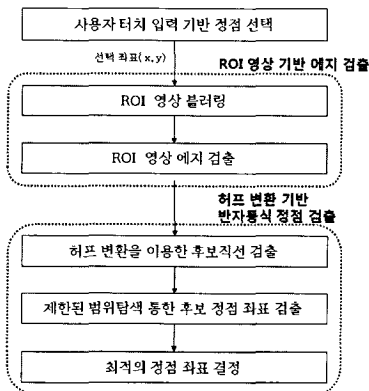


그림 1 제안하는 알고리즘의 블록 다이어그램

2.2 ROI 영상 기반 에지 검출

영상의 ROI 영역은 사용자가 선택한 좌표(x, y)를 기반으로 r만큼 상하좌우로 공간을 넓혀서 설정하였다. 그리고 좌측 상단 좌표(x-r, y-r)와 우측 하단 좌표(x+r, y+r)로 생성된 ROI 영역 내에서는 카메라 잡음의 영향을 최소화하기 위해 가우시안 블러 함수를 적용하였고, 캐니 에지 검출기로 에지를 검출하였다. 본 논문에서는 카메라 입력 영상 (640*480)에 적합한 정도의 ROI 영역을 설정하기 위해 실험을 통한 최적값으로써 r의 값을 60 픽셀로 설정하였다.

2.3 히프 변환 기반 반자동식 정점 검출

2.3.1 히프 변환을 이용한 후보직선 검출

제안하는 알고리즘은 검출된 에지의 좌표 값을 입력으로 하고 히프 변환을 통해 후보직선을 검출한다. 직선 검출에 사용하는 히프 변환 방정식은 식 (1)과 같다.

$$\rho = x \cos \theta + y \sin \theta \quad (1)$$

식 (1)에서 ρ 는 직선과 직교방향으로 떨어져 있는 원점과의 거리를 나타내고 θ 는 직선의 기울어진 각도를 나타낸다. 제안하는 알고리즘은 먼저 검출된 에지 좌표 집합 $ES = \{(x_{E1}, y_{E1}), (x_{E2}, y_{E2}), \dots, (x_{EN}, y_{EN})\}$ 를 직선의 히프 변환 식 (1)에 적용함으로써 후보직선 N 개에 대한 매개변수 집합 $PS = \{(\rho_{L1}, \theta_{L1}), (\rho_{L2}, \theta_{L2}), \dots, (\rho_{LN}, \theta_{LN})\}$ 를 검출하였다. 이때 검출된 매개변수 집합 PS 의 요소(element)들은 히프 변환 수행 시 가장 많은 수로 선출(voting)된 순서대로 나열하였다.

집합 ES 에서 EN 는 캐니 에지 검출기를 통해 검출된 에지 픽셀의 총 개수이고 집합 PS 의 LN 은 히프 변환을 통해 검출한 후보직선의 총 개수이다. 본 논문에서는 1차적으로 검출하고자 하는 후보직선의 총 개수 LN 을 "50"으로 설정하였다.

2.3.2 제한된 범위 탐색을 통한 최적의 교점 선택

제안하는 알고리즘은 최적의 교점 좌표 선택을 위해 최적의 두 후보 직선을 선택하는 과정을 수반한다. 하지만 최악의 경우 $LN \times LN$ 번의 교점 검사과정이 필요하다. 이는 연산 능력이 부족한 모바일 폰에서 처리속도를 느리게 하는 문제가 있다. 따라서 제안하는 알고리즘은 LN 개의 후보직선의 매개 변수(ρ, θ)에 대해 제한된 범위 탐색을 함으로써 연산량을 줄임과 동시에 최적의 교점을 검출하는 방법을 제안한다.

실세계에서 객체 위의 한 정점(교점)에서 만나는 두 모서리(직선)는 서로 최소한 각도 T_r 정도의 각도 차이와 T_r 정도의 거리 차이를 가지고 있다. 히프변환을 통해 추천된 직선들 중에 이 차이를 가지지 않는 직선들은 객체의 같은 모서리(직선)를 나타내는 후보직선이거나 서로 교차하지 않는 두 모서리(직선)이다. 따라서 제안하는 알고리즘은 LN 개의 후보 직선에 대해 다른 비

교되는 직선과의 교점 검출 진행 여부를 식 (2)의 조건을 만족하는지를 먼저 검사함으로써 결정한다. 이처럼 매개변수의 탐색 범위를 제한함으로써 연산량을 줄임과 동시에 잘못된 교점 검출을 피할 수 있다.

