2023 여름 세미나

2023.07.21

Sogang University
Vision & Display Systems Lab, Dept. of Artificial Intelligence

Presented By
김지현

Outline

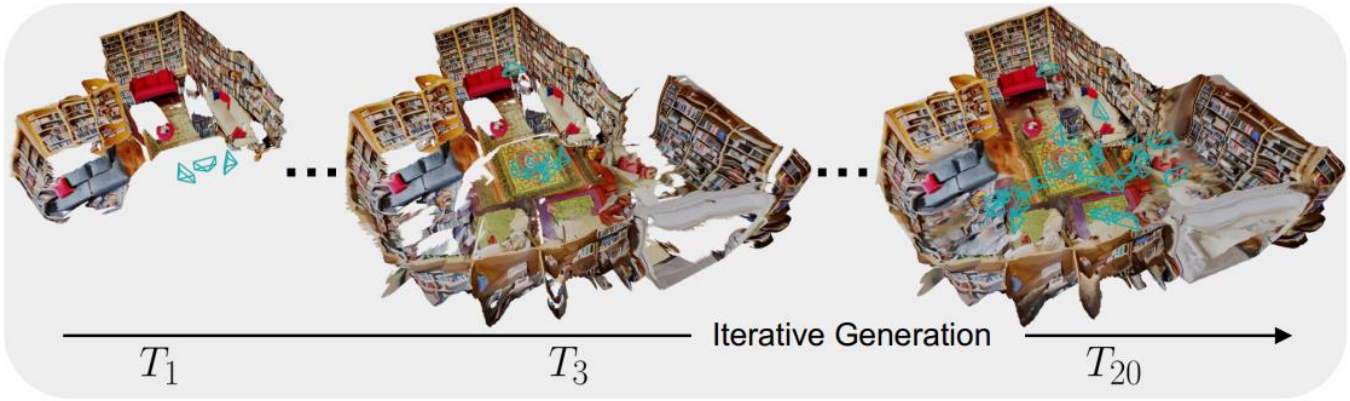
- Paper review
  - Text-guided 3D scene generation
    - Text2Room: Extracting Textured 3D Meshes from 2D Text-to-Image Models (arXiv, 2023)
  - Novel view synthesis
    - Painting 3D Nature in 2D: View Synthesis of Natural Scenes from a Single Semantic Mask (CVPR, 2023)

Task introduction

- Text-guided 3D scene generation
  - Text를 입력하여 3D scene를 생성하는 태스크
    - Object에 국한 되는 것이 아니라 수많은 object와 구조적인 요소로 이루어진 전체 scene의 mesh를 생성한다는 점에서 일반 text-guided 3D generation과 차이
  - 최근 동향
    - 3D text-to-image 모델을 사용하여 3D object (scene X)을 생성
      - 대량의 3D data의 부재로 인하여 특정 domain에 국한
    - 이를 보완하기 위해 2D text-to-image model을 사용하여 3D 생성에 적용시키려는 시도
      - 2D text-to-image의 generality를 이용하여 특정 domain에 국한X
  - 최근 논문
    - 2D text-guided image 모델을 사용하여 3D scene generation을 하는 논문 2편 존재
      - Text2Room: Extracting Textured 3D Meshes from 2D Text-to-Image Models (arXiv, 2023)
      - SceneScape: Text-Driven Consistent Scene Generation (arXiv, 2023)

Paper review

- Text2Room: Extracting Textured 3D Meshes from 2D Text-to-Image Models
  - Method summary



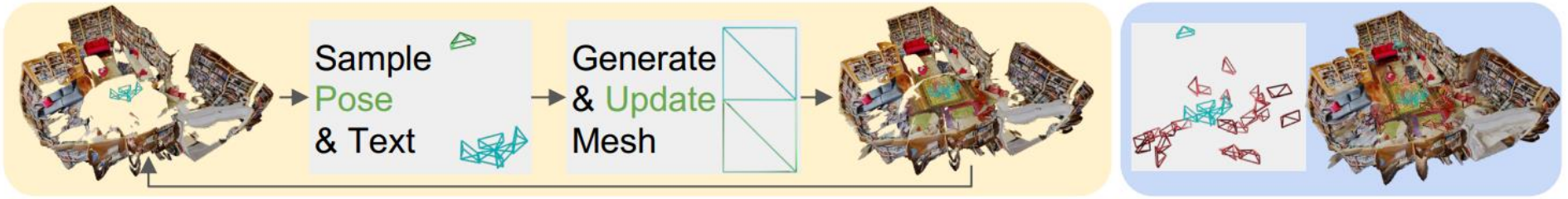
(a) 3D Mesh Generation from Text

(b) Rendered Image + Mesh

- Scene은 iterative한 방식으로 각기 다른 viewpoint에서 생성됨
  - 파란 객체가 viewpoint를 나타냄
  - 최종 output인 3D textured mesh는 texture와 geometry를 포함

Paper review

- Text2Room: Extracting Textured 3D Meshes from 2D Text-to-Image Models
  - Method summary



