

# Kinect camera를 활용한 3D Scanning의 표면 텍스처 정확도 향상을 위한 기법

김민경<sup>1</sup> 이승규\*

경희대학교 컴퓨터공학과

grreen07@khu.ac.kr, seungkyu@khu.ac.kr

## Technique for improving surface texture accuracy of 3D scanning using Kinect camera

Min Kyeong Kim<sup>1</sup> Seung Kyu Lee\*

Kyung Hee University

### 요 약

최근 다양한 크기를 가진 물체들의 형상 정보를 손쉽게 얻을 수 있는 3D스캐닝 기술이 발전하면서 제조업, 건설업 등을 비롯한 각종 산업 분야에서의 쓰임이 점점 증가하고 있다. 3D 스캐닝을 하는 방식 중 하나로 Azure Kinect의 depth camera를 활용한 Kinect Fusion 알고리즘 기반의 측정 방식이 있다. 하지만 이 방식을 통해 카메라를 움직이면서 3D 형상 정보를 얻을 때, 카메라와 물체 사이의 거리가 가까워졌다가 멀어질수록 물체의 세부적인 표면 텍스처 정보가 평균화되어 사라지는 한계점이 발생한다. 본 연구에서는 이런 한계점을 해결하기 위해 카메라와 물체의 거리가 가까워질 때의 표면 텍스처에 대한 가중치를 증가시켜 카메라의 거리가 멀어져도 표면의 텍스처 디테일을 유지하도록 하는 방식을 제안하고 구현한다.

### 1. 서 론

최근 전 세계적으로 산업 패러다임의 변화 즉, 4차 산업이 시작되며 3D 프린팅과 3D 스캐닝 사업이 성장하고 있으며 이미 다양한 산업군에서 활용되며 점차 발전하고 있다. 그중 3D 스캐닝이란 모양과 색깔 등의 데이터를 수집하기 위해 실제 세계의 물체 또는 환경을 분석하는 과정이다. 3D 스캐닝 장비의 경우 현재 고가의 전문장비부터 게임을 비롯하여 간단하게 사용 가능한 저가형 제품까지 다양하게 분포해 상용화되어 있다. 고가의 전문장비는 일반인이 구하기 어려우며 접근성이 떨어지기 때문에 일반인도 손쉽게 사용할 수 있는 Google Kinect Camera를 이용하여 3D 스캐닝을 한다면 다양한 분야에 유용하게 활용될 것이다. Google Kinect Camera를 사용하여 3D 스캐닝을 진행하는 방법은 다양한 각도와 거리에서 물체 또는 환경을 카메라로 촬영하는 것이다. 카메라로 촬영하면서 저장된 3D 데이터가 점차 축적되는 과정에서 3D 데이터들의 평균값을 구하며 전체적인 3D 데이터를 도출하게 된다. 평균값을 구하면서 3D 데이터들의 디테일한 정보가 정확해지는 경우도 있으나 윤곽이 흐려지거나 디테일이 사라지는 한계점이 발생한다. 이는 보통 카메라와 물체의 거리가 가까워지면 세부적으로 잘 표현이 되지만, 거리가 멀어지면 비교적 세세하게 표현되지 않아 카메라와 물체의 거리에 따라 표면 질감(texture)에 대한 세부적인 정보가 달라지기 때문이다. 본 연구에서는 이를 해결하기 위한 방식을 제안하고 구현하여 일반인들도 쉽게 고도화된 3D 스캐닝을 접하며 활용할 수 있는 환경이 되기를 기대한다.

### 2. 연구배경 및 관련 연구

#### 2.1 3D Scanning

3D 스캐닝은 하드웨어 장비를 이용하여 물체의 3D 형태를 측정하는 방법으로, 크게는 대상과의 접촉 여부에 따라 접촉식(3차원 측정기: CMM, Coordinate Measuring Machine)과 비접촉식(3D scanner, 3D digitizer)으로 나뉜다. 접촉식은 탐촉자를 물체에 직접 닿게 하여 스캐닝하는 방식으로 정확도가 높은 것이 장점이지만, 물체의 변형이나 손상 가능성이 있고 물체의 크기에 따라 제한적이기 때문에 주로 제품 검측에 많이 활용되고 있다. 비접촉식은 빛을 이용하여 물체를 스캐닝 하는 방법으로 레이저나 백색광으로 물체에 빛을 쬔 후 반사된 빛의 시간을 측정하고 거리를 수치화하는 방식으로 광범위한 공간을 스캐닝하는 데에 주로 사용되고 있다.

