Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Computer Vision Lab

NPSim: Nighttime Photorealistic Simulation From Daytime Images With Monocular Inverse Rendering and Ray Tracing

Shutong Zhang
Department of Information Technology and Electrical Engineering

Supervisor: Dr. Christos Sakaridis, Prof. Luc Van Gool

Table of Contents

- Motivation
- Related Works
- Method
  - Data Collection
  - Geometry Mesh Reconstruction
  - Realistic Nighttime Scene Relighting
- Experiment
- Conclusion

Motivation

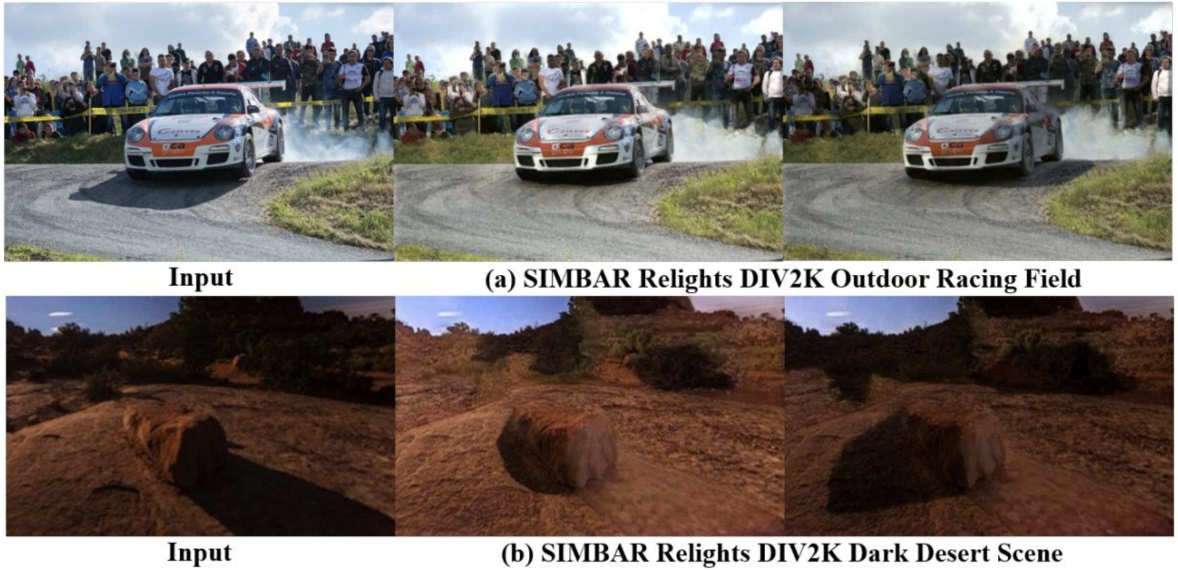
- Poor performance of current semantic segmentation methods on nighttime images.
- Nighttime images are hard to annotate.

Method	Fog	Night	Rain	Snow	All
RefineNet [24]	63.6	52.2	66.4	62.5	62.8
DeepLabv2 [5]	52.2	45.4	57.6	56.8	54.9
DeepLabv3+ [6]	68.7	59.2	73.5	70.5	69.6
HRNet [47]	70.8	63.2	72.7	70.2	70.9
RefineNet [24]	65.7	55.5	68.7	65.9	65.3
DeepLabv2 [5]	54.5	45.3	59.3	57.1	55.3
DeepLabv3+ [6]	69.1	60.9	74.1	69.6	70.0
HRNet [47]	74.7	65.3	77.7	76.3	75.0

Method	Fog	Night	Rain	Snow
Source model	33.5	30.1	44.5	40.2
AdaptSegNet [43]	31.8	29.7	49.0	35.3
ADVENT [46]	32.9	31.7	44.3	32.1
BDL [23]	37.7	33.8	49.7	36.4
CLAN [26]	39.0	31.6	44.0	37.7
FDA [53]	39.5	37.1	53.3	46.9
SIM [48]	36.6	28.0	44.5	33.3
MRNet [62]	38.8	27.9	45.4	38.7
Oracle	52.2	45.4	57.6	56.8

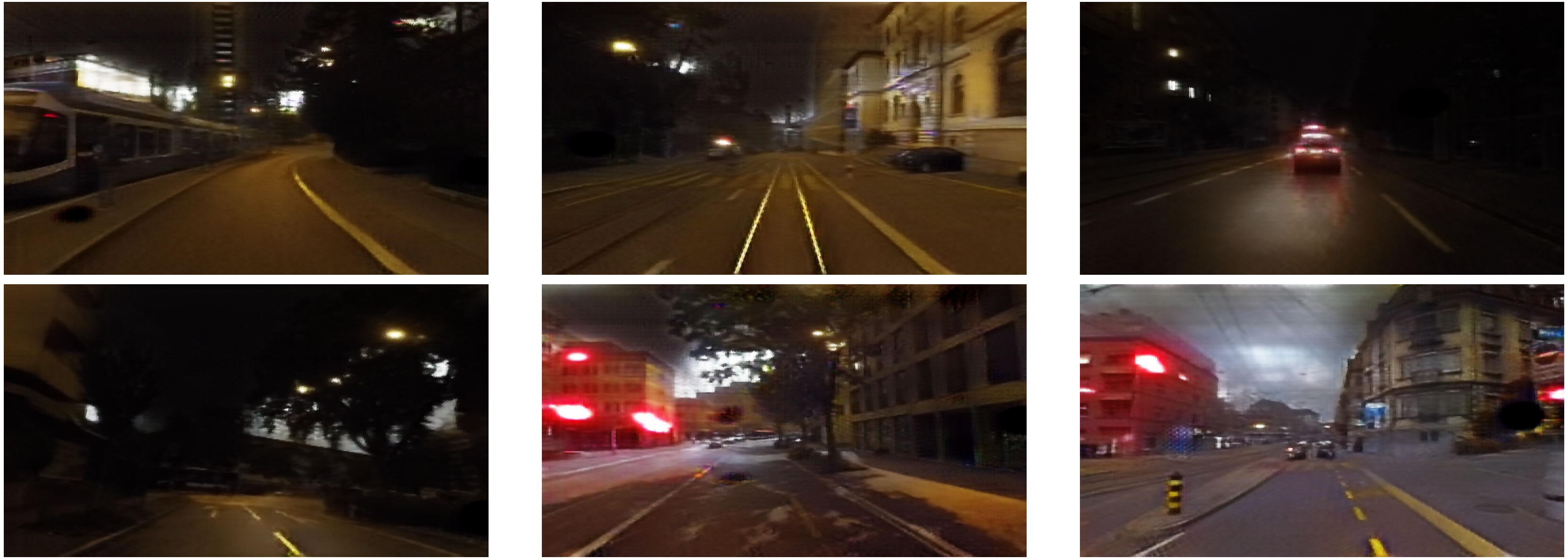
Related Works

- SIMBAR: Single Image-Based Scene Relighting For Effective Data Augmentation (Zhang et al. 2022)
    - Daytime Relighting: consider sun as only light source