(a) Scene Generation Stage (b) Scene Completion Stage

- 2단계에 걸쳐서 진행

(a) Scene layout과 object를 생성,

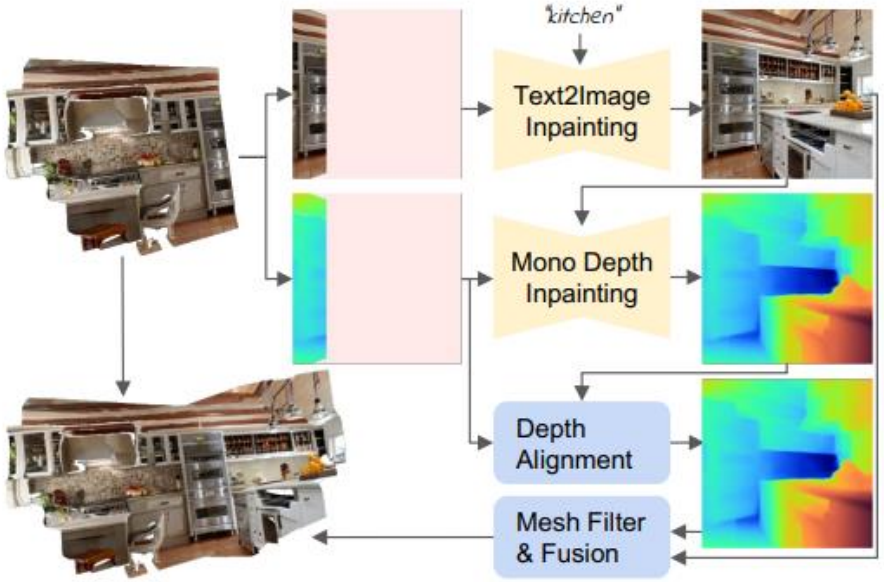
✓카메라의 각 pose는 새롭게 생성된 geometry를 mesh에 추가

(b) 임의로 추가 카메라 포즈를 샘플링하여 3D geometry에서 남은 구멍을 닫음

✓(a)에서 layout과 object를 생성한 뒤의 과정

Paper review

- Text2Room: Extracting Textured 3D Meshes from 2D Text-to-Image Models
  - Iterative scene generation



- Render: 각 카메라 pose에 대하여 현재 mesh로부터 partial RGB와 depth를 렌더링. Text2Image inpainting 모델과 Monocular Depth Inpainting 모델을 사용하여 partial renderings를 완성
- Refine: Depth Alignment와 Mesh Filtering을 통해 최적의 mesh patch를 얻게 되면 이는 기존에 생성된 mesh에 합쳐지게 됨
- Repeat: Iterative하게 render, refine 과정을 반복

Paper review

- Text2Room: Extracting Textured 3D Meshes from 2D Text-to-Image Models
  - Iterative scene generation
    - Scene을 mesh로 표현
      - $\mathcal{M} = (V, C, S)$
        - ✓V = vertices
        - ✓C = vertex colors
        - ✓S = face set
    - 제안 방법의 input은 arbitrary text prompts
      - $\{P_t\}_{t=1}^{T}$
    - 각 arbitrary text prompt는 매칭되는 카메라 포즈 존재
      - $\{E_t z\}_{t=1}^{T}$

Paper review

- T S


```
{
  "prompt": "floor in a single color",
  "negative_prompt": "humans, animals, furniture, couches, chairs, desks, lamps, paintings, seats, decoration, text, distortions",
  "fn_name": "sphere_rot",
  "fn_args": {
    "height": 0.2,
    "radius": 1.0,
    "phi": 35.0
  },
  "adaptive": [
    {
      "arg": "radius",
      "delta": 0.3
    }
  ]
},
{
  "prompt": "floor in a single color",
  "negative_prompt": "humans, animals, furniture, couches, chairs, desks, lamps, paintings, seats, decoration, text, distortions",
  "fn_name": "rot_left_up_down",
  "fn_args": {
    "height": -0.2,
    "rot": -15.0
  }
},

{
  "prompt": "ceiling in a single color",
  "negative_prompt": "humans, animals, furniture, couches, chairs, desks, lamps, paintings, seats, decoration, text, distortions",
  "fn_name": "rot_left_up_down",
  "fn_args": {
    "height": 0.2,
    "rot": 15.0
  }
},
```

Paper review

- Text2Room: Extracting Textured 3D Meshes from 2D Text-to-Image Models
  - Iterative scene generation
    - 각 generation step t마다 새로운 view point에 대해 현재 scene을 렌더링
      - $I_t, d_t, m_t = r(\mathcal{M}_t, E_t)$
        - ✓I_t = rendered image
        - ✓d_t= rendered depth
        - ✓m_t = image-space mask
          - 관찰된 content가 없는 pixel을 표시
        - ✓E_t = selected camera pose
    - Text prompt에 맞추어 관찰되지 않은 pixel들을 inpaint
      - $\hat{I}_t = \mathcal{F}_{t2i}(I_t, m_t, P_t)$
        - ✓$\mathcal{F}_{t2i}$ = text-to-image model
        - ✓$\hat{I}_t$ = inpainted image
        - ✓P_t = text prompt

