특집논문 (Special Paper)
방송공학회논문지 제29권 제6호, 2024년 11월 (JBE Vol.29, No.6, November 2024)
https://doi.org/10.5909/JBE.2024.29.6.972
ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

인물 객체의 3차원 인스턴스 분할 기술 연구

정 영 인[a], 이 석[a)‡]

3D Instance Segmentation for Human Object

Young-in Jung[a)] and Seok Lee[a)‡]

요 약

본 논문에서는 다시점 영상을 이용한 딥러닝 기반 3차원 객체 분할 기술에 대해 연구하였다. 기존의 2차원 객체 분할 기술이 하나의 시점 정보만을 가지고 객체 분할 정보를 획득하는 것과 달리, 제안된 방법에서는 동일한 객체를 서로 다른 복수 개의 시점으로 촬영한 영상으로부터 3차원 분할 정보를 생성한다. 이 경우 다시점 영상간 기하 정보를 활용하여 2차원 객체 분할 결과보다 정확도를 높이고, 또한 시점간 일관성을 높일 수 있는 장점이 있다. 제안된 방법을 이용하여 콘텐츠 생성 시 주요 대상 객체인 사람에 대해 분할 성능을 테스트 하였고, 개별 사람의 instance 또한 구별하여 분할이 가능함을 확인하였다.

Abstract

This paper explores a deep learning-based 3D object segmentation technique using multi-view videos. Unlike traditional 2D object segmentation methods, which obtain segmentation information from a single viewpoint, the proposed method generates 3D segmentation information from videos captured from multiple different viewpoints of the same object. By utilizing geometric information between multi-view videos, this approach improves accuracy compared to 2D segmentation results and enhances consistency between viewpoints. The proposed method was tested on key target objects, such as humans, during content creation, and it was confirmed that it is also capable of distinguishing and segmenting individual instances of people.

Keyword : Segmentation, Instance, SAM, YOLO, NeRF

a) 한국기술교육대학교(Korea University of Technology and Education)
‡ Corresponding Author : 이석(Seok Lee)
E-mail: leeseok@koreatech.ac.kr
Tel: +82-41-560-1641
ORCID: https://orcid.org/0000-0002-2154-4352

※이 논문의 연구 결과 중 일부는 한국방송·미디어공학회 2024년 하계학술대회에서 발표한 바 있음.
※이 논문은 2022년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임. (2022-0-00022, RS-2022 -II220022, 초실감 메타버스 서비스를 위한 실사기반 입체영상 공간컴퓨팅 기술 개발)
· Manuscript October 16, 2024; Revised October 28, 2024; Accepted October 28, 2024.

Ⅰ. 서 론

객체 분할 기술은 입력된 2D 영상 내에서 관심객체의 분할된 영역을 추출하는 것을 목표로 하며, 영상 내 모든 픽셀 위치에 대해 클래스를 지정한 분할 마스크를 생성한다. 자동차, 사람 등과 같이 셀 수 있는 물체 및 하늘, 도로 등과 같이 셀 수 없는 물체 등 적용 가능한 객체 대상에 따라 Semantic segmentation, Instance segmentation, Panoptic segmentation으로 구분된다. 주요 2D 객체 분할 기술에는 CNN을 객체 분할에 처음 성공적으로 적용한 FCN (Fully Convolutional Networks)[1]가 있고, 의료 영상 분할을 목적으로 개발되어 현재 일반적인 자연영상에 폭넓게 사용되고 있는 U-net[2]이 있다. 그리고 최근 발표된 SAM 기술[3]은 사용자 입력 prompt에 기반하여 개발되었으며, 사전 정의된 클래스가 아닌 임의의 객체에 대한 정확한 2D 분할 성능을 나타낸다. 또한 객체 탐지를 위한 알고리즘을 우수한 성능을 가지고 있는 YOLOv8의 경우 개별 Instance에 대한 객체 탐지뿐만 아니라, 찾아진 객체의 bounding box 영역에 대해 영역분할을 적용하여 Instance segmentation이 가능하다[4].

3차원 데이터에 대한 객체분할이 가능한 분할 기술은 입력데이터의 종류에 따라 여러 가지가 제시되었다. 컬러 영상과 함께 깊이 영상을 동시에 사용하는 RGB-D 영상 분할 기술[5-7], 라이다 등 센서에 의해 획득 가능한 Point cloud 데이터 분할 기술[8], Voxel과 같은 3차원 grid 공간에 대한 분할 기술[9] 등이 있다. 최근 활발히 연구가 진행되는 NeRFs (Neural Radiance Fields) 기술[10]은 3차원 객체 분할의 새로운 가능성을 제시하고 있다. NeRF는 다시점 입력을 기반으로 네트워을 학습하여 3D consistent한 임의 시점 영상 합성이 가능하다. Semantic-NeRF[11] 기술은 기존 NeRF에서 학습 가능한 외관, 기하 정보와 더불어 영역분할을 위한 Sematic 정보를 동시에 학습하도록 확장하였다. SA3D (Segment Anything in 3D)[12]의 경우에는 앞서 기술한 2D 분할 기술인 SAM과 NeRF 기술을 접목하여, 다시점 영상으로부터 최적 3차원 그리드 마스크를 계산하고, 이를 2D 마스크 렌더링에 이용하여 임의 시점에서의 객체 분할 마스크를 생성하는 기술을 제시하였다.