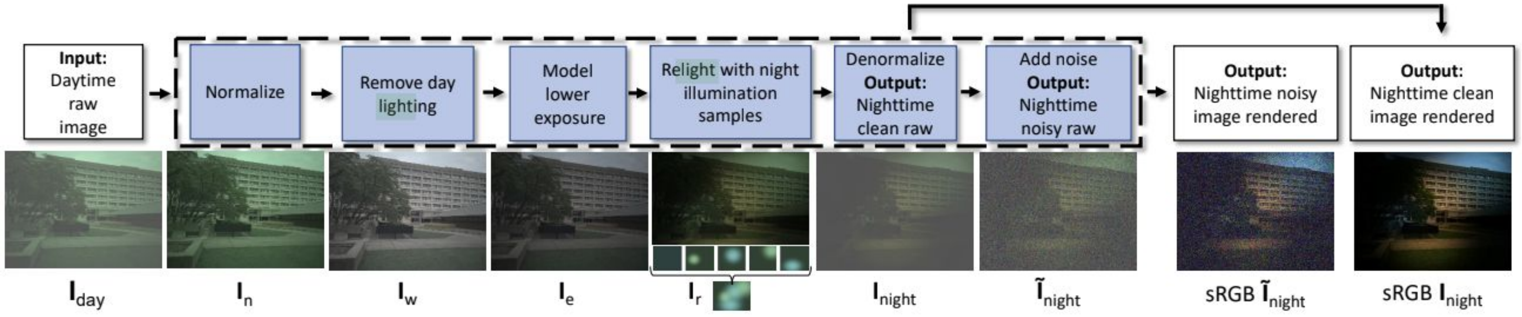
Related Works

- CycleGAN: Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks (Zhu et al. 2017)
    - Not able to accurately activate light sources result in unrealistic images

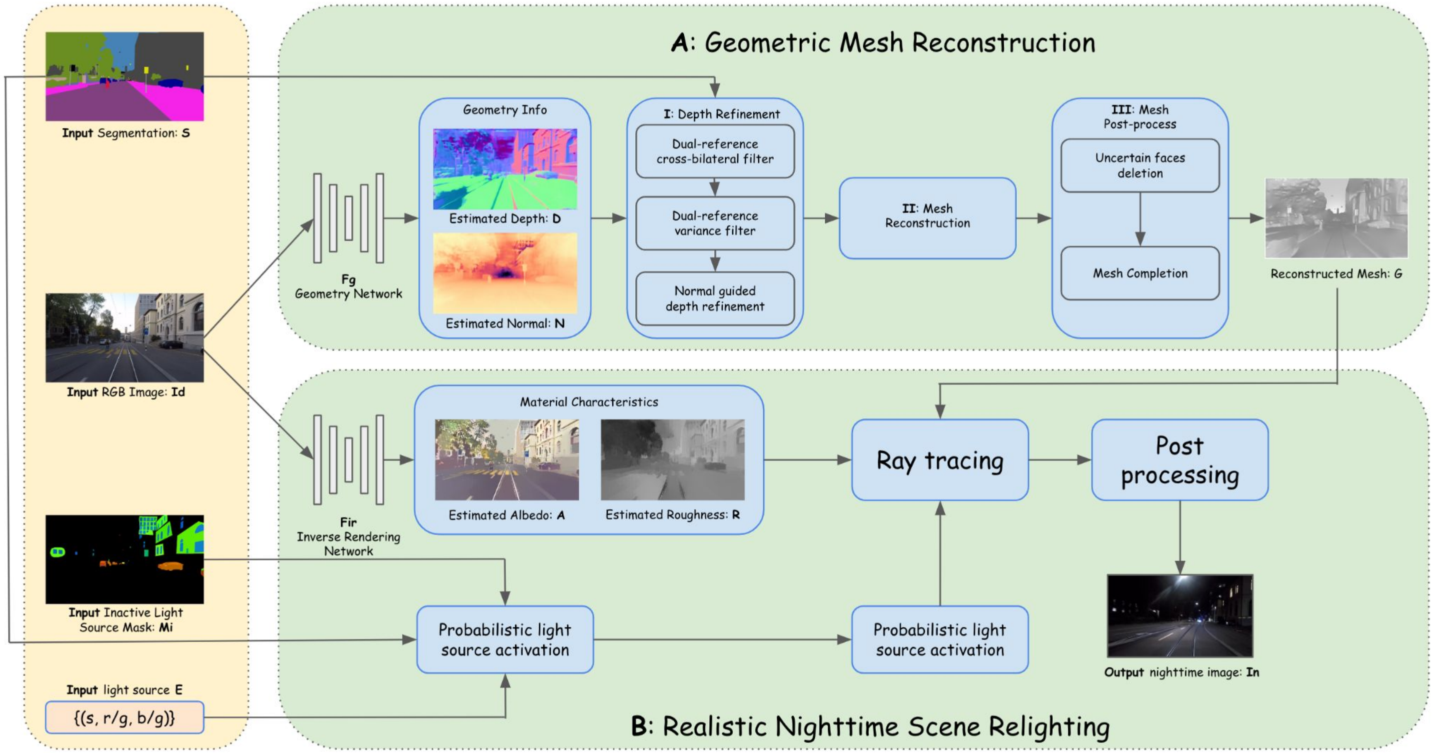


Related Works

- Day-to-Night Image Synthesis for Training Nighttime Neural ISPs (Punnappurath et al. 2022)
    - Consider light source as 2D
    - Result in dimmed daytime image