Paper review

- Text2Room: Extracting Textured 3D Meshes from 2D Text-to-Image Models
  - Iterative scene generation
    - Monocular depth estimator $\mathcal{F}_d$을 사용하여 관찰되지 않은 depth를 align하여 생성

      $\therefore \hat{d}_t = align(\mathcal{F}_d, I_t, d_t, m_t)$

    - 마지막으로 novel content $\{\hat{I}_t, \hat{d}_t, m_t\}$를 기존 mesh와 논문에서 제시하는 fusion scheme으로 합침

      $\therefore \mathcal{M}_{t+1} = fuse(\mathcal{M}_t, \hat{I}_t, \hat{d}_t, m_t, E_t)$

Paper review

- Text2Room: Extracting Textured 3D Meshes from 2D Text-to-Image Models
  - Depth alignment step
    - $\hat{d}_t = align(\mathcal{F}_d, I_t, d_t, m_t)$
    - 2D 이미지를 3D로 변환하기 위해서는 depth 정보가 필요
      - 하지만 각 카메라 포즈마다 동일한 물체에 대해서 다른 depth를 예측할 수 있음
      - 기존 content와 새로운 content를 합칠 때 동일한 object가 비슷한 depth에 위치하도록 aligning하는 과정이 필요
    - Depth alignment를 2단계에 걸쳐서 수행
      - 기존 partial depth를 GT로 취급하여 새롭게 예측한 depth를 align
        - ✓$\hat{d}_p = \mathcal{F}_d(I, d)$
      - Scale, shift 파라미터를 최적화하여 예측한 depth와 rendered depth의 차이를 최소화
        - ✓$\min_{\gamma,\beta} \left\| m \odot \left(\frac{\gamma}{\hat{d}_p} + \beta - \frac{1}{d} \right) \right\|$
        - ✓Aligned depth를 추출
          - $\hat{d} = \gamma \cdot \hat{d}_p + \beta$

Paper review

- Text2Room: Extracting Textured 3D Meshes from 2D Text-to-Image Models
  - Mesh fusion step
    - $\mathcal{M}_{t+1} = fuse(\mathcal{M}_t, \hat{I}_t, \hat{d}_t, m_t, E_t)$
    - Mesh fusion은 4개의 step으로 이루어짐

1. 가장 먼저 이미지의 pixel들을 point cloud로 backproject

$$P_t = \left\{ E_t^{-1} K^{-1} \left(u, v, \hat{d}_t(u,v)\right)^T \right\}_{u=0,v=0}^{W,H}$$

✓K = camera intrinsics

- Camera extrinsics = 카메라의 설치 높이, 방향 등 외부 공간과 관련된 파라미터
- Camera intrinsics = 카메라 초점 거리, 중심점 등 카메라 자체의 내부적인 파라미터

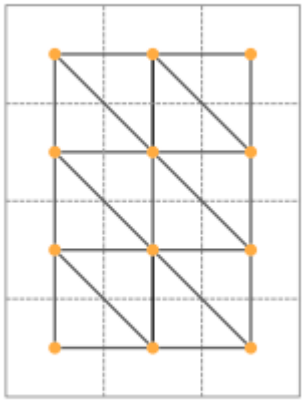
✓u, v = 픽셀의 width, height 위치

Paper review

- Text2Room: Extracting Textured 3D Meshes from 2D Text-to-Image Models
  - Mesh fusion step

    2. 4개의 이웃하는 pixel들로 두 개의 삼각형(face)을 형성하도록 함

       ⁘ $\{(u, v), (u+1, v), (u, v+1), (u+1, v+1)\}$



(a) Pixel Triangulation

Paper review

- Text2Room: Extracting Textured 3D Meshes from 2D Text-to-Image Models
  - Mesh fusion step

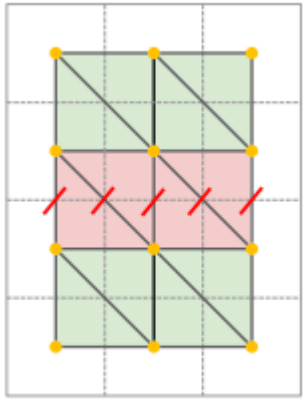
3. Face 필터링

Monocular depth estimator로 예측한 각 픽셀의 depth에 잡음이 많기 때문에 naïve한 triangulation은 3D geometry를 늘어뜨릴 수 있음

따라서 2가지 기준으로 face를 필터링하여 제거함

✓Euclidean distance 기준으로 face edge가 threshold δ_{edge} 보다 클 경우 필터링

✓Viewing direction과 surface normal 사이의 각도가 threshold δ_{sn} 보다 클 경우 필터링



(b) Face Filtering

- $S = \{(i_0, i_1, i_2) | n^T v > \delta_{sn}\}$
- S = face set
- (i_0, i_1, i_2) = 삼각형의 vertex indices
- δ_{sn} = threshold
- n = normalized face normal
- v = world space 상에서 카메라의 중심점으로부터 삼각형의 중심점을 향한 normalized viewing direction