이러한 객체 분할 기술들은 문제 중 하나는 인공지능 학습을 위한 Labeling된 분할 데이터의 획득이 어렵다는 것이다. 객체 분할 네트웍의 성능은 학습용 데이터의 양과 Labeling의 정확도에 크게 의존적인 반면, 해당 학습데이터는 각 시점 영상의 모든 픽셀에 대해 분할 결과가 필요하기에, 실사 영상에 대한 객체 분할 학습데이터의 확보는 한계가 있다. 이를 위해 비지도 학습 (Unsupervised learning) 기반의 3차원 객체 분할 기술들이 연구되고 있으며, [13]의 경우 객체 및 배경에 대해 독립적인 3D radiance field와 semantic field를 생성하고, 배타적인 field 조합으로 생성된 영상과 입력영상간 오차가 최대화 되도록 각 field를 학습함으로써 labeling된 사전 데이터 없이 3차원 객체 분할을 시도하였다. 하지만 비지도 학습 방법의 특성상 지도학습 대비 성능이 낮아, 아직은 전경/후경 수준의 구분만 가능한 정도이다.

본 논문에서는 다시점 영상을 이용하여 3차원 객체를 분할하는 방법에 대해 연구하였다. 특히 대상 객체를 일반적인 물체가 아닌 사람에 한정하여 성능을 높일 수 있는 방법을 고민하였다. 또한 SA3D 기술이 객체 영역에 대한 수동 입력 정보를 필요로 하는 것과 달리, 사용자의 개입 없이 자동으로 객체를 분할 가능한 기술을 개발하였다. 제안된 방법은 각 입력 시점에 대한 2D 분할 정보를 3차원으로 확장하여 3D consistent한 결과를 얻었으며, 개별 객체에 대한 instance 분할도 가능하였다.

Ⅱ. 기존 기술

1. SAM

Segment Anything Model (SAM)은 Meta AI에서 개발한 범용 이미지 분할 모델이다[3]. 이 기술은 다양한 종류의 이미지에서 객체에 대해 높은 수준의 분할 성능을 제공한다. 이 모델은 약 11억 장의 방대한 양의 사전 데이터 학습을 통해 다양한 객체 및 장면에서의 일반화된 성능을 제공한다. 이 모델은 사용자에게 다양한 형태의 프롬프트(점, 박스, 마스크) 등을 입력받아 분할을 수행한다.

SAM의 아키텍처는 이미지 인코더, 프롬프트 인코더, 마스크 디코더로 총 세 가지 주요 구성 요소로 이루어져 있다. 이미지 인코더는 Vision Transformer (ViT)[15]를 기반으로 하는 MAE pre-trained ViT를 사용하여 입력 이미지를 고해