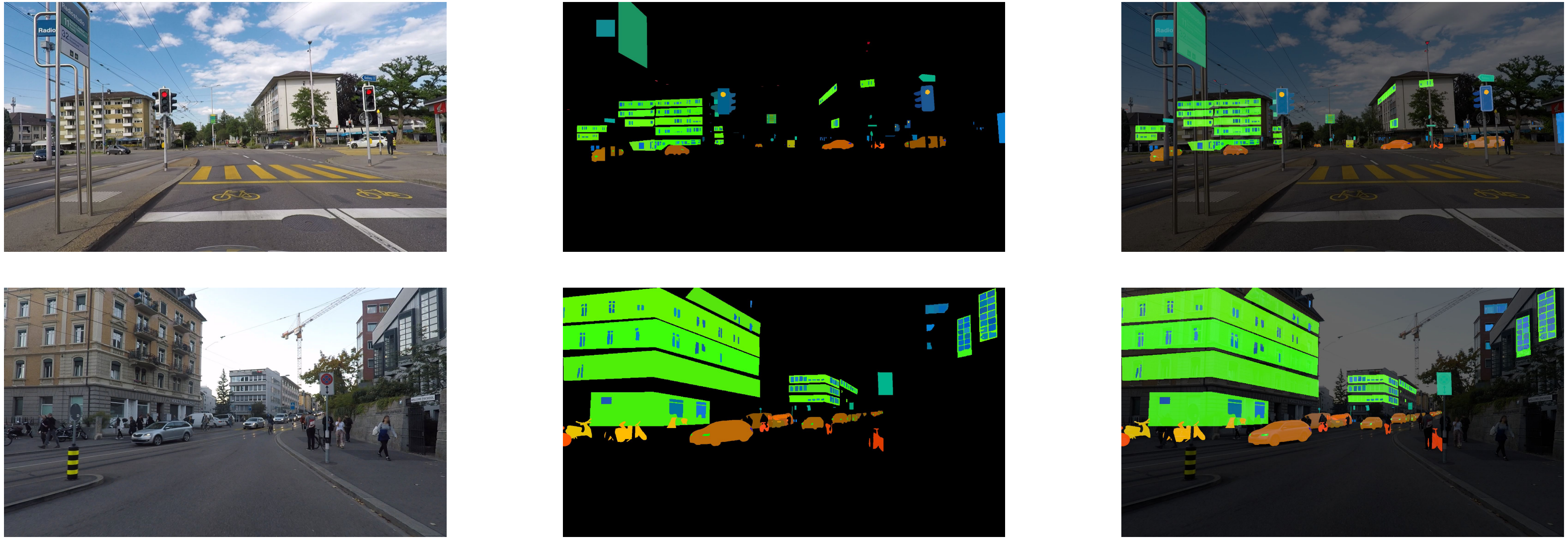


Method - Overview



Method - Data Collection

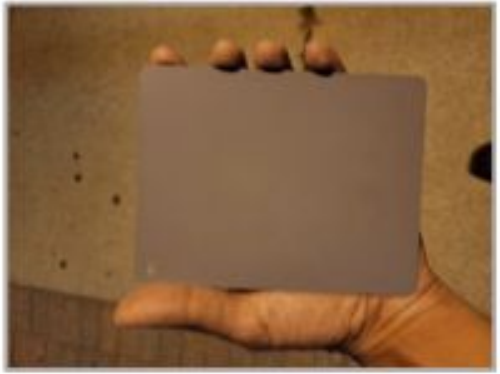
- Inactive light source mask $\mathrm{M_i}$
  - Defined 21 classes of invalid light sources, including: building, vehicle, object and group
  - Manual annotation using Segments.ai



Input RGB   Binary mask   Superposition

Method - Data Collection

- Light source $\mathbf{E}$
  - Capture from real world using gray card
  - Extract both chromaticity and strength value: {(s, r/g, b/g)}


Method - Geometry Mesh Reconstruction

- $\mathbf{F_g}$: Learning based method iDisc to estimate depth and surface normals
  - Depth: pre-trained iDisc depth model on KITTI dataset
  - Surface normal: retrain iDisc model on DIODE dataset (outdoor depth)

Method - Geometry Mesh Reconstruction

- **Reconstruction**: Worldsheet to reconstruct mesh based on depth
  - Mesh grid depends only on depth and fixed offset $\Delta\hat{x}$ and $\Delta\hat{y}$.
  - Each mesh vertex maps to one depth value.

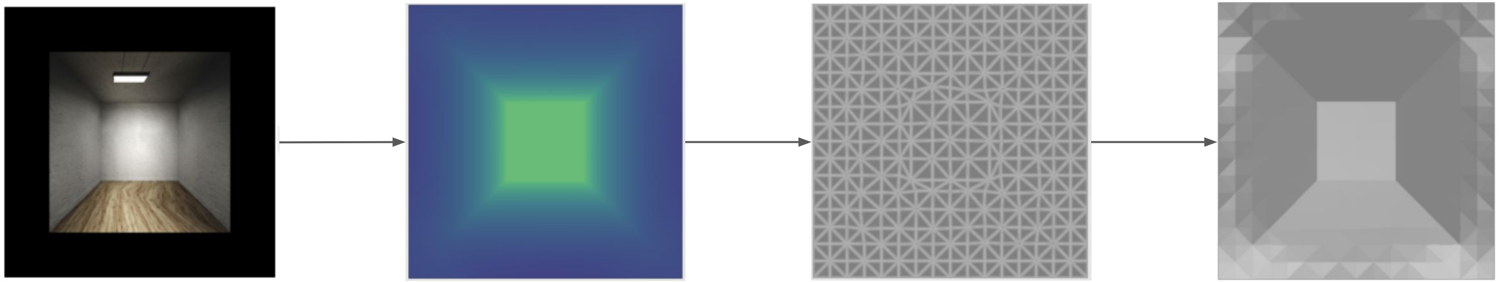
$$V_{w,h} = \begin{bmatrix} d_{w,h} \cdot (\hat{x}_{w,h} + \Delta\hat{x}_{w,h}) \cdot \tan(\theta_F/2) \\ d_{w,h} \cdot (\hat{y}_{w,h} + \Delta\hat{y}_{w,h}) \cdot \tan(\theta_F/2) \\ d_{w,h} \end{bmatrix}$$

d: depth
$\hat{x}, \hat{y}$: equally spaced anchor positions on the grid
$\Delta\hat{x}, \Delta\hat{y}$: grid offset
θ_F: camera angle of view

Method - Geometry Mesh Reconstruction

- **Reconstruction**: Worldsheet to reconstruct mesh based on depth
  - Mesh grid depends only on depth and fixed offset $\Delta\hat{x}$ and $\Delta\hat{y}$.
  - Each mesh vertex maps to one depth value.

$$V_{w,h} = \begin{bmatrix} d_{w,h} \cdot (\hat{x}_{w,h} + \Delta\hat{x}_{w,h}) \cdot \tan(\theta_F/2) \\ d_{w,h} \cdot (\hat{y}_{w,h} + \Delta\hat{y}_{w,h}) \cdot \tan(\theta_F/2) \\ d_{w,h} \end{bmatrix}$$