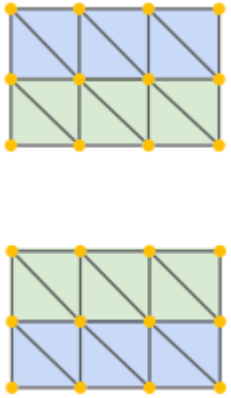
Paper review

- Text2Room: Extracting Textured 3D Meshes from 2D Text-to-Image Models
  - Mesh fusion step

4. Mesh fusion

∴ Inpainted 영역에 해당하는 픽셀로부터 backproject된 pixel로 만들어진 face에 해당

∴ 위 face들은 기존 mesh의 face에 연결됨



(c) Mesh Fusion

Paper review

- Text2Room: Extracting Textured 3D Meshes from 2D Text-to-Image Models
  - Qualitative results

Editorial Style Photo, Modern Living Room, Large Window, Leather, Glass, Metal, Wood Paneling, Apartment

Editorial Style Photo, Modern Nursery, Table Lamp, Rocking Chair, Tree Wall Decal, Wood, Cotton, Faux Fur

Paper review

- Text2Room: Extracting Textured 3D Meshes from 2D Text-to-Image Models
  - Quantitative results

Method	2D Metrics		User Study	
	CS ↑	IS ↑	PQ ↑	3DS ↑
PureClipNeRF [38]	24.06	1.26	2.34	2.38
Outpainting [58, 53]	23.10	1.60	2.90	2.58
Text2Light [11]+Ours	25.99	2.21	2.82	2.97
Blockade [37]+Ours	26.29	2.13	3.35	3.36
Ours w/o alignment	26.73	1.78	3.12	2.96
Ours w/o stretch removal	27.72	1.86	3.28	3.75
Ours w/o completion	27.97	2.18	3.72	3.87
Ours	**28.02**	**2.31**	**4.01**	**4.19**

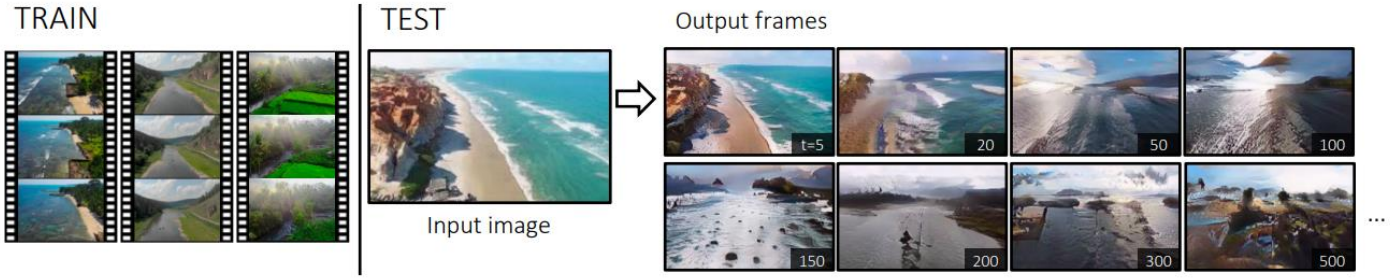
Table 1. **Quantitative comparison.** We report 2D metrics and user study results, including: Clip Score (*CS*), Inception Score (*IS*), Perceptual Quality (*PQ*), and 3D Structure Completeness (*3DS*). Our method creates scenes with the highest quality.

Paper review

- Text2Room: Extracting Textured 3D Meshes from 2D Text-to-Image Models
  - Limitations
    - 임계값 설정
      - 모든 stretched region을 감지하지 못할 수 있으며, 이로 인해 왜곡이 남을 수 있음
    - 논문의 scene representation은 조명으로부터 texture을 분리하지 않으며, 이로 인해 그림자나 밝은 램프와 같은 효과가 포함됨

Task introduction

- Novel view synthesis
  - 기존에 관찰되지 않은 새로운 시점(view point)에서 씬 또는 객체를 합성
    - Single semantic mask를 사용하여 자연 scene의 novel view를 합성하는 특수 환경에 집중
    - Semantic mask를 입력 조건으로 사용함으로써 이를 편집하여 3D contents를 만드는 작업이 가능
  - 기존 방법
    - 2D semantic image synthesis method
      - underlying 3D world를 고려하지 않아 free-viewpoint video를 생성하는 데에 사용될 수 없음
    - Single-view view synthesis method
      - Test시에는 single image만 필요하지만 학습 데이터로는 multi-view images를 요구
      - Test시에는 single image만 필요하지만 보통 아래 그림처럼 학습 데이터로는 multi-view images를 요구하기 때문에 해당 논문 task에는 기존 방법 적용이 어려움