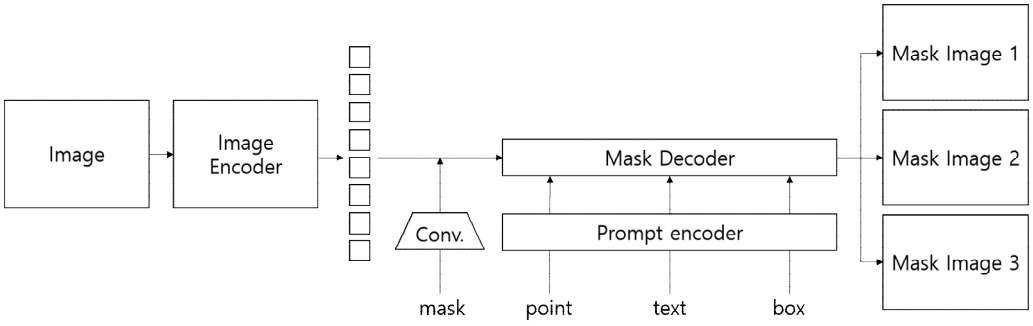

그림 1. SAM 모델의 개략적인 구조
Fig. 1. Rough Architecture of SAM model

상도의 이미지 임베딩으로 변환하여 이미지의 특징을 추출한다. 이미지 인코더는 이미지당 한 번 실행되며, 모델을 프롬프트 하기 전에 적용된다. 이후 프롬프트 인코더는 사용자로부터 입력받은 다양한 형태의 프롬프트를 임베딩 벡터로 변환한다. 프롬프트는 크게 두가지 종류가 있으며, Sparse 프롬프트인 점, 박스, 텍스트와 Dense 프롬프트인 마스크가 그 예이다. 이 중 텍스트 프롬프트는 CLIP의 텍스트 인코더를 활용하여 점, 박스와 같은 위치 인코딩으로 변환된다. 또한 마스크 프롬프트는 convolution을 사용하여 임베딩된다. 마지막으로 마스크 디코더에서는 전 단계의 실행 후 결과인 임베딩, 토큰 등을 마스크에 효율적으로 매핑하여 분할 마스크를 생성한다. 이 디코더는 임베딩을 효과적으로 업데이트하기 위해 Self-attention과 Cross-attention을 두 방향으로 사용하고, 이후 이미지 임베딩을 업샘플링 하고 MLP는 출력토큰을 dynamic linear classifier로 매핑한 후 각 이미지에서 마스크된 전경 확률을 계산한다.

이와 같은 SAM의 아키텍쳐가 모두 실행 되고 나면 프롬프트의 제공 정보에 따라 모호한 결과가 출력 될 수 있다. 이를 해결하기 위해 단일 프롬프트에 대해 총 3가지의 마스크 출력 중 한가지를 사용자가 선택하도록 하여 모호성을 방지한다. 그림 1에선 SAM 아키텍쳐를 보기 쉽게 나타내었다.

SAM model은 다양한 프롬프트를 사용할 수 있다는 것과 속도가 빠르다는 장점이 있지만, 2D 이미지 분할에 국한되어 있고, 프롬프트의 의존성이 높다는 문제가 있다. 또한 마스크의 모호성으로 인해 사용자의 결정이 필요하다는 문제가 있다. 따라서 사용자의 결정 과정을 줄여 분할 하고 싶은 객체를 자동적으로 찾아 분할할 수 있도록 개선을 해보고자 한다.

2. SA3D

SAM은 2D 이미지에서 훌륭한 능력을 가진 vision foundation model이지만 직접적으로 3D 장면으로 확장할 수는 없었다. 따라서 SAM의 파이프라인을 복제하여 3D 장면에서의 분할을 수행하는 방식을 고안하려고 하였다. Segment Anything in 3D (SA3D)[12]에서는 3D 표현 모델을 통해 2D foundation modal에 3D 지각을 제공하는 방식으로 이를 구현하였다. Radiance field는 미분 가능한 렌더링 기술을 사용하여 2D 멀티뷰 이미지를 3D로 구현하는 역할을 하므로 이를 이용하여 SAM을 Radiance field와 통합하였다. 연구에서 Radiance field를 구현하기 위해 사용한 기술은 NeRF이며, 이 기술은 이미지 픽셀값을 계산하기 위해 각각의 뷰에서 특정 픽셀 위치에서 출발한 ray함수 $r(t)$를 계산한다. $r(t)$는 카메라의 위치 $\mathbf{x}_o$, ray의 방향 벡터 $\mathbf{d}$, ray 상의 특정 위치를 나타내기 위한 매개변수 t를 이용하여 계산된다.

$$\mathbf{r}(t) = \mathbf{x}_o + t\mathbf{d} \tag{1}$$

또한 w값은 이 픽셀에서의 해당 픽셀이 NeRF 렌더링 상에서 c라는 색을 가지기 위한 확률 가중치이다. $\sigma(\mathbf{r}(t))$는 $r(t)$에서의 밀도함수 이며, 이는 해당위치에서의 물체의 존재 가능성이다. $\exp(-\int_{t_n}^{t}\sigma(\mathbf{r}(s))\mathrm{d}s)$는 해당경로 상의 빛이 흡수된 정도로, 밀도함수와 투과율을 이용하여 가중치를 구할 수 있다.

$$w(\mathbf{r}(t)) = \exp(-\int_{t_n}^{t}\sigma(\mathbf{r}(s))\mathrm{d}s) \cdot \sigma(\mathbf{r}(t)) \tag{2}$$

이를 이용하여 Radiance field 내에서 Mask 픽셀값을 구할 수 있다. 3D voxel grid $\mathbf{V} \in \mathbf{R}^{L \times W \times H}$를 기반으로 t_n에서 t_f까지의 광선 상의 마스크 값을 구하며, 각 뷰의 2D 마스크의 특정 픽셀은 다음과 같이 렌더링 된다.

$$\mathbf{M}(\mathbf{r}) = \int_{t_n}^{t_f} w(\mathbf{r}(t))\mathbf{V}(\mathbf{r}(t))dt \tag{3}$$

2D 이미지 세트에서 학습된 radiance field가 주어지면 SA3D는 단일 뷰에서 프롬프트를 입력으로 사용하여 해당 뷰에서 SAM으로 2D 마스크를 생성한다. 그런 다음 다양한 뷰에서 Mask inverse rendering과 Cross-view self-prompring을 번갈아가며 수행하여 객체의 3D 마스크를 반복적으로 개선한다.