#### 2.2 Kinect Depth Camera

kinect는 Microsoft사에서 게임을 목적으로 개발한 가정용 게이밍 제품으로, kinect의 기본 구성은 적외선 레이저 카메라와 색상을 감지하는 RGB 카메라, 그리고 적외선 센서로 구성되어 있다. 기존에는 XBOX 360의 주변기기로 활용되었지만, 현재는 PC 어플리케이션 개발에 특화된 Kinect for Window가 출시되어 있는 상태이며, 기존 기기에 비해 성능이 소폭 향상되었다. kinect는 사람의 관절을 20개 영역으로 나누어 처리한다. 물론 각각의 관절에는 Tracking ID가 할당되어 있으므로 여러 사람이 들어가도 한 사람의 관절을 계속 따라 들어갈 수가 있다. kinect는 RGB 영상뿐만 아니라 특정한 적외선 점 패턴을 물체에 투영하여 점 패턴의 특성을 분석하고 깊이

정보를 획득하고 스켈레톤 트래킹을 제공한다. 이처럼 깊이가 영상 생성을 위해 물체에서 반사되어 되돌아오는 적외선의 패턴을 적외선 센서로 검출한다. 하지만 발광 장치와 적외선 센서 간의 거리 차이로 인한 사각지대 발생, 매끄러운 물체 표면으로 인하여 적외선의 난반사가 적게 일어나 적외선 센서로 레이저가 돌아오지 못하는 경우에 적외선 패턴을 감지할 수 없기 때문에 검출되지 않은 위치에 대한 값은 깊이가 영상에서 홀(hole)의 형태로 나타난다. kinect를 통해 흑백 음영 또는 히트맵으로 구성된 깊이가 영상으로 표현할 수 있게 된다.

### 2.3 Kinect Fusion Algorithm

Kinect Fusion은 2011년 Microsoft Research에서 개발한 알고리즘이다. [4] 이 알고리즘을 사용하면 사용자가 실제 장면 주위에서 Microsoft Kinect 센서3를 움직여 3D 장면을 실시간으로 강력하게 재구성할 수 있다. SLAM 기술이 효율적인 카메라 추적을 제공하지만, 기초적인 재구성만 제공하는 경우 Kinect Fusion의 결과는 높은 수준의 견고성과 세부 사항을 모두 얻을 수 있게 된다.

Kinect Fusion Algorithm의 3D reconstruction 과정은 다음과 같다. 먼저, Kinect Camera로부터 얻은 실시간 Depth 정보와 Calibration Matrix를 사용하여 카메라 좌표계(Camera coordinates)의 3D point cloud(vertices), normal map을 생성한다. 이후 ICP 알고리즘을 활용하여 이전 프레임에서 현재 프레임 카메라로의 translation과 rotation을 계산하여 전체 좌표계(World coordinates)로의 변환 행렬 T를 구하고 Camera Tracking을 수행한다. 측정된 depth 정보는 이때 구한 변환 행렬을 통해 image coordinates로부터 single consistent global coordinate 공간으로 변환될 수 있다. Global coordinate 상의 3D vertices는 Signed Distance Functions(SDFs)를 사용하여 voxels로 통합될 수 있다. 이때 표면에 대한 정보를 효율적으로 저장하기 위해 SDFs Map에서 표면에 가장 가까운 값들만 저장하게 되는데, 이러한 형식을 TSDFs (Truncated Signed Distance Fields)라고 한다. 매 프레임이 갱신될 때마다 [그림 1]에 따라 저장된 TSDF 데이터를 사용하여 volumetric integration을 실행한다.

voxel 당 GPU 스레드를 실행하는 것은 현실적으로 불가능하다. 따라서 통합된 메모리 접근을 보장하기 위해 GPU 스레드가 volume의 앞쪽 slice에 있는 각 (x, y) 위치에 할당된다. GPU 스레드는 Z축의 각 slice를 따라 이동하며 volume을 swap 한다. 볼륨의 해상도와 이것이 매핑되는 물리적 차원이 주어지면 각각의 개별 3D 그리드 위치를 global coordinates의 vertex로 변환할 수 있다. 카메라 중심(전역 카메라 변환의 변환 벡터)에서 이 정점까지의 미터법 거리를 계산할 수 있다. 이 3D 정점은 빛을 따라 실제 깊이를 측정하기 위해 이미지 좌표로 다시 project된 perspective일 수도 있다. 측정된 거리와 계산된 거리의 차이는 voxel에 대한 새로운 SDFs 값을 제공한다. (line 7) 앞서 제공된 SDFs 값은 TSDFs 값으로 정규화되고, (line 9, 11) weighted average를 통해 이전에 저장된 값과 평균화된다. (line 13) 새로운 가중치와 평균 TSDF는 모두 voxel에 저장된다.

TSDF 데이터가 통합된 이후 Raycasting 과정을 거치면 표면 데이터를 view로 추출하여 렌더링할 수 있게 된다.

### 3. 3D Scanning Algorithm 개발

#### 3.1 거리 기반 weight 알고리즘

본 연구에서는 Kinect Fusion Algorithm에서 제안한 weight 계산방식을 개선한다. 기존 방식에 따르면 표면을 저장해둔 TSDF 데이터를 새로운 프레임이 들어올 때 weighted sum 하여 갱신한다. i번째 weight 값을 구하는 방식은 다음과 같다. i-1번째 weight에 1을 더한 값과 미리 설정해둔 max weight값 중 작은 것을 택해 i번째 weight로 설정한다.