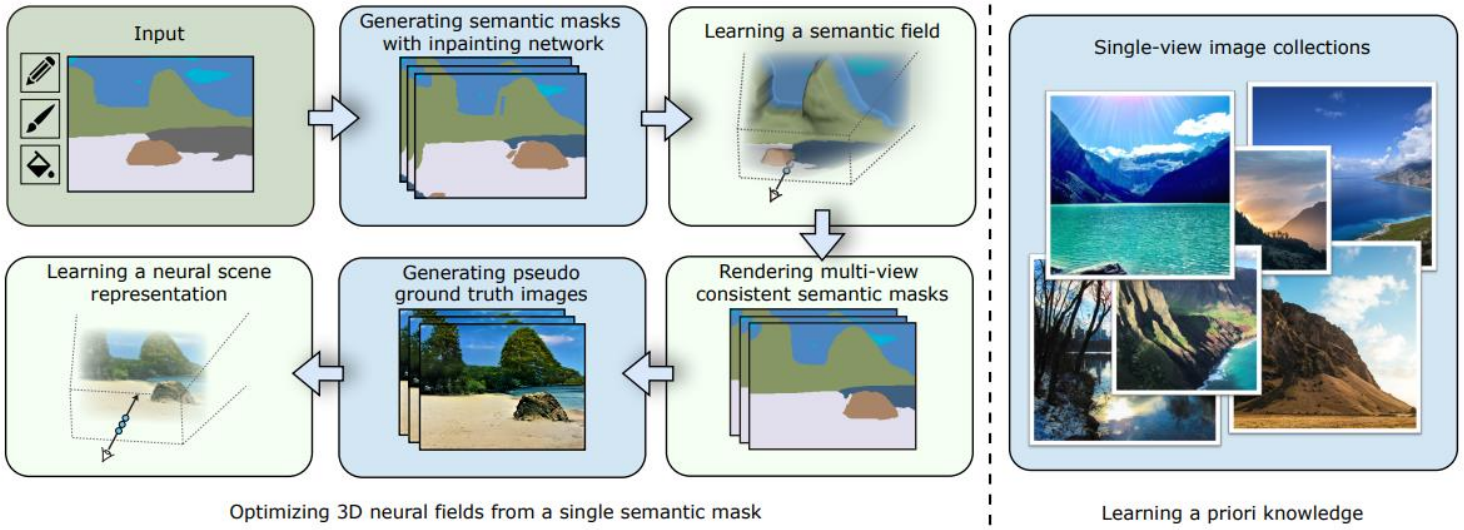


Paper review

- Painting 3D Nature in 2D:View Synthesis of Natural Scenes from a Single Semantic Mask
  - Summary
    - Single-view image 데이터셋으로 학습이 가능한 semantics-guided view synthesis of natural scene 프레임워크를 제안
      - Task를 더 간단한 task 2개로 구분
        - ✓Novel view의 semantic mask를 생성
          - 입력 semantic mask를 SPADE 모델을 사용하여 RGB 이미지로 변환
          - Depth estimator을 사용하여 RGB 이미지의 depth map을 추정
          - 추정된 depth map을 사용하여 입력 semantic mask를 novel view로 warp
        - ✓SPADE 모델을 사용하여 생성된 semantic mask를 RGB 이미지로 변환
    - SPADE 모델을 사용하여 생성된 semantic mask를 RGB 이미지로 변환
      - Inpainting model로 생성한 semantic mask는 view inconsistent한 경향이 있음
      - Views마다 다른 사소한 차이도 content 상에서 큰 차이를 만들어낼 수 있기 때문에 이를 해결하기 위함

Paper review

- Painting 3D Nature in 2D:View Synthesis of Natural Scenes from a Single Semantic Mask
  - Pipeline



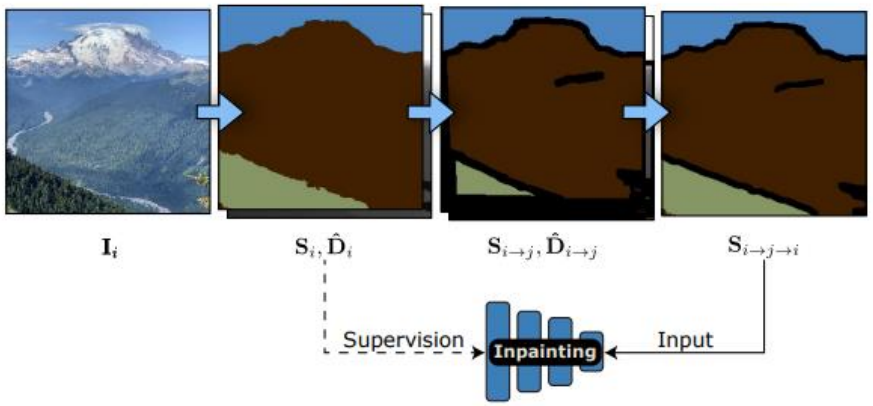
- Pipeline을 2단계로 구분
  - view-consistent semantic mask를 생성
  - PADE를 사용하여 multiview semantic mask를 컬러 이미지로 변환하고, view-consistent 렌더링을 위해 neural scene representation을 복구