Mask inverse rendering은 SAM으로 얻은 2D segmentation mask를 density guided inverse rendering을 통해 3D 공간에 projection 되는 단계이다. 최적화를 위해 SAM이 생성한 마스크와 radiance field의 형상의 mask projection loss를 계산한다. SAM의 Segmentation 결과가 항상 정확하지 않기 때문에 loss에 음의 정제항을 추가하여 최적화를 수행한다.

최적화를 수행하면 SAM이 다른 뷰에서 일관되게 전경으로 예측하는 경우에만 SA3D가 전경으로 인식할 수 있다. 이 단계를 거치게 되면 3D 공간에서 개략적인 3D 마스크가 생성된다.

이후 이 3D 마스크를 이용하여 다른 뷰에서의 2D segmentation mask를 추출한다. 해당 마스크는 부정확할 수 있고, 여기서 생성된 몇가지의 포인트 프롬프트들을 SAM에 입력하여 보다 정확하게 2D 마스크를 개선한다. 이 과정이 Cross-view self-prompting 단계이며, 모든 뷰가 샘플링될 때까지 반복적으로 실행하여 3D segmentation 과정을 수행한다.

Ⅲ. 제안된 방법

1. 2D 분할 방법

제안된 3D 객체 분할 기술은 기존 기술[12]과 마찬가지로 입력 시점 영상에 대한 2D 객체 분할 결과를 이용하여 3D grid mask를 생성하게 된다. 따라서 2D 분할 결과가 3D mask 정확도에 큰 영향을 미치게 되며, 이를 고려하여 기존의 주요한 2D 객체 분할 기술인 SAM[3]과 YOLOv8[4]에 대해 성능을 검토하였다.

YOLOv8은 Ultralytics에서 개발되었으며 약 33만장의 이미지와 80개의 클래스를 가진 COCO dataset을 기반으로 학습된다. v8에선 다양한 기능을 가진 YOLO가 개발되었으며, 이중 검출기 출력단에 분할 기능의 레이어를 추가하여 자동으로 개별 Instance 분할이 가능한 모델을 사용하여 분할을 수행하였다. SAM의 경우 Meta AI Research에서 개발되었으며, 사용자 입력 prompt 정보를 이용하여 임의 영상에 대한 영역분할을 수행한다. 본 실험에서는 YOLO를

$$L_{proj} = -\sum_{r \in R(I)} M_{SAM}(r) \cdot M(r) + \lambda \sum_{r \in R(I)} (1 - M_{SAM}(r)) \cdot M(r) \tag{4}$$

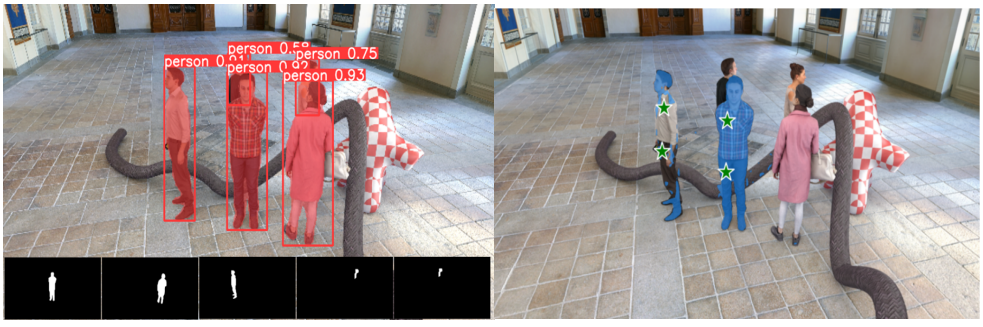

그림 1. 기존 2D 객체 분할 기술 적용 결과. YOLOv8(좌), SAM(우)
Fig. 1. Results of applying existing 2D object segmentation techniques. YOLOv8 (left), SAM (right)

이용하여 객체 분할을 수행하고, 분할된 객체의 세부 조정을 위해 SAM이 사용된다.

그림 1은 MPEG의 MIV[14] 영상에 대해 21시점을 추출하여 학습한 두 방법의 2D 분할 결과이다. YOLOv8의 경우 영상에 포함된 5명의 사람에 대해 각각 instance 검출 및 분할이 되었고, 개별 마스크 획득이 가능하였다. SAM의 경우 초록색 별로 표시된 위치에 사용자 prompt를 입력하여 결과를 얻었으며, 하나의 prompt로는 사람의 전체 영역을 분할하지 못하였고, 전체 영역이 포함될 때까지 prompt를 반복하여 입력해야 하는 문제가 있었으며, 또한 머리카락, 상의, 하의, 신발 등 개별 사람도 서로 다른 특성을 갖는 여러 영역으로 이루어져 있기에, 전체 사람 영역을 분할하는 것은 어려웠다. 다만 YOLO는 자동으로 분할이 이루어지기에 검출이 안되는 경우 해당 영역이 분할되지 않았지만, SAM의 경우 추가 입력을 통해 자동 분할되지 않는 영역을 수작업으로 분할이 가능한 장점이 있었다.