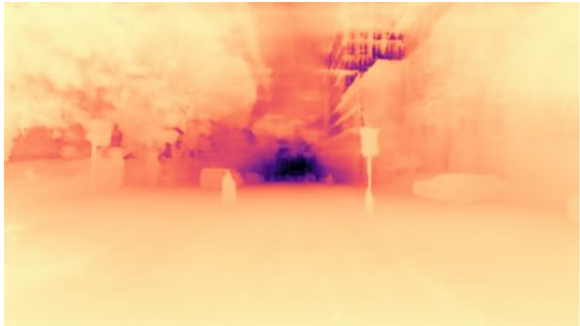


Input RGB — Depth map — Vertices — Mesh

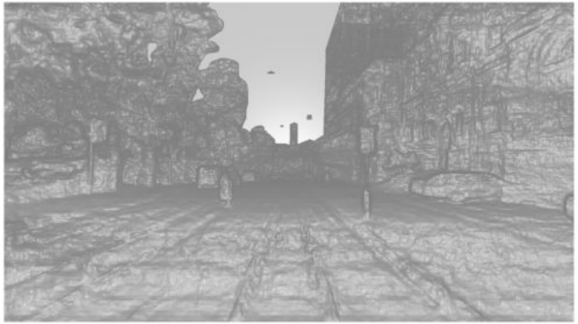
Method - Geometry Mesh Reconstruction

- Intermediate result

Input RGB



Depth



Reconstructed Mesh

Not so good… :(

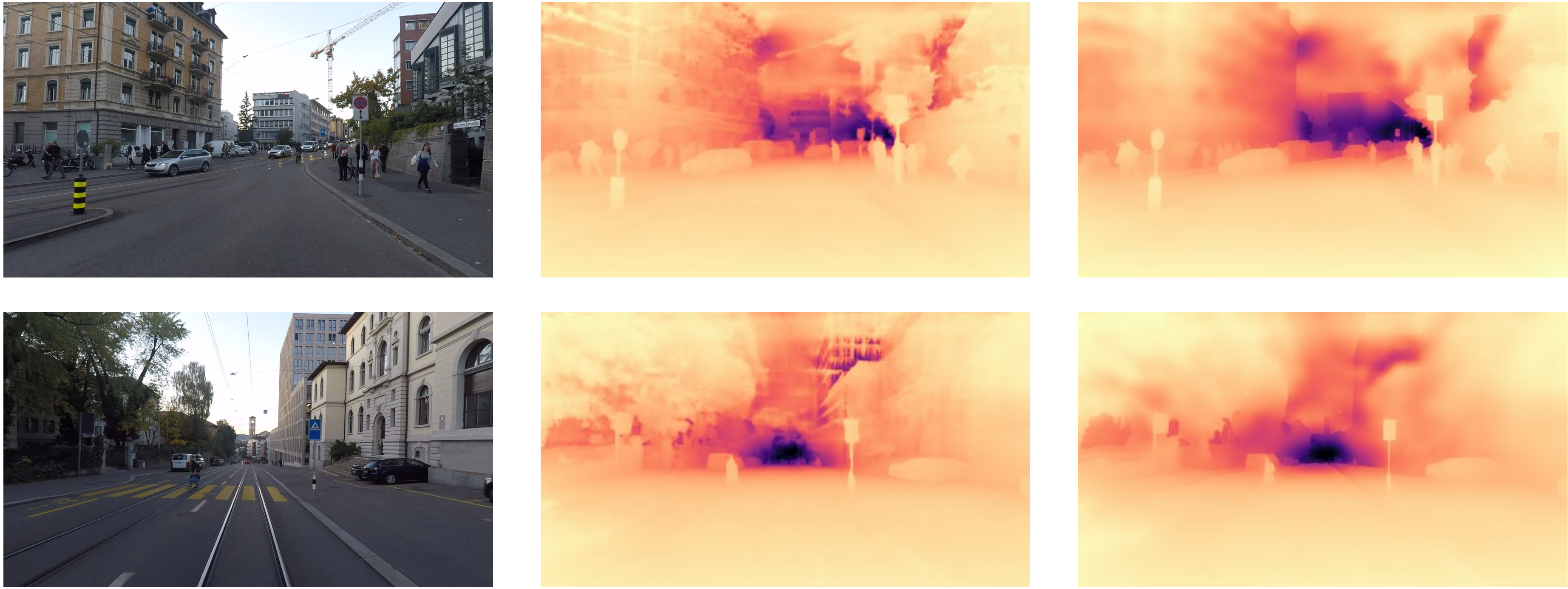
Method - Geometry Mesh Reconstruction

- **Depth refinement**: Dual-reference cross-bilateral filter
  - Dual reference: color and semantic annotations
  - Make depth sharper at semantic boundaries, preserve depth can be inferred by color

$$d(\mathbf{p}) = \frac{\sum_{q \in \mathcal{N}(\mathbf{p})} G_{\sigma_s}(\|\mathbf{q}-\mathbf{p}\|)\left[\delta(h(\mathbf{q})-h(\mathbf{p})) + \mu G_{\sigma_c}(\|\mathbf{J}(\mathbf{q})-\mathbf{J}(\mathbf{p})\|)\right]\hat{d}(\mathbf{q})}{\sum_{q \in \mathcal{N}(\mathbf{p})} G_{\sigma_s}(\|\mathbf{q}-\mathbf{p}\|)\left[\delta(h(\mathbf{q})-h(\mathbf{p})) + \mu G_{\sigma_c}(\|\mathbf{J}(\mathbf{q})-\mathbf{J}(\mathbf{p})\|)\right]}$$

J: CIELAB counterpart of the input RGB image
p, q: pixel location
$\mathcal{N}$: neighbouring pixels
h: semantic classes
δ: Kronecker delta
$G_{\sigma_s}, G_{\sigma_c}$: spatial Gaussian kernel colour Gaussian kernel

Method - Geometry Mesh Reconstruction

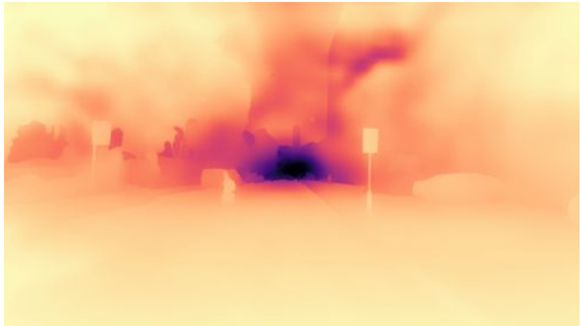


Input RGB Depth before Depth after

Method - Geometry Mesh Reconstruction

- Intermediate result

Input RGB



Depth

Reconstructed Mesh

Better now, but still not so good… :(

Method - Geometry Mesh Reconstruction

- **Depth refinement**: Dual-reference variance filter
  - Depth discontinuity: Create spurious faces
  - Uncertain region: Depth discontinuity at semantic boundaries

$$\mathcal{U}(r(\mathbf{p},l)) = (\mathcal{V}(d(r(\mathbf{p},l))) > \mu) \text{ and } (\mathcal{V}(h(r(\mathbf{p},l))) > 0)$$

Method - Geometry Mesh Reconstruction

- **Depth refinement**: Dual-reference variance filter
  - Depth discontinuity: create spurious faces
  - Detect depth discontinuity at semantic boundaries

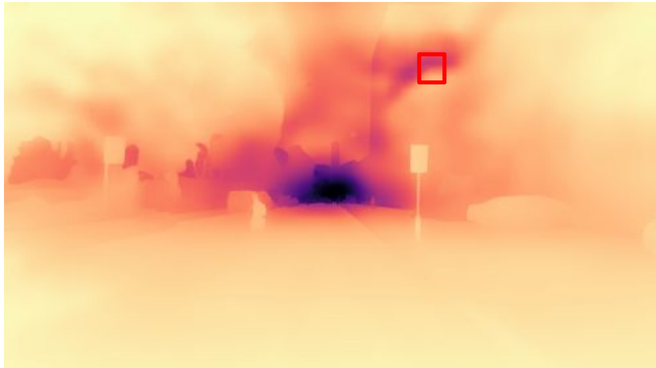
$$\mathcal{U}(r(\mathbf{p}, l)) = (\mathcal{V}(d(r(\mathbf{p}, l))) > \mu) \text{ and } (\mathcal{V}(h(r(\mathbf{p}, l))) > 0)$$