$$F(\rho_b, \theta_b, \rho_c, \theta_c) = \begin{cases} true, (T_R < Dif(\rho_b, \rho_c)) \\ \quad \wedge (T_T < Dif(\theta_b, \theta_c)) \\ false, otherwise \end{cases} \quad (2)$$

식 (2)에서 (ρ_b, θ_b) 는 그림 2에서 보이는 바와 같이 기준(base) 후보직선의 매개변수를 나타내고 (ρ_c, θ_c) 는 비교되는 후보직선의 매개변수를 의미한다. T_T 와 T_R 은 서로 비교되는 직선이 가지는 매개변수인 각도(θ)와 거리(ρ)에 대한 차이 임계값으로써 각각은 사각형 객체의 한 정점을 이루는 두 직선이 가질 수 있는 최소 각도와 폭이 좁은 객체의 정점이 가질 수 있는 최소거리가 기준이 된다. 본 논문에서는 실험을 통해 최적의 정점 검출 결과를 보이는 임계값으로써 T_T 는 60 그리고 T_R 은 5로 설정하였다. 식 (2)에서 함수 $Dif(tmp_1, tmp_2)$ 는 매개변수 tmp_1 과 tmp_2 의 차이 값을 결과로 반환한다.

제안하는 알고리즘은 식 (2)에서 보이는 바와 같이 기준 후보직선의 θ_b 와 비교 되는 다른 후보직선 θ_c 의 차이가 T_T 이상이고 기준 후보직선의 ρ_b 와 비교 되는 다른 후보직선 ρ_c 의 차이가 T_R 이상인 경우에만 두 직선 간의 교점을 구한다. 결과적으로 식 (2)에서 함수 $F(\rho_b, \theta_b, \rho_c, \theta_c)$ 의 결과 값으로 true가 반환되었을 경우에만 교점 검출을 수행한다.

그림 2의 예시에서 보이는 바와 같이 기준 후보직선이 3인 경우에 θ_3 과 T_T 미만의 차이가 나는 후보직선 1, 2, 10, 11, 12와 기준 후보직선 3을 나타내는 다른 매개변수인 ρ_3 과 T_R 미만의 차이가 나는 후보직선 4의 경우는 교점 좌표 계산을 하지 않음으로써 속도향상의 결과를 얻을 수 있다.

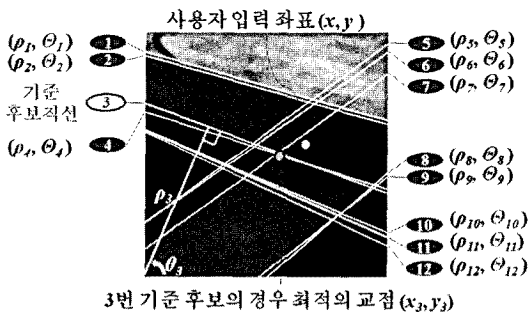


그림 2 ROI 영상의 검출된 후보직선 및 제한된 범위 탐색을 통한 최적의 교점좌표 선택 과정 예시 (LN=12)

그림 2에서 보이는 바와 같이 제한된 범위 탐색 기준으로 걸러진 후보직선 5, 6, 7, 8, 9는 기준 후보직선 3과의 교점 좌표를 구하게 되고 각 교점 좌표들은 초기에 사용자가 입력한 좌표(x, y)와 픽셀 거리를 검사함으로써 가장 가까운 교점을 기준 후보직선 3에 대한 최적의 교점좌표로 선택하게 된다. 그림 2에서 보이는 바와 같이 기준 후보직선 3의 경우 최적의 교점 좌표는 (x_3, y_3) 이다. 결과적으로 최적의 교점좌표는 교점좌표 후보 집합 $CS = \{(x_1, y_1), (x_2, y_2), \dots, (x_{LN}, y_{LN})\}$ 의 요소 중 사용자 입력 좌표 (x, y)와의 픽셀 거리가 가장 가까운 후보 교점좌표가 최적으로 검출된다.

3. 구현 및 분석

본 논문에서는 모바일 폰 터치스크린을 이용함으로써 제안하는 알고리즘을 구현하였다. 모바일 폰에 내장된 카메라를 통해 취득된 영상은 RGB565 형태로 전달되고 이 영상에 대한 기본적인 영상 처리는 'OpenCV' 라이브러리를 사용하였다. 제안하는 알고리즘은 윈도우즈 모바일 6.1 플랫폼에서 동작하도록 구현되었다.