YOLOv8을 이용하여 객체 분할 시, 객체를 구별하기 위한 방법으로는 Instance Matching이 있다. 그림 2의 좌측 사진에서 보듯 5명의 사람에 대한 각각 instance 검출 및 분할이 되면, IoU 최댓값 연산을 수행하여 이전 시점의 특정 인스턴스 마스크 M_i^n과 현재 시점의 인스턴스 마스크 M^n을 비교하는 방식으로 Mask rendering을 수행하게 된다.

$$i^* = \arg\max_{i \in S} IoU(M^n, M_i^n) \tag{5}$$

그림 2는 이를 나타내는 개략적인 흐름도이다. 초기시점에서는 생성된 Mask rendering이 없으므로 사용자가 직접 선택하는 방식으로 생성을 시작한다. 다만 이와 같은 방법은 시점이 연속되어야 정확도가 높아진다는 단점이 있다. YOLO는 2D 객체 분할 기술로 가장 보편화된 기술이므로 다른 객체를 분할하고 싶으면 다른 학습 모델을 이용하기만 하면 된다는 장점이 있다. 이를 이용하여 객체 분할 기술을 연구하면 확장성에도 더욱 좋은 결과를 얻을 수 있을 것이다.

2. 3D 분할 방법

[12]에서 제안된 SA3D 3차원 분할 방법의 경우 먼저 기준이 되는 단일 시점 영상에 대해 사용자 입력 prompt를 받고 SAM을 이용하여 2D 분할 mask를 생성한다. 2D mask는 Mask inverse rendering 과정을 통해 3D grid의 Voxel 값에 반영되고, 이 결과를 가지고 2D mask rendering 과정을 통해 다시 다른 시점의 2D mask를 생성한다. 투사된 2D mask 영역에 대해 cross-view self-prompting 과정으로 수동 입력 없이, 자동으로 prompt를 생성하고 이를 다시

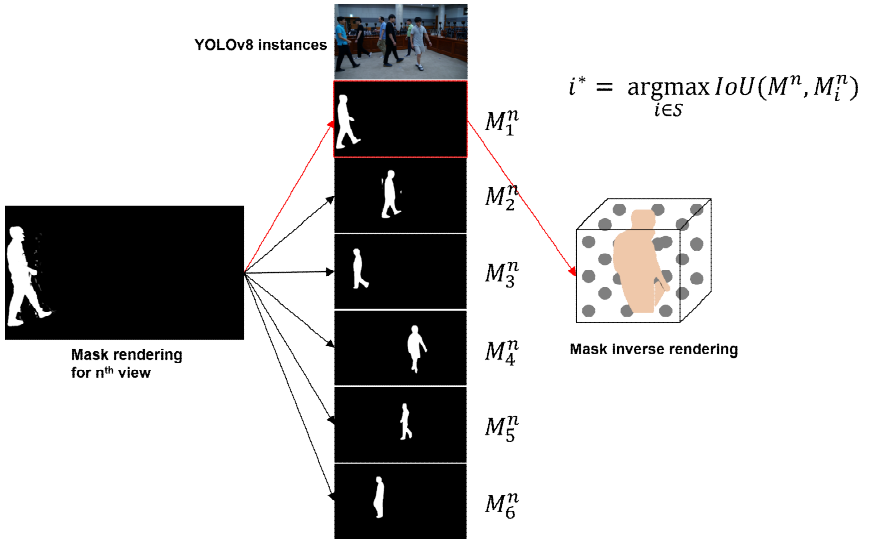

그림 2. YOLOv8 Instance에 3D Mask를 투사하여 최적화를 반영
Fig. 2. Optimizing by projecting a 3D mask onto YOLOv8 instances

SAM에 적용하여 해당 시점의 2D mask를 생성한다. 생성된 결과를 다시 3D grid Voxel에 반영하는 과정을 전체 다시점에 대해 반복하여 최적 3D grid Voxel을 완성한다. 이러한 3D grid Voxel을 임의 시점에 대해 2D rendering을 수행하면, 임의 시점의 객체 분할 결과를 얻을 수 있다.

기존 SA3D는 반드시 기준 시점에 대해 수동으로 prompt를 입력하는 과정이 필요하다. 특히 앞서 살펴본 바와 같이 사람과 같은 객체를 분할 시, 얼굴, 상의, 하의 등 다양한 영역을 포함하도록 prompt를 반복해서 입력해야 하고, 그 경우에도 손, 발 등 끝부분에서 영역이 제외되는 오류가 자주 발생한다. Prompt 입력에 의한 문제는 시점 최적화 과정에서도 발생하는데, 3D grid voxel 최적화 과정 도중 2D 투사된 mask의 정밀도가 낮기에 객체 내부가 아닌 외부에 prompt가 발생할 위험이 있고, 사람이 입력이 아닌 자동으로 prompt 생성시 정밀도가 낮다. 수동 입력보다 분할 오류가 커질 수 있다. 또한 다시점 영상에서 객체가 일부 시점에서만 등장할 경우, 기준 시점에 나타나지 않은 객체는 최적화 도중 검출에서 제외되는 문제가 있다.