$r(\mathbf{p}, l)$: $l * l$ square region with $\mathbf{p}$ as the upper-left pixel location
$d(\cdot)$: depth of the region
$h(\cdot)$: semantic label of the region
$\mathcal{V}$: variance operation

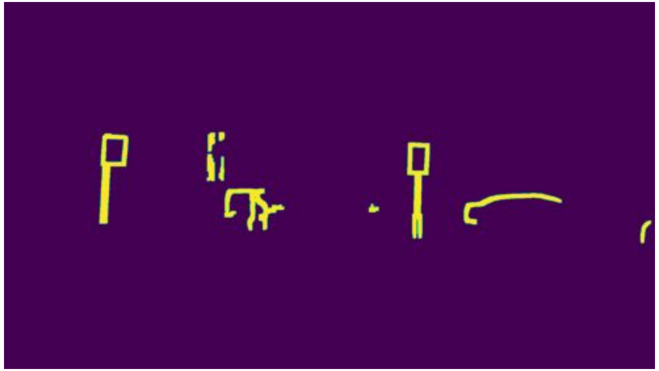
Method - Geometry Mesh Reconstruction

Input RGB

Semantic annotation



Depth map

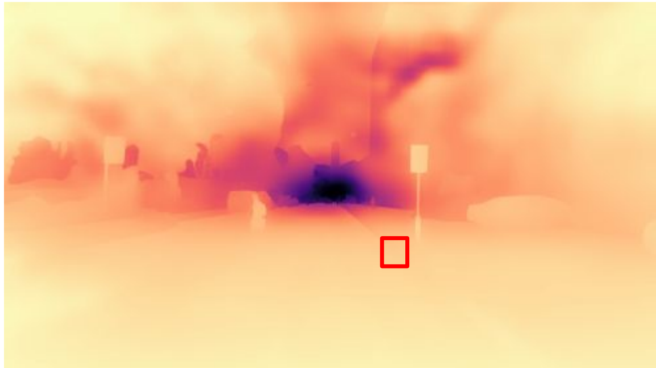


Uncertain region

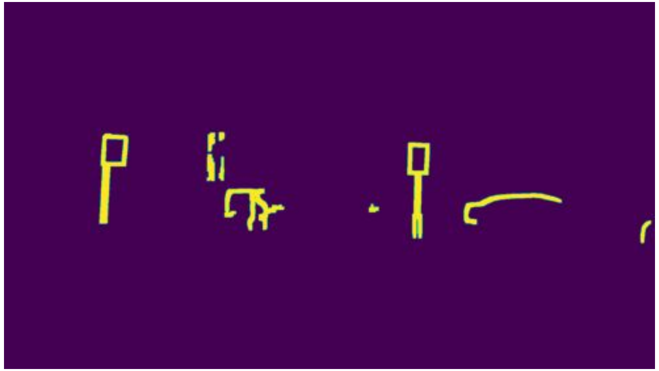
Method - Geometry Mesh Reconstruction

Input RGB

Semantic annotation



Depth map

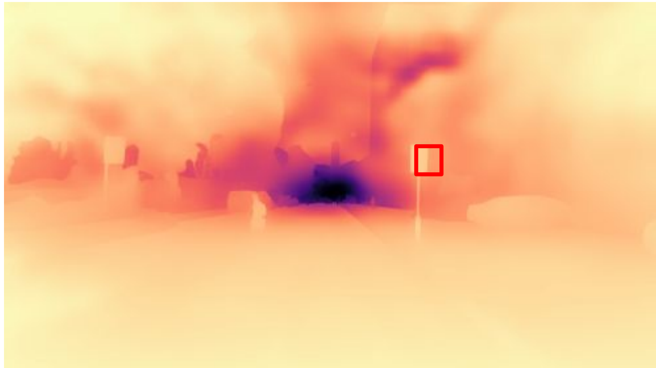


Uncertain region

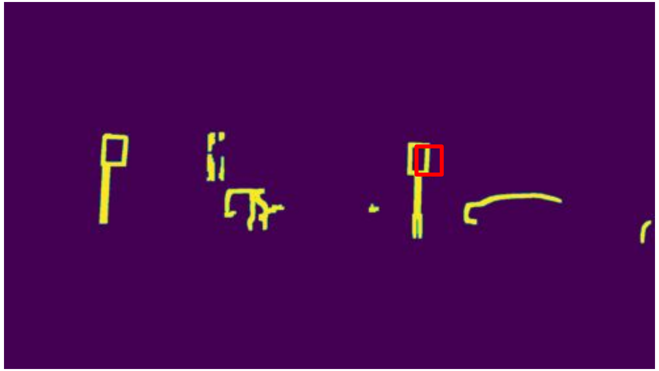
Method - Geometry Mesh Reconstruction

Input RGB

Semantic annotation



Depth map



Uncertain region

Method - Geometry Mesh Reconstruction

- **Depth refinement**: Normal-guided depth refinement via optimization
  - Normal loss: Normals inferred from depth should match estimated normals
  - Continuity loss: Depth change should be continuous
  - Depth loss: Optimized depth should respect initial estimation

$$\hat{\mathbf{N}} = \nabla\mathbf{X} \times \nabla\mathbf{Y} = (1, 0, \frac{\partial z}{\partial x}) \times (0, 1, \frac{\partial z}{\partial y}) = (-\frac{\partial z}{\partial x}, -\frac{\partial z}{\partial y}, 1)$$

$$L_{\text{normal}} = \|\hat{\mathbf{N}} - \mathbf{N}_{\text{est}}\|_2^2$$

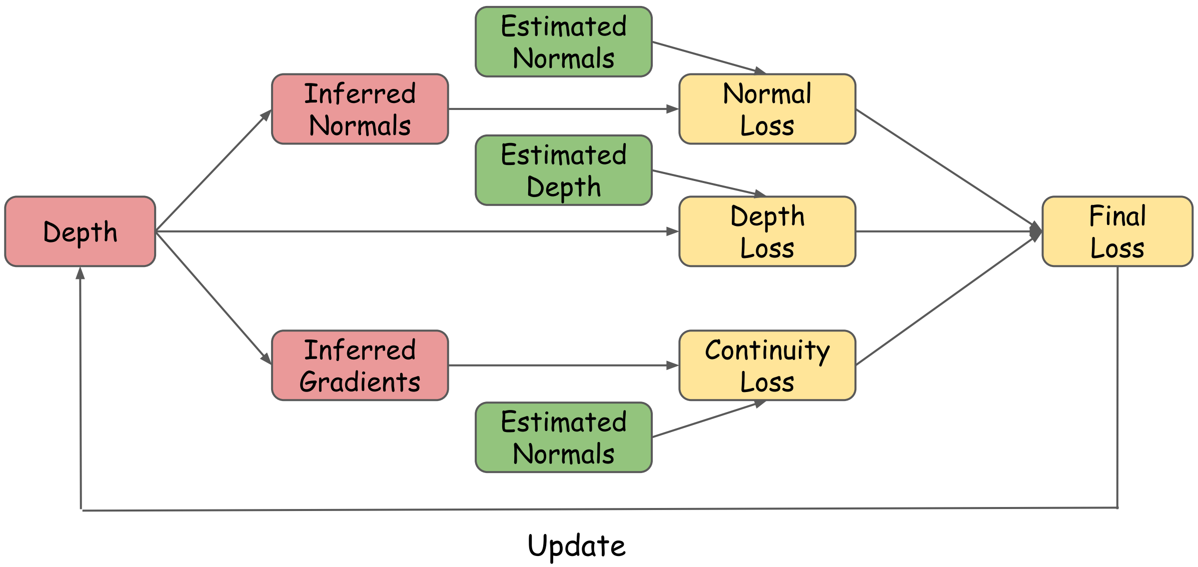
$$L_{continuity} = \frac{1}{n}\sum_{i=1}^{n}((\nabla\mathbf{X}_{\text{i}} \cdot \mathbf{N}_{\text{i}})^2 + (\nabla\mathbf{Y}_{\text{i}} \cdot \mathbf{N}_{\text{i}})^2) \cdot (1 - \mathcal{U}_{\text{i}})$$

