Behind the Veil: Enhanced Indoor 3D Scene Reconstruction with Occluded Surfaces Completion

CVPR2024

by

Su Sun, Cheng Zhao, Yuliang Guo, Ruoyu Wang, Xinyu Huang, Yingjie Victor Chen, Liu Ren

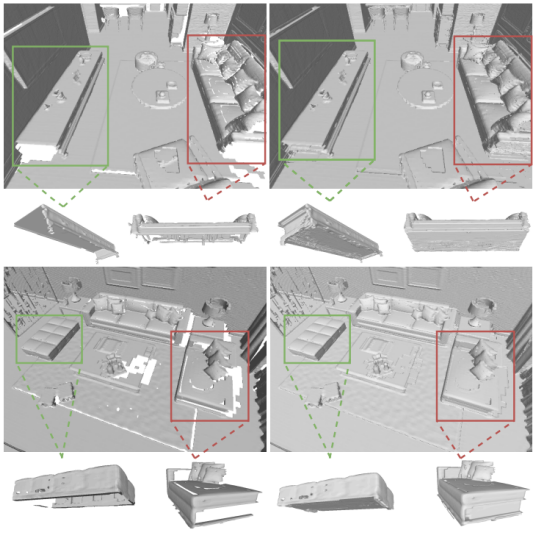
2024.07.04

Index

Index

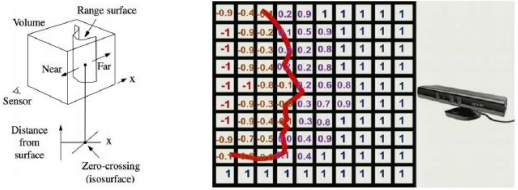
주제

- Interact 할 수 있는 3D scene을 만드는 것은 AR/VR 및 AI 응용에서 중요
- 최근에는 고품질, 고해상도로 3D surface를 재건하는 다양한 방법이 존재
- 3D 공간을 편집 및 재구성할 수 있게 만들어야 함
- 즉, 가구의 뒷면이나 소파의 아래 등 완전한 3D 형태를 보장해야 함
- 본 연구에선 가려진 영역까지 완전히 구성하는 프레임워크를 제시



주제

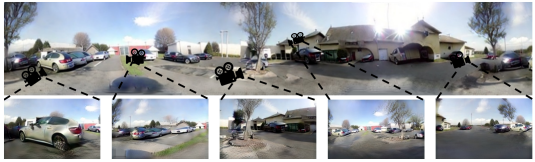
- Voxel 형태로 나누어 surface를 구분하는 TSDF 기반의 reconstruction은 시판 수준의 depth sensor를 사용
  - noise나 가려진 물체 등의 문제로 인해 depth 측정이 정확하지 않음
- NeRF 같은 shape modeling의 방식은 이러한 문제를 어느 정도 방지
  - 그러나 보이는 표면을 주로 나타내기 때문에 가려진 부분을 나타내기 어려움



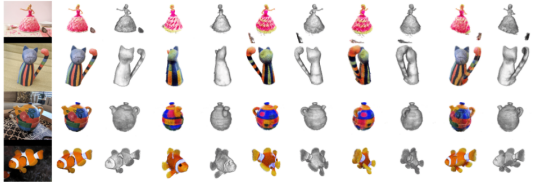
TSDF(Truncated Signed Distance Field) 기반 표현

주제

- NeRF를 기반으로 한 Scene 생성 또한 여러 문제가 있음
  - 단일 객체 수준의 3D Object를 생성하는데 그치거나
  - 편집이 불가능한 형태의 야외 Scene 생성에 초점이 맞춰짐
- 본 연구에선, 작은 모델로 고품질의 Scene geometry를 생성하는 데 초점을 두었음