기존 SA3D의 문제점을 해결하고자 다음과 같은 3차원 객체 분할 기술을 제안하였다. 제안된 방법에서는 SAM 대신 YOLOv8을 이용하여 기준 시점에 대한 2D mask 생성 시 수동 prompt 과정을 생략하고 자동으로 개별 instance에 대한 2D 분할 결과를 얻었다. 그리고 3D mask 최적화 과정에서도 3D grid voxel의 투사 결과에서 prompt를 추출하지 않고, YOLOv8를 통해 현재 instance의 2D mask를 생성하고 이를 3D mask 최적화에 사용하였다. 이 과정을 통하여 2D mask 투사 과정에서 에러 위치에 추출된 prompt가 사용되는 문제를 제거하였고, 처음 기준 시점에서 나타나지 않은 객체라고 하더라도 중간 시점 최적화 과정에서 검출이 가능하도록 알고리즘을 개선하였다.

Ⅳ. 실험 결과

제안된 3D 객체 분할 방법의 성능 평가를 위해 실험을 수행하였다. 실험데이터는 MPEG MIV에서 제공된 21시점 그래픽 영상을 사용하였다. 그림 4의 좌측은 사용된 영상 데이터 및 획득한 마스크 데이터이다. 제안된 객체 분할 방법을 적용하여 6명의 각 instance에 대해 개별적으로 생성된 3D 마스크를 1번 시점으로 투사한 객체 분할 결과를 나타낸다. 개별 사람에 대한 영역 중복 없이 독립적으로 분할이 되었음을 확인하였다. 하지만 손, 발과 같은 말단 영역에서는 일부 오차가 발생함도 확인할 수 있었다.

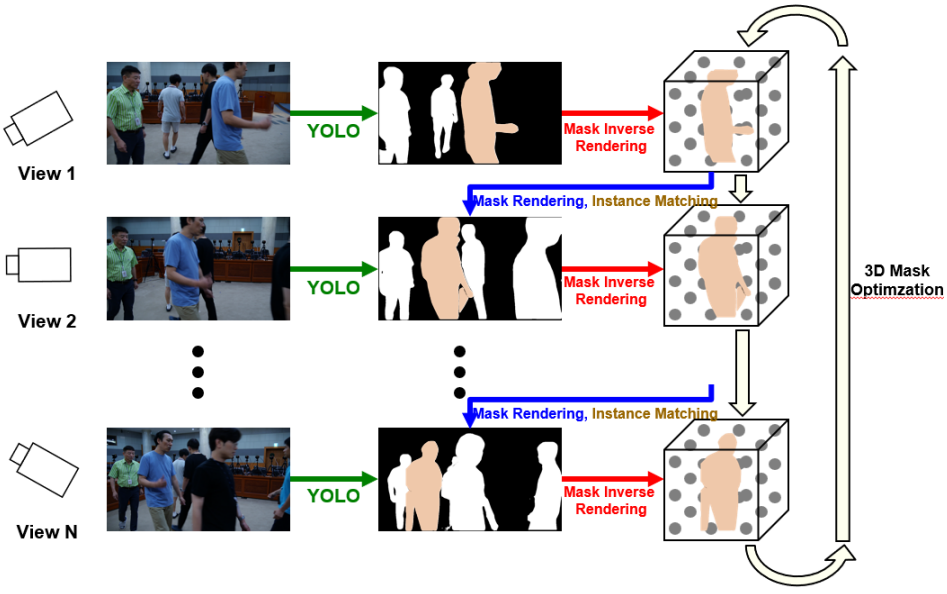

그림 3. 제안된 3차원 객체 분할 기술의 흐름도
Fig. 3. Flowchart of the proposed 3D object segmentation technique

그림 4. 사용된 데이터 및 마스크 (좌), 생성된 임의 시점의 영상과 마스크 (우)
Fig. 4. Data and masks used (left), generated video and masks from arbitrary viewpoints (right)

그림 4의 우측은 NeRF를 통해 생성된 임의 시점 영상과 해당 시점에 대해 3D 마스크를 투사하여 생성한 1번 instance의 객체 분할 결과이다. 가상시점에 대해서도 시점간 consistency를 유지하며 안정적으로 1번 사람 영역이 분할됨을 확인하였다. 특히 대상 객체간에 겹침 (Occlusion)이 발생할 경우에도 해당 영역을 제외하고 분할이 됨을 알 수 있다. 다만 입력 시점에 대한 분할과 다르게 가상시점 분할의 경우 객체 외부의 영역에서 NeRF에 의한 영상 생성 오차가 발생하는 부분에 대해 동일하게 분할 오차가 발생하는 경우가 있었다.