제안하는 알고리즘은 먼저 사용자 입력을 통한 선택 좌표를 기반으로 ROI를 설정하고 객체의 무늬 또는 영상의 잡음을 줄이기 위해 가우시안 블러 함수를 적용한 후에 에지를 검출하였다. 그리고 검출된 에지를 기반으로 그림 3(a)와 같이 허프 변환을 통한 초기 후보직선을 검출 하였다. 마지막으로 제안하는 알고리즘은 제한된 범위의 매개변수 탐색을 통해 후보 직선들의 교점좌표를 검사함으로써 최적의 정점좌표를 반자동식으로 검출 할 수 있었다. 검출된 결과는 그림 3(b)와 같다.

제안하는 알고리즘을 이용하여 객체의 평면을 모델링 한 결과는 그림 4와 같다. 그림 4(a)에서 검정색 선은 사용자의 실제 선택 좌표에 의한 모델링 결과이고 노란색 선은 제안하는 알고리즘을 이용하여 반자동적으로 검출된 정점 좌표로부터 모델링한 결과이다. 그림 4(b)와 (c)에서 오른쪽 그림은 실제 모바일 폰 터치스크린에서 제안하는 알고리즘을 이용한 모델링 결과이고, 이를

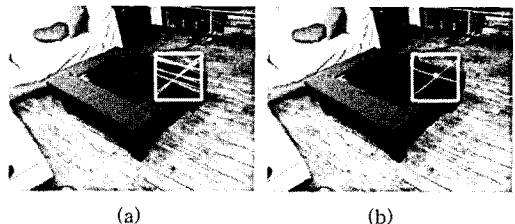


그림 3 허프 변환을 통한 직선 검출 및 제안하는 알고리즘을 통한 반자동식 정점 검출의 예시: (a) 후보직선 검출 예시 (b) 최적의 정점 검출 예시

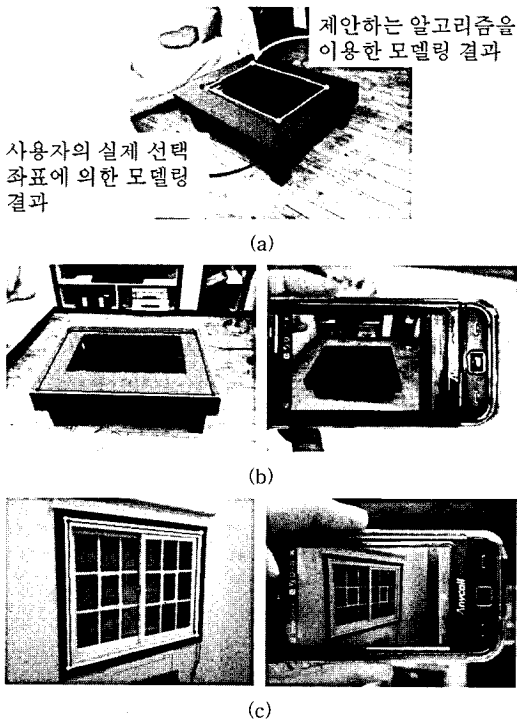


그림 4 제안하는 알고리즘을 통한 객체 평면 모델링 예시: (a) 테이블에 대한 예시 (b) (a)와 다른 각도에서의 모델링 예시 (c) 폭이 좁은 창문틀의 평면 모델링 예시

보다 명확히 나타내기 위해서 그림의 왼쪽에는 데스크톱에서 수행한 결과를 보이고 있다. 그림 4(c)에서 보이는 바와 같이 제안하는 알고리즘은 폭이 좁은 곳에서의 정점 검출도 정확히 수행하는 것을 확인할 수 있다.

4. 실험 결과 및 분석

본 논문에서는 모바일 폰으로 미리 촬영한 영상을 이용하여 알고리즘의 성능을 평가하였다. 실험에 이용한 영상은 탁자, 창문 등의 실내에서 볼 수 있는 객체를 각 3장씩 서로 다른 위치에서 촬영하여 구성하였다. 각 영상은 (640*480)크기이고 영상의 픽셀은 24 비트의 컬러 값을 가진다. 또한 본 논문에서는 3.3 인치의 터치스크린과 800MHz의 ARM11 프로세서, 128MB SDRAM을 장착한 모바일 폰을 사용함으로써 실험을 진행하였다.