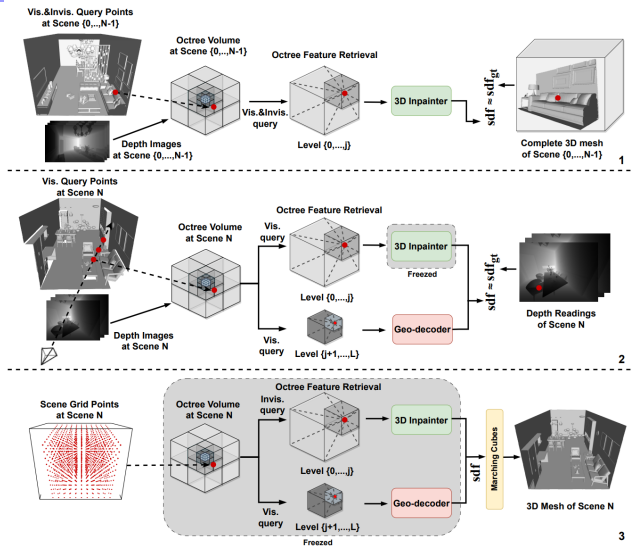
NeRF 기반 야외 Scene 생성



NeRF 기반 단일 Object 생성

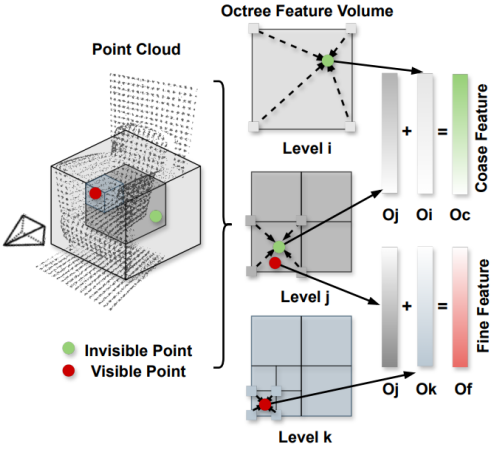
Index

방법


Vis.&Invis. Query Points at Scene {0,..,N-1}
Octree Volume at Scene {0,..,N-1}
Octree Feature Retrieval
Vis.&Invis. query
3D Inpainter
Level {0,...,j}
Complete 3D mesh of Scene {0,..,N-1}
Depth Images at Scene {0,...,N-1}
1
Vis. Query Points at Scene N
Octree Volume at Scene N
Octree Feature Retrieval
Vis. query
3D Inpainter
Freezed
Level {0,...,j}
Geo-decoder
Level {j+1,...,L}
Depth Readings of Scene N
Depth Images at Scene N
2
Scene Grid Points at Scene N
Octree Volume at Scene N
Octree Feature Retrieval
Invis. query
3D Inpainter
Level {0,...,j}
Geo-decoder
Level {j+1,...,L}
Freezed
Marching Cubes
3D Mesh of Scene N
3

방법

- 다양한 시점에서의 depth 이미지를 통해 scene을 point cloud 형태로 표현하고, octree 형태로 나타냄
- level L$(=9)$의 octree를 구성하고, 각 octree의 여덟 코너에 latent feature를 저장함
- 이 코너의 feature들은 random하게 초기화해서 학습을 통해 optimize됨



Coarse-fine 계층 구조

방법

- fine-level의 feature는 세밀한 기하학적 패턴이나 보이는 부분을 구성하는 데 적합하고, Coarse-level의 feature는 가려진 부분을 구성하는 데 적합함
- 그래서 계층 구조를 coarse feature layer과 fine feature layer로 나눔
  즉, octree의 feature O에 대해서,
  $$O = \{O_c, O_f\}, O_c = \{O_i\}_{i=0}^{i=j}, O_f = \{O_j\}_{i=j+1}^{i=L}, j = 4, L = 9$$
- 두개로 나누어진 feature를 concatenate하여 dual-decoder에 적용함
  - O_c는 3D Inpainter로, O_f는 Geo-decoder로
- dual-decoder에서 보이는 표면과 보이지 않는 표면의 SDF값을 각각 추론
- encoder-decoder 아키텍처는 다양한 장면에서의 적용이 제한적기 때문에 decoder-only latent optimization 구조를 사용함