이에 본 연구에서는 가까운 때의 디테일한 표면 저장을 위해 weight의 값을 거리에 따라 다르게 설정한다.

```

if  $dist_i < min\_dist$  then
     $weight_i = weight_{min\_dist} + 1$ 
     $min\_dist = dist_i$ 
     $tsdf^{avg} \leftarrow \frac{tsdf_{i-1}weight_{i-1} + tsdf_i weight_i}{weight_{i-1} + weight_i}$ 
    store  $w_i$  and  $tsdf^{avg}$  at voxel g

```

[그림 2]

[그림 2]에 따르면 만약 현재 거릿값이 최소 거릿값보다 작다면 가장 가까운 상태이므로, 가장 가까운 상태의 weight를 증가시킨다. 그리고 현재 거릿값을 최소 거릿값에 저장한다. 이렇게 저장된 weight 값을 사용해 TSDF 평균값을 계산하여 두 프레임을 통합하고, weight 값과 TSDF 평균값을 voxel g에 저장한다.

#### 3.2 거리 기반 weight 알고리즘 kinect fusion 어플리케이션에 적용

Kinect Fusion을 활용한 3D Scanning 프로그램에 적용하기 위해 weight를 구하는 부분과 표면 정보를 담은 TSDF들을 통합하는 부분을 탐색하여 개선된 알고리즘을 적용한다. openCV에 내장된 함수에 해당하는 부분이 존재하므로, 본 연구에서 제안한 weight 수식을 적용하면 [그림 4]와 같이 카메라와 물체 간의 거리가 짧을 때 측

**Listing 2** Projective TSDF integration leveraging coalesced memory access.

```

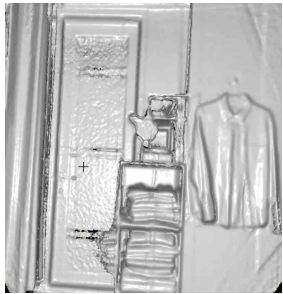
1: for each voxel g in x,y volume slice in parallel do
2:   while sweeping from front slice to back do
3:      $v^g \leftarrow$  convert g from grid to global 3D position
4:      $v \leftarrow T_i^{-1} v^g$ 
5:      $p \leftarrow$  perspective project vertex v
6:     if v in camera view frustum then
7:        $sdf_i \leftarrow ||t_i - v^g|| - D_i(p)$ 
8:       if ( $sdf_i > 0$ ) then
9:          $tsdf_i \leftarrow \min(1, sdf_i / \text{max truncation})$ 
10:      else
11:         $tsdf_i \leftarrow \max(-1, sdf_i / \text{min truncation})$ 
12:       $w_i \leftarrow \min(\text{max weight}, w_{i-1} + 1)$ 
13:       $tsdf^{avg} \leftarrow \frac{tsdf_{i-1}w_{i-1} + tsdf_i w_i}{w_{i-1} + w_i}$ 
14:      store  $w_i$  and  $tsdf^{avg}$  at voxel g

```

[그림 1]

일반적으로 volume 내에 많은 수의 voxel들이 존재하여

정된 자세한 표면 정보를 [그림 3]처럼 거리가 멀어지더라도 그대로 담고 있는 개선된 3D 모델을 얻을 수 있을 것으로 기대된다.



[그림 3]



[그림 4]

#### 4. 결론 및 기대 효과

본 연구는 비접촉 3D Scanning 방식 중 하나인 kinect camera를 활용해 Scan을 진행하고, 향상된 표면 텍스처를 가진 3D 모델을 도출해내는 것이 목표이다. 더욱 향상된 표면 텍스처를 생성하기 위해 기존 3D Scanning의 한계점인 표면 텍스처 평균화로 무더지는 현상을 개선하고 해결한다. 이러한 과정에서 kinect fusion algorithm을 참고하여 본 프로그램에서 개선된 형태로 사용된다.

본 연구의 기대효과로는 첫째, 고급 장비를 마련하기 어려운 일반인 등의 사용자들도 손쉽게 3D Scanning을 접할 수 있어 다양한 분야에서 접근성이 한 층 좋아진 상태에서 활용될 수 있다.

둘째, 게임 개발, 제조업 등에서 좀 더 편리하고 간단하게 3D Scan을 할 수 있다. kinect 카메라로 실제 풍경을 찍어 게임의 맵을 생성할 수도 있고, 부품의 디테일한 표면을 찍어 3D 모델을 생성할 수도 있다. 추후 kinect camera의 성능이 더욱 개선된다면, 본 프로그램과 kinect camera를 이용하여 상업적으로 충분히 활용 가능한 향상된 표면 텍스처를 얻을 수 있을 것으로 기대된다.

#### 참고문헌

- [1] 최순혁, 김영균, 류관희 (2017). 키넥트와 유니티3D를 이용한 물체 스캔 및 3D 모델 생성기법. 한국정보과학회 학술발표논문집, 1963-1965
- [2] 이병도, 김태혁 (2018). 4차 산업혁명을 위한 3D 스캐너와 BIM 데이터의 활용
- [3] Michele Pirovano, Kinfu - an open source implementation of Kinect Fusion + case study: implementing a 3D scanner with PCL
- [4] Shahram Izadi1, David Kim1,3, Otmar Hilliges1, KinectFusion: Real-time 3D Reconstruction and Interaction Using a Moving Depth Camera