Paper review

- Painting 3D Nature in 2D:View Synthesis of Natural Scenes from a Single Semantic Mask
  - Generating view-consistent semantic masks
    - Depth-based warping technique 사용
      - 입력 semantic map을 SPADE 를 사용하여 RGB 이미지로 변환
      - monocular depth estimation network를 사용하여 생성된 RGB 이미지의 depth map을 예측
      - 예측된 depth에 기반하여 3D triangular mesh 생성
      - semantic mask는 mesh로 변환되어, 해당하는 semantic label로 색상을 할당
      - 생성된 3D triangular mesh는 depth map의 불연속성으로 인해 가짜 엣지를 포함. 이를 해결하기 위해, 서로 먼 vertices를 가진 엣지를 제거
      - 생성된 3D triangular mesh는 depth map의 불연속성으로 인해 가짜 엣지를 포함. 이를 해결하기 위해, 서로 먼 vertices를 가진 엣지를 제거

Paper review

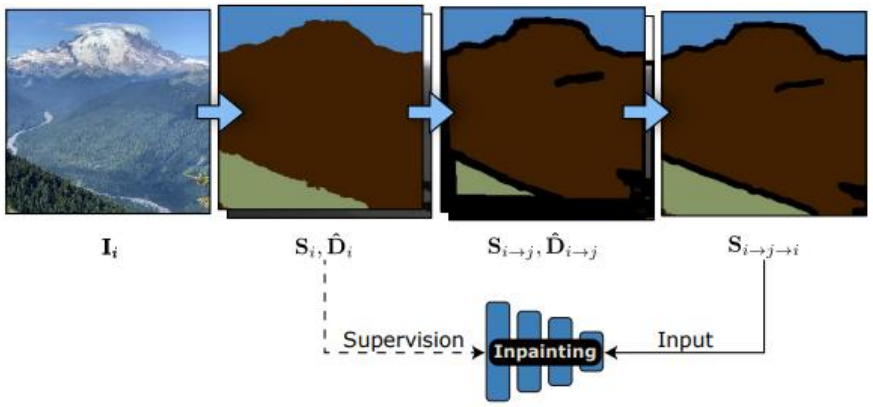
- Painting 3D Nature in 2D:View Synthesis of Natural Scenes from a Single Semantic Mask
  - Semantic mask inpainting



- 누락된 영역을 inpaint하기 위해 self-supervised 기법을 사용하여 single-view natural images로 semantic inpainting network를 훈련
  - 사전 훈련된 image segmentation model과 monocular depth estimation model을 사용하여 natural images의 semantic mask와 depth map 생성
  - 각 training iteration마다, source image I_i를 데이터셋에서 무작위로 샘플링
  - Depth map D^i를 사용하여 원본 semantic mask S_i를 random target view j로 warping하여 warped semantic mask $S_{(i\to j)}$와 target view j에서의 depth map $D^{(i\to j)}$를 생성
  - Semantic mask $S_{(i\to j)}$ 를 다시 source view로 depth map $D^{(i\to j)}$ 를 사용하여 warping하여 구멍이 있는 semantic mask $S_{(i\to j\to i)}$를 생성

Paper review

- Painting 3D Nature in 2D:View Synthesis of Natural Scenes from a Single Semantic Mask
  - Semantic mask inpainting



- 누락된 영역을 inpaint하기 위해 self-supervised 기법을 사용하여 single-view natural images로 semantic inpainting network를 훈련
  - 마지막으로, semantinc inpainting network에 $S_{(i\to j\to i)}$를 입력하고 훈련시켜 구멍을 inpainting. 원본 semantic mask S_i 와 함께 지도 학습
- 테스트 시에는 무작위로 viewpoint set을 샘플링한 후, 주어진 semantic mask S_0 를 이 viewpoints로 warping하여 warped semantic mask를 생성. 그 다음 inpainting network를 사용하 disoccluded 영역을 inpainting

Paper review

- Painting 3D Nature in 2D:View Synthesis of Natural Scenes from a Single Semantic Mask
  - Semantic field
    - Inpainted semantic mask가 view-consistent하지 않을 경우 SPADE로 생성한 이미지에서 큰 artifacts가 생성됨
      - 이 문제를 해결하고자 semantic field를 도입하여 inpainted semantic mask를 융합하고 노이즈를 제거
    - 3D scene의 semantics와 geometry를 표현하기 위해 도입
    - Continuous neural field에서는 MLP 네트워크 f_θ는 3D 공간에서의 임의의 쿼리 포인트 x를 SDF 값 d와 중간 피쳐 z로 매핑. 그 다음 다른 MLP 네트워크 f_φ가 intermediate feature z를 semantic logit s로 매핑
      - Signed distance field (SDF) = 점으로부터 가장 가까운 표면까지의 거리를 나타내는 값

$$f_\theta : \mathbf{x} \in \mathbb{R}^3 \mapsto (d \in \mathbb{R}, \mathbf{z} \in \mathbb{R}^c)$$
$$f_\phi : \mathbf{z} \in \mathbb{R}^c \mapsto \mathbf{s} \in \mathbb{R}^{M_s}, \qquad (1)$$