$$L_{depth} = \|\hat{d} - d_{est}\|_2^2$$

Method - Geometry Mesh Reconstruction

- **Depth refinement**: Normal-guided depth refinement
  - Final loss: Linear combination of all three losses

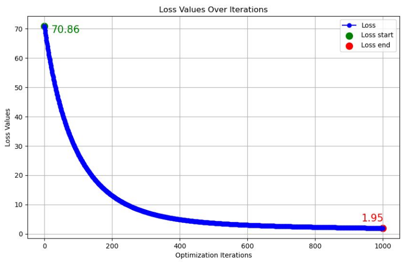
$$L_{final} = \lambda_1 L_{normal} + \lambda_2 L_{continuity} + \lambda_3 L_{depth}$$

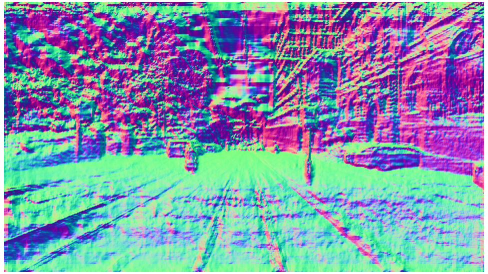


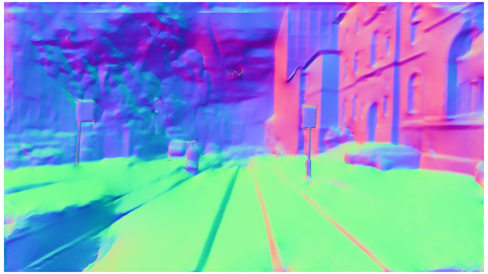
Method - Geometry Mesh Reconstruction

- **Depth refinement**: Normal-guided depth refinement
  - Final loss: Linear combination of all three losses

$$L_{final} = \lambda_1 L_{normal} + \lambda_2 L_{continuity} + \lambda_3 L_{depth}$$



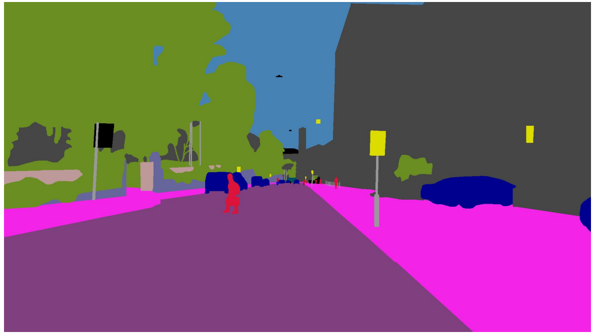




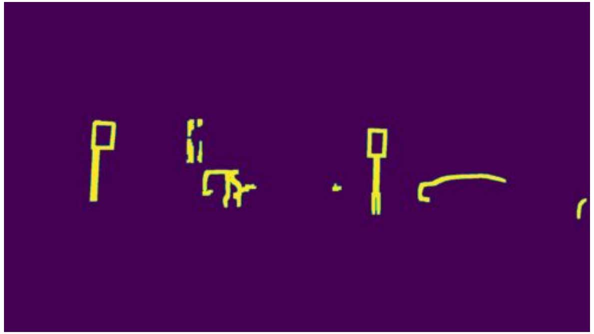
Method - Geometry Mesh Reconstruction

- **Mesh post-processing**: Remove uncertain spurious
  - Face deletion: Delete spurious faces between foreground and background objects
  - Spurious faces: At least one vertex falls into uncertain region

Input RGB



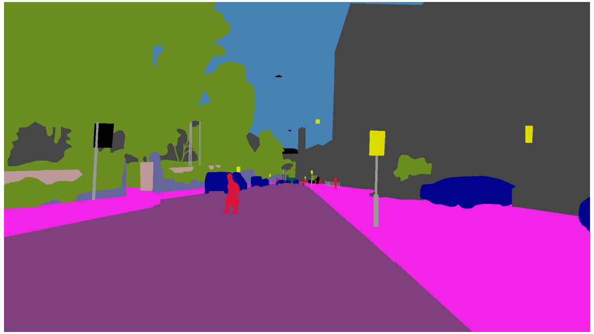
Semantic annotation



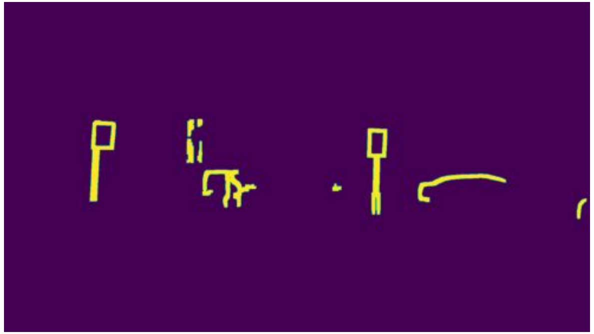
Uncertain region

Method - Geometry Mesh Reconstruction

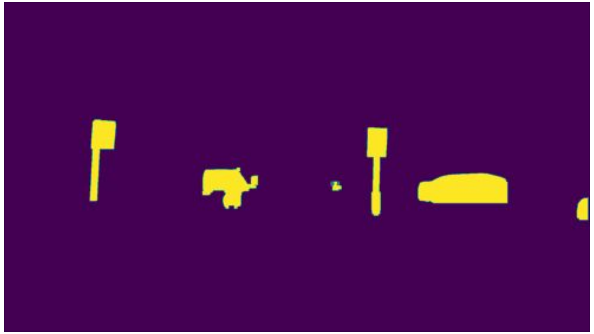
- **Mesh post-processing**: Background face completion
  - Determine regions needed to be completed: union of uncertain region and it neighbouring foreground object semantics