가장 먼저 우리는 제안하는 알고리즘의 수행속도를 표 1과 같이 측정하였다. 표 1에서 보는 바와 같이 수행된 주요 단계 중 허프 변환을 통한 후보직선 검출이 평균 159.3 ms로써 가장 많은 처리시간을 요구하였다. 이는 객체의 모서리가 어느 정도 위치에서 얼마의 각도로 기울어져 있는지 알지 못하기 때문에 매개변수 ρ 와 θ 의

표 1 제안하는 알고리즘의 수행시간: 사용자에게 영상 속 객체의 정점 선택과정을 100번 반복 수행하도록 함

	알고리즘 주요 단계 수행 시간 (ms)			
	min	max	μ	σ
영상 블러링	0.0	9.0	2.5	2.9
에지 검출	1.0	3.0	1.5	0.5
허프 변환	69.0	245.0	159.3	70.0
정점 검출	1.0	8.0	4.4	1.9
총 수행 시간	142.0	294.0	237.6	23.9

범위를 세밀하게 탐색함으로써 생기는 문제이다. 이때 입력 값으로 이용되는 에지의 숫자가 많을수록 허프 변환의 처리 시간이 길어질 수 있는데 제안하는 알고리즘은 한정된 ROI 영역에서 블리 함수를 적용함으로써 영상에서 잡음의 영향을 줄일 수 있었다. 따라서 제안하는 알고리즘은 객체의 모서리를 나타내는 주요한 에지만을 사용하여 허프 변환을 수행할 수 있었고 상호작용에 불편함이 없을 만큼의 빠른 처리속도를 얻을 수 있었다. 그리고 제한된 매개변수 범위 탐색으로 인해 정점 검출에 요구되는 수행시간은 알고리즘의 전체 처리 시간에 거의 영향을 미치지 않는 것을 확인할 수 있었다.

결과적으로, (알고리즘의 각 주요 절차 이외에도 내부적으로 수행된 영상 복사 및 형 변환 과정 등이 모두 포함된) 정점 검출의 전체 수행 시간은 평균 237.6ms였다. 제안하는 알고리즘은 최적화 과정을 거침으로써 보다 향상된 처리 속도로 동작할 수 있다.

두 번째 실험에서는 사용자가 모바일 폰 터치스크린을 이용할 때 어느 정도의 오류를 가지고 입력을 하는지에 대한 실험을 수행하였다. 실험을 위해서 우리는 미리 정해놓은 4개의 타겟(점)을 순차적으로 보여주는 모바일 폰 터치스크린에 사용자가 최대한 정확한 위치를 터치 하도록 하였다. 사용자로부터 각 타겟에 대하여 모두 100번의 터치 입력을 하도록 요청하여 입력좌표 데이터를 모으고 분석하였다. 분석을 통해 미리 설정된 타겟의 위치와 사용자가 실제로 입력한 위치와의 RMS (Root Mean Square) 픽셀에러를 확인한 결과 평균(μ) 6.3 그리고 표준 편차(σ) 3.39 정도의 픽셀 에러를 보이는 것을 확인 할 수 있었다. 이는 모바일 폰 터치스크린을 사용하면서 평균적으로 일어날 수 있는 센싱 에러로써 기존의 모델링 과정에서 반복적으로 요구되는 정점 선택 및 정밀한 조정 단계를 모바일 폰에서 수행하기에는 어려움이 있다는 것을 확인할 수 있었다.

본 논문에서는 마지막으로 표 2와 같이 제안하는 허프 변환 기반 반자동식 정점 검출 알고리즘에 대한 정확도를 확인하였다. 실험을 위해서 미리 모바일 폰으로 촬영한 정지 영상을 이용하였고 사용자에게 영상에서

표 2 제안하는 알고리즘의 정점 검출 정확도: 사용자에게 영상 속 객체의 정점 선택과정을 100번 반복 수행하도록 함

	RMS 픽셀 에러 (pixel)			
	min	max	μ	σ
제안하는 알고리즘을 사용하지 않은 경우	1.0	11.0	5.7	2.8
제안하는 알고리즘을 이용한 경우	0.0	18.0	1.4	2.7
제안하는 방법의 검출 실패 경우(4회)를 제외한 결과	0.0	3.0	0.9	0.7

보이는 객체의 정점을 최대한 정확하게 선택하도록 요구하였다. 이 실험에서는 같은 영상에 대해 총 100번의 정점 선택을 하도록 하여 데이터를 수집하였다. 정확도는 제안하는 알고리즘을 이용하여 자동으로 검출된 정점 위치(좌표)와 기존에 실험을 위해 수동으로 미리 표시해둔 정점 위치와의 RMS 픽셀 에러를 기반으로 측정하였다. 같은 방법으로 제안하는 알고리즘을 사용하지 않은 경우에 대한 RMS 픽셀 에러 역시 측정하였다.