Paper review

- Painting 3D Nature in 2D:View Synthesis of Natural Scenes from a Single Semantic Mask
  - Semantic field
    - 최종 semantic probability는 아래 식 (2)와 같이 정의됨
      - Foreground와 sky는 매우 멀리 떨어져 있기 때문에 별도로 처리
      - Sky는 먼 2D plane으로 가정되며 semantic probobality는 일정한 원핫 벡터 P_sky로 정의됨
      - T_fg는 카메라 광선 r을 따라 foreground의 누적 투과도(transmittance)이며, P_fg는 렌더링된 semantic logit에 softmax 레이어를 적용하여 얻은 semantic probability이다.

$$\mathbf{Y}(\mathbf{r}) = \mathbf{P}_{\text{fg}}(\mathbf{r})T_{\text{fg}}(\mathbf{r}) + (1 - T_{\text{fg}}(\mathbf{r}))\mathbf{P}_{\text{sky}}, \qquad (2)$$

Paper review

- Painting 3D Nature in 2D:View Synthesis of Natural Scenes from a Single Semantic Mask
  - Semantic field
    - Training loss

Semantic field. To optimize the semantic field, we use the following overall loss function:

$$\begin{aligned}\mathcal{L} =& \lambda_0 \mathcal{L}_{\text{depth}} + \lambda_1 \mathcal{L}_{\text{trans}} + \lambda_2 \mathcal{L}_{\mathbf{P}} + \lambda_3 \mathcal{L}_{\text{eikonal}} \\ &+ \lambda_4 \mathcal{L}_{\text{rank}} + \lambda_5 \mathcal{L}_{\text{src}},\end{aligned} \tag{9}$$

where $\lambda_0 = 0.1, \lambda_1 = 10, \lambda_2 = 1, \lambda_3 = 0.01, \lambda_4 = 0.1, \lambda_5 = 1$. The $\mathcal{L}_{\text{depth}}$, $\mathcal{L}_{\text{trans}}$ and $\mathcal{L}_{\mathbf{P}}$ are defined in the main paper.

Paper review

- Painting 3D Nature in 2D:View Synthesis of Natural Scenes from a Single Semantic Mask
  - Semantic field
    - Cross entropy loss
      - Rendered semantic probability P(r)와 infilled semantic mask의 semantic probability P*(r)의 cross entropy loss로 semantic field를 학습

$$\mathcal{L}_{\mathbf{P}} = -\sum_{\mathbf{r}\in\mathcal{R}}\sum_{k=1}^{M_s}\mathbf{P}_k^*(\mathbf{r})\log\mathbf{P}_k(\mathbf{r}). \qquad (3)$$

    - Scale-and shift-invariant loss
      - Rendered dapth map D와 predicted depth map D^의 차이를 계산하기 위해서 사용
        - ✓R'은 camera rays of image pixels (sky region을 제외한)을 의미
        - ✓w과 q는 scale과 shift를 scaling하기 위해서 사용

$$\mathcal{L}_{\text{depth}} = \sum_{\mathbf{r}\in\mathcal{R}'}\|(w\mathbf{D}(\mathbf{r})+q)-\hat{\mathbf{D}}(\mathbf{r})\|^2, \qquad (4)$$

Paper review

- Painting 3D Nature in 2D:View Synthesis of Natural Scenes from a Single Semantic Mask
  - Natural scene representations
    - Multi-view masks를 바로 RGB 이미지로 변환하는 것은 multi-view consistent한 이미지를 생성하지 못함
    - 이를 해결하기 위해 SPADE로 부터 얻은 appearance information을 fuse할 수 있는 natural scene representation을 학습
    - Appearance field: "To represent the scene's appearance, we recover an appearance field"

$$f_\xi : \mathbf{x} \in \mathbb{R}^3 \mapsto \mathbf{c} \in \mathbb{R}^3$$

- Appearance field에서 point x는 tri feature planes에 orthogonally하게 projected되어 3개의 feature vectors가 얻어짐. 이러한 feature vectors는 합쳐져 final feature vector가 얻어짐
- MLP(다중 퍼셉트론) 네트워크를 사용하여 aggregated feature vector에서 RGB 값을 회귀