- 각 training scene에서 $p \in \mathbb{R}^3$ 인 임의의 point를 sampling, 이 point는 보이는 점과 보이지 않는 점을 포함 $(p = \{p_v, p_i\})$
- point의 signed distance 값 d_p 를 구하기 위해 p 를 3D Inpainter D_{Ip} 에 넣음

$$d_p = D_{Ip}(f(p), O_c(p))$$

- f 는 position encoding function, O_c 는 coarse octree feature
- 이를 Binary Cross Entropy loss를 사용하여 학습함

$$\mathcal{L}_{bce}(p) = S(d_{gt}) \cdot \log(S(d_p)) + (1 - S(d_{gt})) \cdot \log(1 - S(d_p))$$

  - $S(x) = 1/(1 + e^{x/\sigma})$, σ 를 hyperparameter로 가지는 sigmoid함수
- 3D Inpainter는 Coarse-level의 latent feature을 찾도록 학습되어 가려진 부분의 SDF를 구할 수 있다.

Geo-decoder

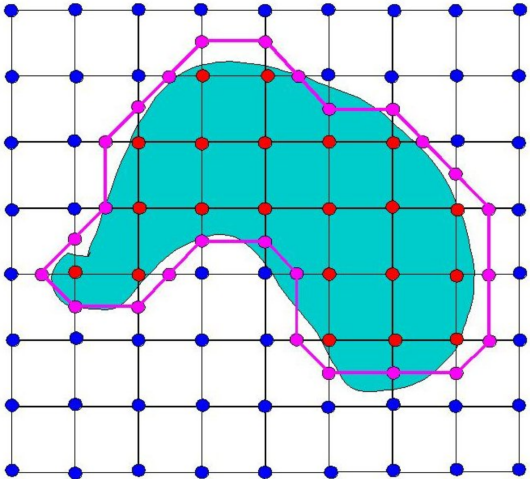
- 입력 이미지가 depth를 가지고 있기 때문에, Geo-decoder를 online으로 최적화 할 수 있다.
- point p_v 를 카메라 광선을 따라 sampling하고, 이 point와 광선의 끝 까지의 거리를 supervision d_{gt} 로 정함
- p_v 는 3D Inpainter와 Geo-decoder를 둘 다 적용함

$$d_{p_v} = D_{Geo}(f(p_v), O_f(p_v))$$
$$d_{p_v} = D_{Ip}(f(p_v), O_f(p_v))$$

- D_{Ip} 의 parameter는 frozen되어서 optimize되지 않음

3D Surface Generation

- Surface 생성 단계에선, online으로 최적화된 Geo-decoder의 SDF와 octree를 기반으로 학습된 3D Inpainter를 사용해 3D mesh를 생성
- 3D 공간에서 균일하게 point를 sampling하고, octree의 모든 layer에서 feature를 찾음
- feature가 threshold $\alpha = 3$ 이상의 layer에서 찾을 수 없었다면 invisible, 아니라면 visible point로 분류
- 각 point를 3D Inpainter와 Geo-decoder에 통과시켜 최종적으로 각 point의 SDF값 $d_p = \{d_{p_v}, d_{p_i}\}$ 을 얻어내고, 이 값을 Marching Cube 알고리즘을 통해 3D mesh를 생성한다.



Marching Cube 알고리즘

Index

결과

- dataset은 AI2-THOR의 3D-CRS와 iTHOR dataset을 사용, real-world dataset은 ScanNet을 사용
- metric은 정확성, 안정성, F1 score을 사용