실험 결과 표 2에서 보이는 바와 같이 제안하는 알고리즘을 사용하지 않고 사용자가 객체 모델링을 위해 선택한 정점들은 객체의 실제 정점과 거리가 평균 5.7 픽셀 정도로 모바일 폰의 터치스크린 센싱 정밀도를 확인하기 위해 수행한 실험과 비슷한 정도의 픽셀 에러를 보였다. 반면 제안하는 알고리즘을 이용한 경우에는 1.4 픽셀 정도의 픽셀 에러를 보임으로써 제안하는 알고리즘을 사용하지 않은 경우에 비해 약 4.3 픽셀 정도의 정확도가 향상된 것을 확인 할 수 있었다. 이때 표준 편차는 2.7 픽셀로 안정적이지 않게 보일 수 있으나 이는 제안하는 알고리즘의 특성상 초기 사용자의 선택이 객체의 실제 정점과 멀리에 위치한 경우 선택된 위치와 가장 가까운 정점을 검출함으로써 생길 수 있는 오류이다. 본 실험에서 이와 같이 잘못 검출된 경우는 100번의 시도 중 총 4번 발생하였으며, 이 경우 검출된 정점은 15 픽셀 정도의 픽셀 에러 값을 보였다. 사용자가 정확한 초기 입력을 하였다든 가정 하에 이와 같이 정점을 잘못 검출한 경우를 제외했을 때 제안하는 알고리즘은 표 2에서 보이는 바와 같이 평균 0.9 그리고 표준편차 0.7 정도의 RMS 픽셀 에러로써 강건한 검출 성능을 보이는 것을 확인할 수 있었다. 결과적으로 제안하는 알고리즘을 사용할 경우 뒤에 따르는 모델링 과정에서 추가적인 정점 조정과정 없이도 정확한 정점 검출을 수행할 수 있고 따라서 적은 상호작용으로도 모바일 폰에서 손쉽게 모델링을 할 수 있음을 기대할 수 있다.

5. 결론 및 향후 연구

본 논문에서는 모바일 폰 터치스크린에서 허프 변환 기반의 반자동식 정점 검출 알고리즘을 제안하였다. 제안하는 알고리즘은 사용자가 입력한 선택 좌표를 기준으로 ROI를 설정하고 그 영역 내에서 블러 함수를 적용한 뒤 에지를 검출함으로써 영상의 잡음에 강건하게 객체의 주요한 부분의 에지 검출을 할 수 있었다. 그리고 제안하는 알고리즘은 허프 변환을 통해 검출된 후보 직선에 대해 제한된 범위의 매개변수를 탐색함으로써 연산 능력이 부족한 모바일 폰에서도 모델링하기에 불편하지 않을 정도의 빠른 처리속도(237.6ms)로 정확한 정점 검출(정확도: 1.4 픽셀 오류)을 수행할 수 있었다. 본 논문에서 제안하는 알고리즘은 반자동식으로 정점을 검출함으로써 세밀한 선택 및 조정이 어려운 모바일 폰 터치스크린에서 대략적인 선택 과정만으로도 정확하게 객체의 정점을 검출할 수 있었고, 이로써 결국 적은 상호작용으로 모델링을 할 수 있었다. 이는 추후 모바일 폰 기반 객체 인식에서 초기 객체 검출단계에 사용될 수 있으며 적은 상호작용으로 빠른 검출을 수행할 수 있다.

하지만 제안하는 알고리즘은 에지 검출결과에 의존적인 성능을 보인다. 따라서 향후 연구는 에지 주변이 비슷한 색으로 이루어진 물체에서도 강건한 반자동식 정점 검출 및 객체 모델링 방법에 대해 연구할 계획이다.

참고 문헌

- [1] Y. Lee, et al., "Ubiquitous Virtual Reality and Its Key Dimension," *IWUVR 2009*, pp.5-8, 2009.
- [2] K. Kim, et al., "u-Contents: Describing Contents in an Emerging Ubiquitous Virtual Reality," *IWUVR 2009*, pp.9-12, 2009.
- [3] S. Oh, et al., "CAMAR: Context-aware Mobile Augmented Reality in Smart Space," *IWUVR 2009*, pp.48-51, 2009.
- [4] C. Shin, et al., "CAMAR 2.0: Future Direction of Context-Aware Mobile Augmented Reality," *ISUVR 2009*, pp.21-24, 2009.
- [5] A. v. d. Hengel, et al., "VideoTrace: Rapid Interactive Scene Modelling from Video," *ACM Transactions on Graphics*, vol.26, no.3, article 86, July, 2007.
- [6] X. Ren, et al., "Learning a Classification Model for Segmentation," In *Proc. 9th ICCV*, vol.1, pp.10-17, 2003.
- [7] S. Gibson, et al., "Interactive Reconstruction of Virtual Environments from Video Sequences," *Computers & Graphics*, vol.27, pp.293-301, 2003.