실제 환경에서 제작한 32시점 6 Instance 데이터셋에 대한 객체 분할 또한 수행하였다. 이 데이터셋은 6명의 사람이 자유롭게 이동하는 임의의 시점에서 동시에 32개의 카메라로 촬영하였으며, 전면 부분의 무대를 제외한 3면에 카메라를 일정 간격으로 중앙점을 보도록 배치하였다. 데이터셋에는 사람이 자연스럽게 이동 중에 찍은 것으로 인해 초점이 소실되어 경계면이 정확하지 않은 부분이 존재한다. 이번 실험에서는 초기 시점부터 32시점까지의 객체 분할을 수행하였으며, GT 라벨링도 생성하여 실제 YOLOv8으로 생성한 마스크와 IoU를 비교하였다.

표 1. 32시점 6 인스턴스 데이터셋에 대한 마스크 IoU 비교
Table 1. Mask IoU comparison for a 32-viewpoint, 6-instance dataset

	YOLOv8	SA3D
p1	0.8178	0.7828
p2	0.8039	0.7909
p3	0.8675	0.5724
p4	0.8969	0.9047
p5	0.9054	0.8828
p6	0.9190	0.9303
avg.	0.8684	0.8106

Total 평균 값과 각 객체의 평균 값을 보았을 때, 평균적으로 높은 정확도를 보임을 알 수 있다. 이 값을 개선 전 SA3D 코드와 함께 비교해 보았을 때, SAM의 2D 분할 시 객체 분할 오류 과정을 개선하여 IoU가 개선된 것을 알 수 있다. 또한 3D 분할 시에도 NeRF 렌더링 상의 구조까지 함께 객체로 인식하여 정확도가 크게 떨어지던 문제를 해결하였다. 원본 SA3D를 이용하여 수행한 결과 평균 IoU가 5% 정도 떨어지며, 데이터를 직접 확인한 결과 외곽처리를 제대로 하지 못하는 결과를 확인하였다. 특정 인스턴스에서 객체를 정확히 잡지 못하는 경우에 한해서는 30% 정도의 차이가 나는 경우도 발생하였다.

또한 SA3D의 경우 몇 개의 뷰를 적용하지 않는 누락이 발생하였고, 인물이 아닌 부분을 계속하여 prompt를 이용하여 인식하는 등의 오류가 발생하였다. 이와 같은 부분은 IoU 집계에서도 잘못된 정보로 계산되어 정확도 저하의 원인이 될 수 있다.

그림 5는 데이터에 대한 GT값과, 제안된 방법에 따른 YOLO mask, 원본 코드를 실행하여 나온 mask를 차례로 제시한 결과이다. 프롬프트, 포인트로는 잡을 수 없던 특정

그림 5. 사용된 데이터 원본 (위), 데이터에 대한 참값 (좌하단), YOLO를 이용한 마스크 (중하단), SA3D를 이용한 마스크 (우하단)
Fig. 5. Data Source (Top), Ground Truth (Bottom Left), Mask by YOLO (Bottom Middle), Mask by SA3D (Bottom Right)

개체의 외곽선을 YOLO는 잘 수행한 것을 볼 수 있다. 이를 이용한 3D 객체에서도 주변의 NeRF 구조들이 잘 생성되지 않은 깔끔한 분할 결과를 볼 수 있었다. 다만 외곽선 자체가 부드러워지는 경향이 있어 YOLOv8과 다른 2D 분할 기술들도 추후 연구에서 비교를 해보면 좋을 듯하다.

V. 결 론

본 논문에서는 다시점으로 촬영된 실사 영상으로부터 개별 객체의 영역을 3차원 공간상에서 분할함으로써, 메타버스와 같은 가상환경에서 실감 콘텐츠를 쉽게 저작하는데 적용할 수 있는 기술을 제시하였다. 제안된 기술은 다시점 영상으로부터 3차원 구조 및 외관 정보를 학습할 수 있는 NeRF 기술, 2차원 영상에 대해 개별 instance 검출 및 분할이 가능한 YOLOv8 기술, 다시점 2D 마스크로부터 최적 3D mask를 학습하기 위한 SA3D 기술 등을 활용하였으며, 기존 기술에서 부족했던 사람 영역에 대한 분할 정확도 문제, 개별 instance에 대한 독립적인 최적화 방안, 일부 시점에서 신규로 나타나는 객체에 대한 분할 문제 등을 개선하였다. SAM과 같은 뷰 전체 마스크 방식을 사용하지 않고 각 뷰 당 필요로 하는 객체만 분할하여 메타버스 환경에서 필요로 하는 빠른 3D 분할이 가능하게 되었으며, YOLO 모델이 학습한 객체에 따라 다른 물체까지 분할할 수 있을 것이다. 추후 다양한 데이터에 대해 제안된 방법의 성능을 평가를 진행할 예정이며, 또한 성능면에서 3차원 마스크 최적화 과정에서 시점 간 consistency를 증가시키는 연구를 진행할 예정이다. 또한 최적화 과정에서 다중 입력된 뷰를 재구성하여 정확도를 높이고, 사람의 손, 발 등 말단 영역에서의 분할 정확도 향상을 위한 알고리즘 등에 대해 연구할 계획이다.