Semantic annotation



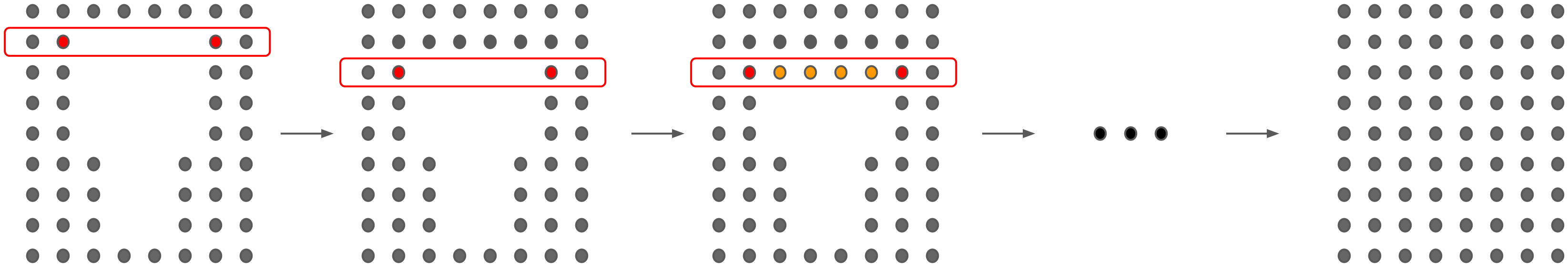
Uncertain region



Background completion region

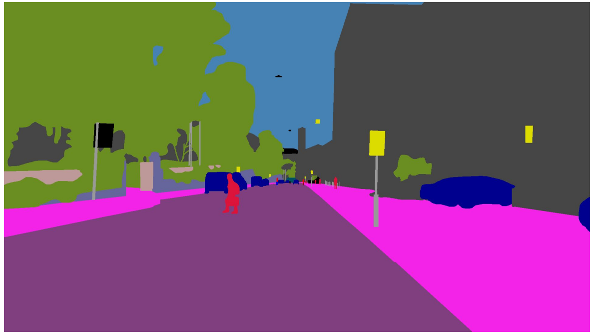
Method - Geometry Mesh Reconstruction

- **Mesh post-processing**: Background face completion
  - Determine regions needed to be completed: union of uncertain region and foreground object semantics
  - Background objects: For each line of missing vertices, add new vertices linearly distributed between their left and right vertices

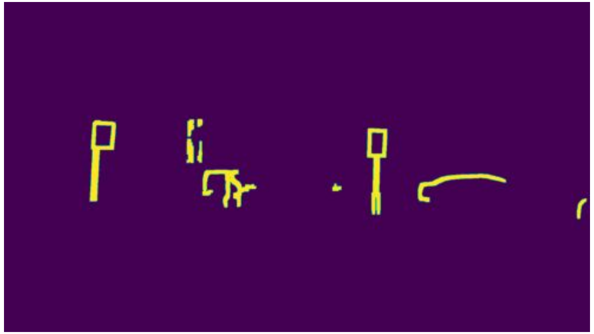


Method - Geometry Mesh Reconstruction

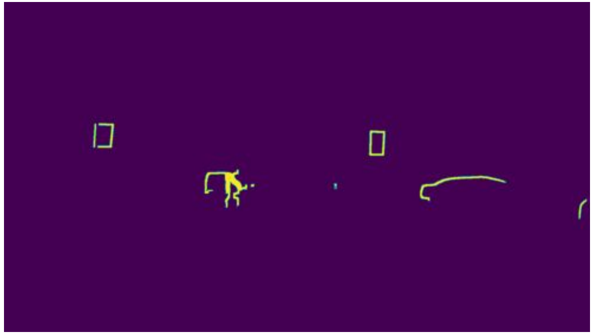
- **Mesh post-processing**: Foreground face completion
    - Determine regions needed to be completed: intersection of uncertain region and neighbouring foreground object semantics



Semantic annotation



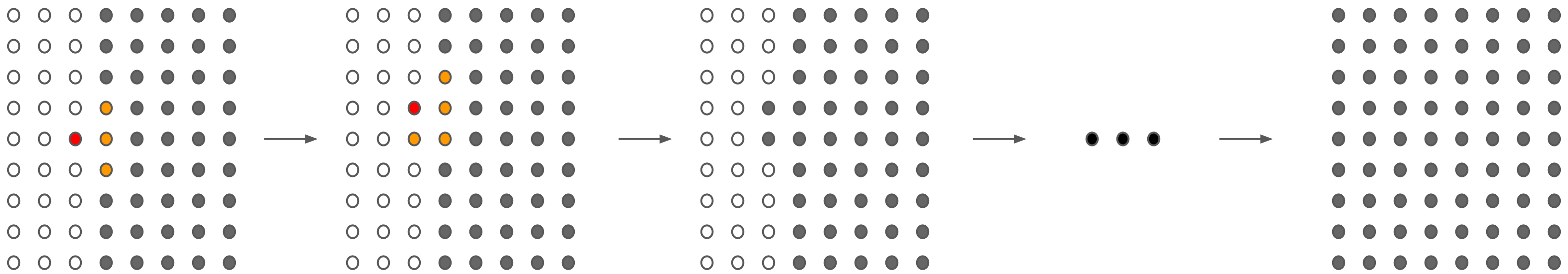
Uncertain region



Foreground completion region

Method - Geometry Mesh Reconstruction

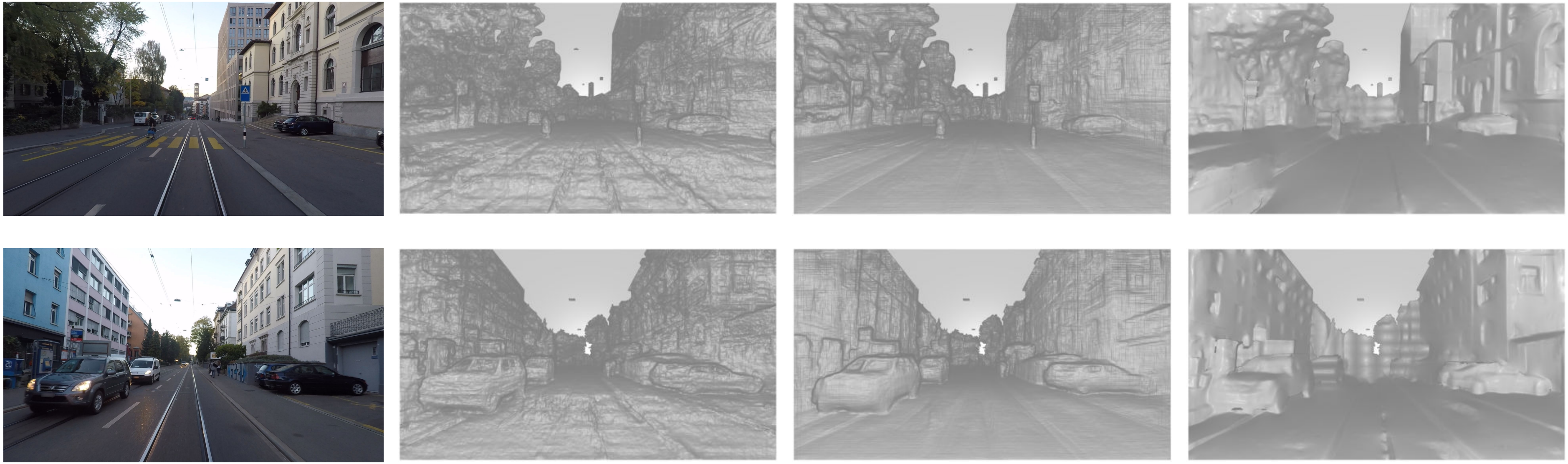
- **Mesh post-processing**: Foreground face completion
  - Determine regions needed to be completed: intersection of uncertain region and foreground object semantics
  - Foreground objects: Set depth of each missing vertex to the average of its neighbouring vertices belongs to the foreground object