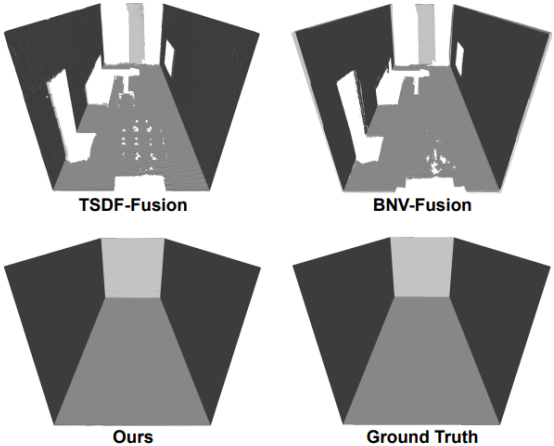
Method	Scene01 Accu./Comp./F1	Scene05 Accu./Comp./F1	Scene06 Accu./Comp./F1	Scene09 Accu./Comp./F1	Scene17 Accu./Comp./F1
TSDF Fusion [28]	84.3/65.8/73.9	92.1/66.2/77.0	79.2/56.0/65.6	81.3/51.6/63.1	86.1/61.3/71.6
Go-Surf [22]	88.6/75.3/81.4	**95.8**/71.3/81.8	**90.5**/71.7/80.0	88.4/64.7/74.7	87.4/71.1/78.4
BNV-Fusion [11]	**93.7**/81.7/87.3	95.2/81.2/87.7	89.7/74.9/81.6	**89.4**/68.1/77.3	**94.4**/76.6/84.6
Ours	91.1/**92.3/91.7**	94.3/**90.0/92.1**	88.0/**89.0/88.5**	88.2/**83.3/85.7**	90.6/**92.0/91.3**

Table 1. Performance comparison between the proposed method and baselines on the 3D-CRS dataset.

Method	FloorPlan207 Accu./Comp./F1	FloorPlan210 Accu./Comp./F1	FloorPlan213 Accu./Comp./F1	FloorPlan220 Accu./Comp./F1	FloorPlan225 Accu./Comp./F1	FloorPlan229 Accu./Comp./F1
TSDF Fusion [28]	80.3/64.2/71.3	86.0/68.9/76.5	85.8/67.9/75.8	80.8/66.4/72.9	78.3/62.0/69.2	76.6/67.4/71.7
Go-Surf [22]	91.0/70.2/79.3	92.2/71.0/80.2	91.0/70.2/79.2	83.8/67.0/74.5	**88.8**/67.5/76.7	86.3/73.1/79.2
BNV-Fusion [11]	**92.0**/71.6/80.5	**93.3**/73.6/82.3	**93.9**/71.9/81.4	**89.1**/70.6/78.8	86.5/69.9/77.3	**93.1**/76.2/83.9
Ours	91.3/**90.1/90.7**	92.1/**92.6/92.3**	89.0/**88.3/88.6**	86.0/**89.7/87.8**	87.1/**88.8/87.9**	87.2/**89.4/88.3**

Table 2. Performance comparison between the proposed method and baselines on the iTHOR dataset.

결과



3D-CRS dataset에서, 가구들을 치운 후의 방 모습

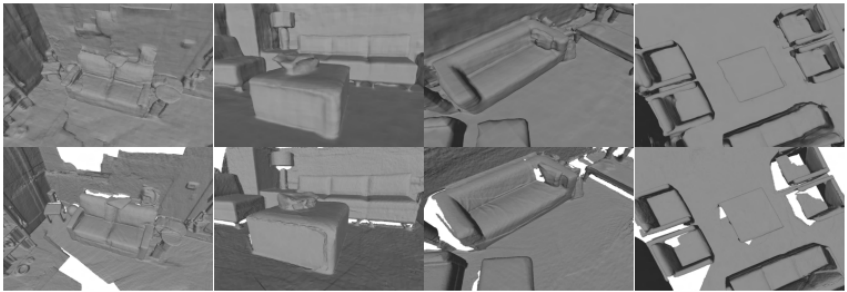


Figure 5. Visualization on ScanNet dataset: row1-Ours, row2-GT

real-world dataset에서의 비교

- ScanNet 자체는 불완전한 유사 GT를 갖고 있어서, 정량적 평가가 불가능했다.