참 고 문 헌 (References)

[1] Jonathan Long, Evan Shelhamer, and Trevor Darrell, “Fully convolutional networks for semantic segmentation,” *CVPR*, 2015. doi: https://doi.org/10.1109/CVPR.2015.7298965

[2] Ronneberger, O., Fischer, P., Brox, T., “U-Net: Convolutional Networks for Biomedical Image Segmentation,” In: Navab, N., Hornegger, J., Wells, W., Frangi, A. (eds) *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*.

doi: https://doi.org/10.1007/978-3-319-24574-4_28
[3] Alexander Kirillov et al., “Segment anything,” *arXiv preprint arXiv:2304.02643*, 2023. https://arxiv.org/abs/2304.02643
[4] Dillon Reis et al., “Real-Time Flying Object Detection with YOLOv8,” *arXiv preprint arXiv:2305.09972,* 2023. https://arxiv.org/abs/2305.09972
[5] Weiyue Wang and Ulrich Neumann, “Depth-aware CNN for RGB-D segmentation,” *ECCV*, 2018. doi: https://doi.org/10.1007/978-3-030-01252-6_9
[6] Yajie Xing et al., “Malleable 2.5D convolution: Learning receptive fields along the depth-axis for RGB-D scene parsing,” *ECCV*, 2020. https://arxiv.org/abs/2007.09365
[7] Hang Chu et al., “Surfconv: Bridging 3D and 2D convolution for RGBD images,” *CVPR*, 2018. https://openaccess.thecvf.com/content_cvpr_2018/papers/Chu_SurfConv_Bridging_3D_CVPR_2018_paper.pdf
[8] Nikhil Gosala and Abhinav Valada, “Bird’s-eye-view panoptic segmentation using monocular frontal view images,” *IEEE Robot. Autom. Lett.*, 2022. https://ieeexplore.ieee.org/document/9681287
[9] Jing Huang and Suya You, “Point cloud labeling using 3D convolutional neural network,” *ICPR*, 2016. https://www.cvlibs.net/projects/autonomous_vision_survey/literature/Huang2016ICPR.pdf
[10] Ben Mildenhall et al., “NeRF: Representing scenes as neural radiance fields for view synthesis,” *ECCV*, 2020. https://arxiv.org/abs/2003.08934
[11] Shuaifeng Zhi et al., “In-place scene labelling and understanding with implicit scene representation,” *ICCV*, 2021. https://openaccess.thecvf.com/content/ICCV2021/html/Zhi_In-Place_Scene_Labelling_and_Understanding_With_Implicit_Scene_Representation_ICCV_2021_paper.html
[12] Jiazhong Cen et al., “Segment Anything in 3D with NeRFs,” *NeurIPS,* 2023. https://openreview.net/forum?id=2NkGfA66Ne
[13] Xinhang Liu et al., “Unsupervised Multi-View Object Segmentation Using Radiance Field Propagation,” *NeurIPS* 2022. https://arxiv.org/abs/2210.00489
[14] ISO/IEC DIS 23090-12, Information technology – Coded Representation of Immersive Media - Part 12: MPEG immersive video. https://www.iso.org/standard/79113.html
[15] Dosovitskiy, Alexey. “An image is worth 16x16 words: Transformers for image recognition at scale.” *arXiv preprint arXiv:2010.11929,* 2020. https://arxiv.org/abs/2010.11929

저 자 소 개

정 영 인

- 2024년 2월 : 한국기술교육대학교 메카트로닉스공학부 학사
- 2024년 3월 ~ 현재 : 한국기술교육대학교 메카트로닉스공학과 석사과정
- ORCID : https://orcid.org/0009-0007-3648-1562
- 주관심분야 : AI딥러닝, 컴퓨터비전

이 석

- 2000년 2월 : 서울대학교 기계항공공학부 학사
- 2002년 2월 : 서울대학교 기계항공공학부 석사
- 2007년 2월 : 서울대학교 기계항공공학부 박사
- 2020년 7월 : 삼성종합기술원 수석연구원
- 2022년 2월 : 대통령경호처 경호사무관
- 2022년 3월 ~ 현재 : 한국기술교육대학교 메카트로닉스공학부 조교수
- ORCID : https://orcid.org/0000-0002-2154-4352
- 주관심분야 : 영상처리, 컴퓨터비전, 3차원 디스플레이