Method - Realistic Nighttime Scene Relighting [preview]

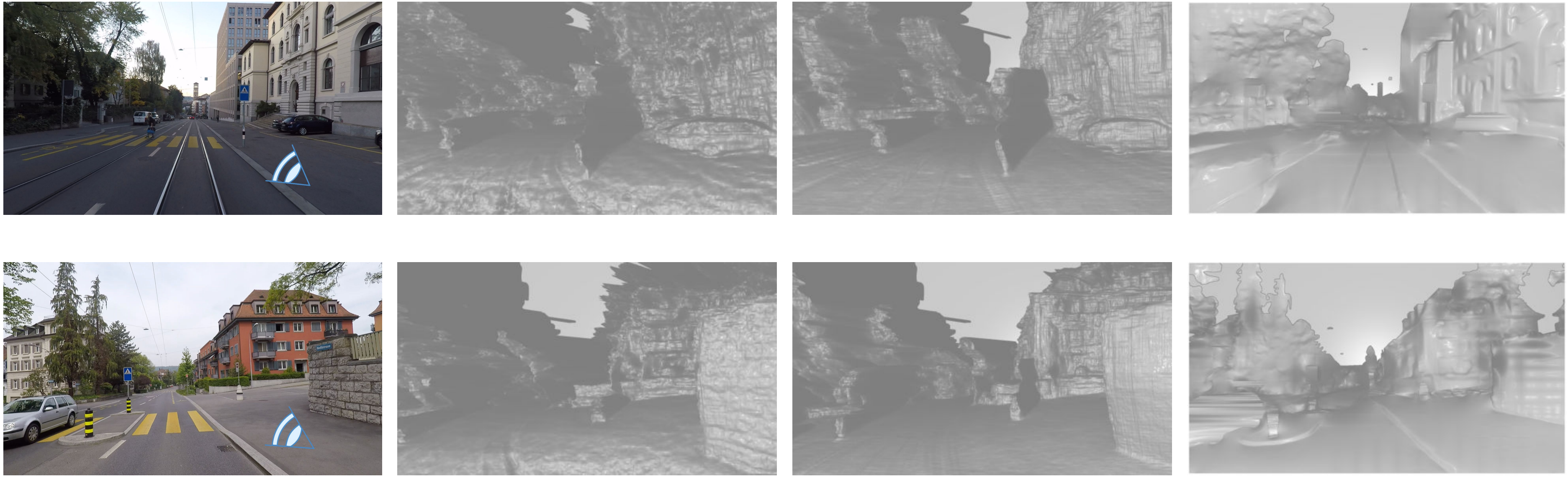
- F_{ir} : Inverse Rendering for Complex Indoor Scenes to estimate albedo and roughness
- Probabilistic light source activation:
  - Each light source as independent random variable with a Bernoulli Distribution
  - Light sources from same group shares the same activation parameter
- Ray tracing with scene mesh, material characteristics and light sources
- Post-processing to add noise on the clear nighttime image

Experiment - Mesh Comparison



Input RGB | iDisc + Worldsheet | SIMBAR | NPSim

Experiment - Mesh Post-processing



Input RGB

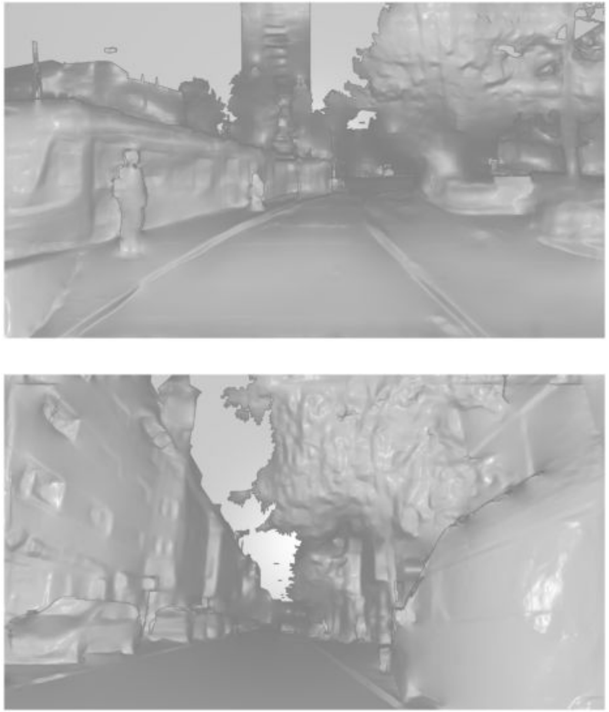
iDisc + Worldsheet Rotation View

SIMBAR Rotation View

NPSim Remove Foreground Objects

Experiment - ACDC Dataset

Input RGB



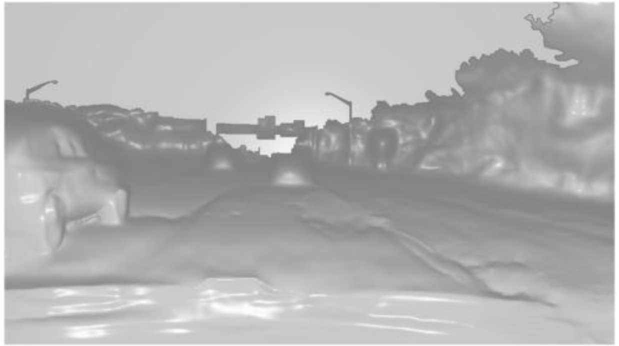
Reconstructed Mesh

Experiment - Generalization Cityscapes Dataset

Input RGB

Reconstructed Mesh

Experiment - Generalization BDD100K Dataset



Input RGB

Reconstructed Mesh

Conclusion

- NPSim: Day-to-night simulation pipeline based on monocular inverse rendering and ray tracing
- Mesh reconstruction: depth refinement kernel and mesh post-processing kernel
- Reconstruction result: on ACDC, Cityscapes, BDD100K
- Limitations:
  - Inactive light source mask requires manual annotation
  - Depth refinement may become inaccurate due to wrong normals estimation