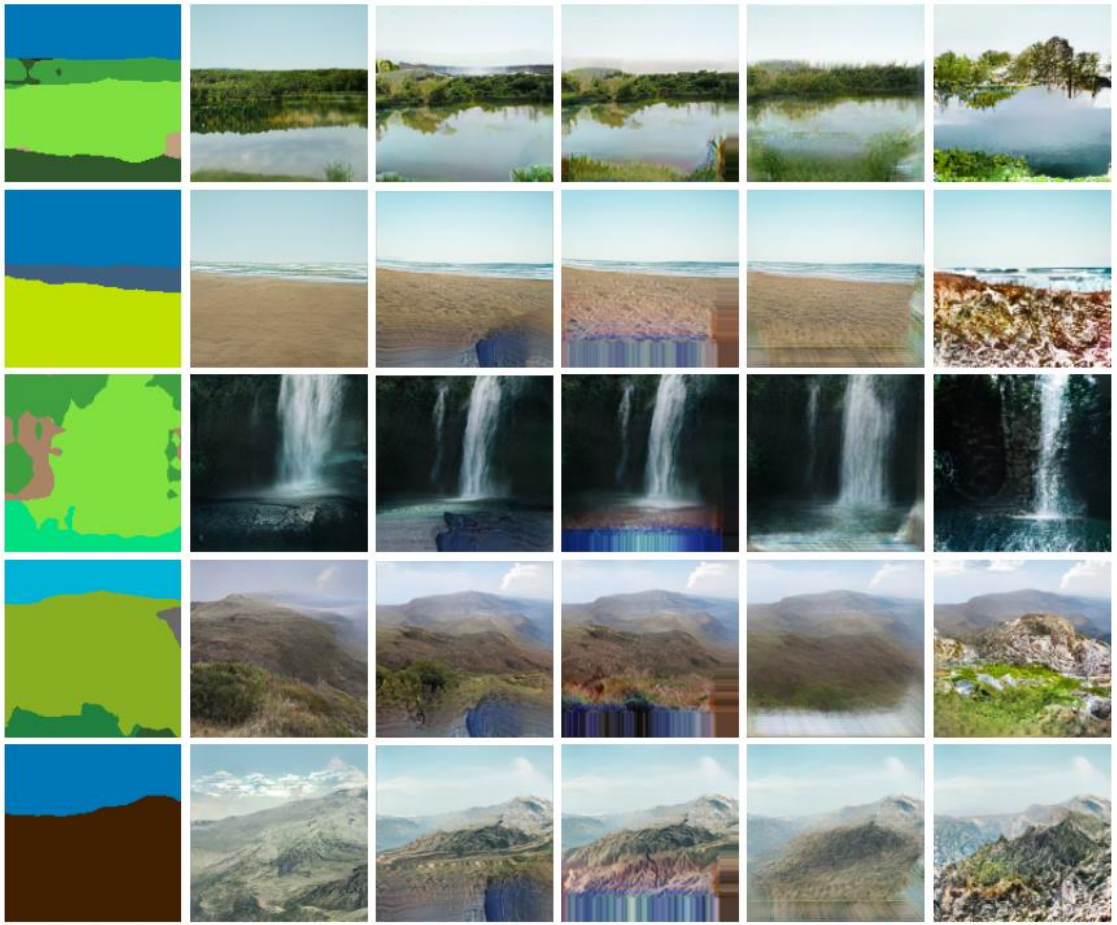
Paper review

- Painting 3D Nature in 2D:View Synthesis of Natural Scenes from a Single Semantic Mask
  - Natural scene representations
    - 학습 중에서 geometry network fθ가 고정되고 appearance network는 perceptual loss와 adversarial loss로 최적화됨
      - 먼저 학습된 semantic field를 사용하여 multi-view semantic mask를 렌더링한 뒤 이를 SPADE로 이미지로 변환.
    - Adversarial loss에서는 변환된 이미지 C가 "fake" 샘플, 생성된 이미지 C^가 "real" 샘플이 된다.
      - semantic field에서 나온 mask를 바로 SPADE로 변환한 이미지를 변환된 이미지 (rendered image) 그리고 appearance field에서 나온 feature을 사용하여 MLP로 생성한 이미지를 생성된 이미지 (generated image)
    - 변환된 이미지 C와 생성된 이미지 C^를 비교하기 위해 perceptual loss가 사용됨.

$$\mathcal{L}_{\text{feat}}\left(\hat{\mathbf{C}}, \mathbf{C}\right) = \left\| \phi(\hat{\mathbf{C}}) - \phi(\mathbf{C}) \right\|_2^2, \qquad (7)$$

Paper review

- Painting 3D Nature in 2D:View Synthesis of Natural Scenes from a Single Semantic Mask
  - Qualitative results



Paper review

- Painting 3D Nature in 2D:View Synthesis of Natural Scenes from a Single Semantic Mask
  - Quantitative results

Methods	FID↓	KID↓
SPADE [52]+InfiniteNatureZero [36]	149.80	0.080
SPADE [52]+3DPhoto [63]	127.74	0.064
SPADE [52]+AdaMPI [20]	185.96	0.115
GVS* [19]	141.64	0.084
Ours	**109.85**	**0.050**

Table 1. **Quantitative comparisons on the LHQ dataset.** "SPADE + *" means a two-stage pipeline that first generates an image with SPADE and then performs single-view view synthesis. "GVS*" means that we train GVS on the LHD dataset using the strategy in AdaMPI.