결과

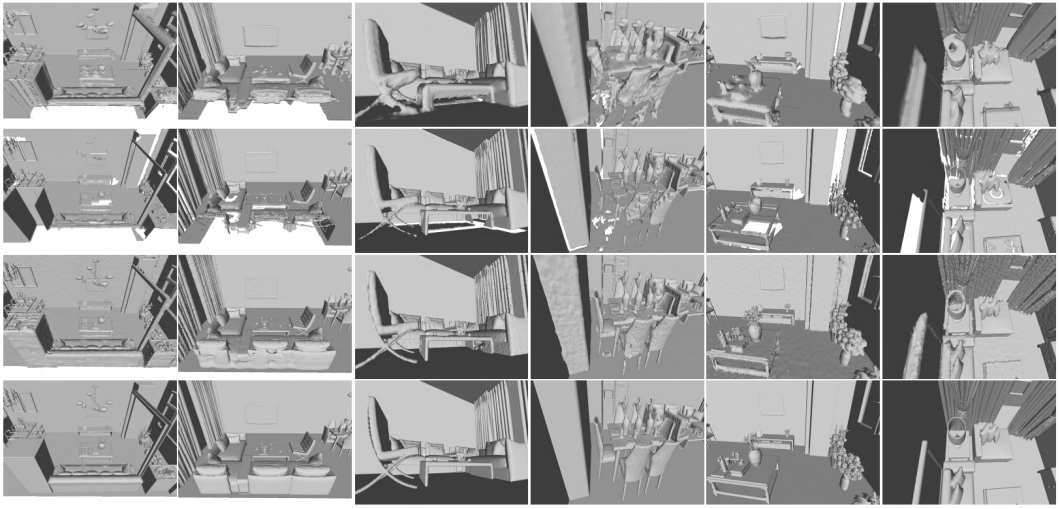


Figure 6. Visual comparison on the 3D-CRS dataset: row1-TSDF Fusion, row2-BNV Fusion, row3-Ours, row4-GT.

결과

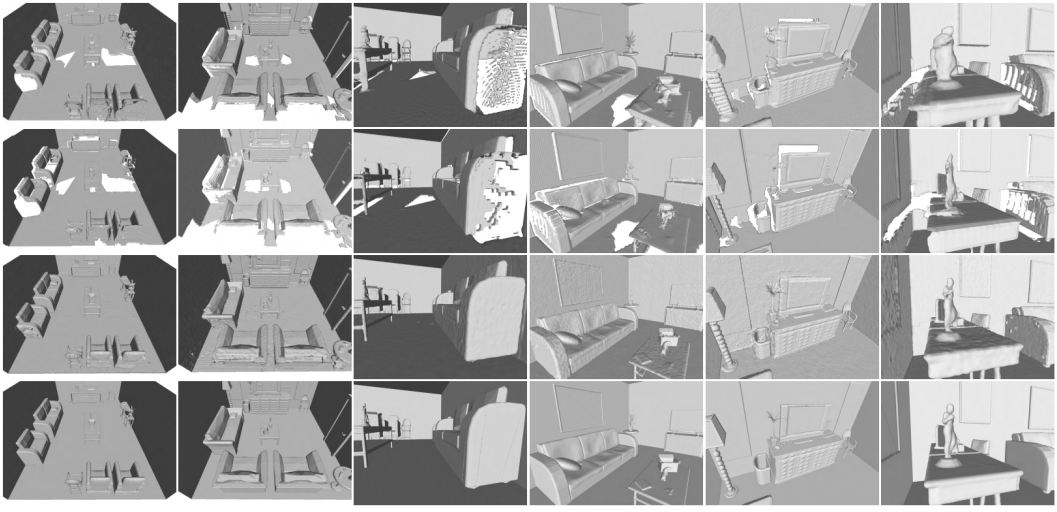


Figure 7. Visual comparison on the iTHOR dataset: row1-TSDF Fusion, row2-BNV Fusion, row3-Ours, row4-GT.

Index

결론

- 보이는 표면과 가려진 표면을 완성하는 새로운 indoor 3D reconstruction 방법을 제시
- hierarchical octree의 coarse fine feature와 이중 디코더 방식으로 이전 연구과 비해 상당한 개선을 보임