Denoising for Path Tracing

CS 482 Interactive Computer Graphics

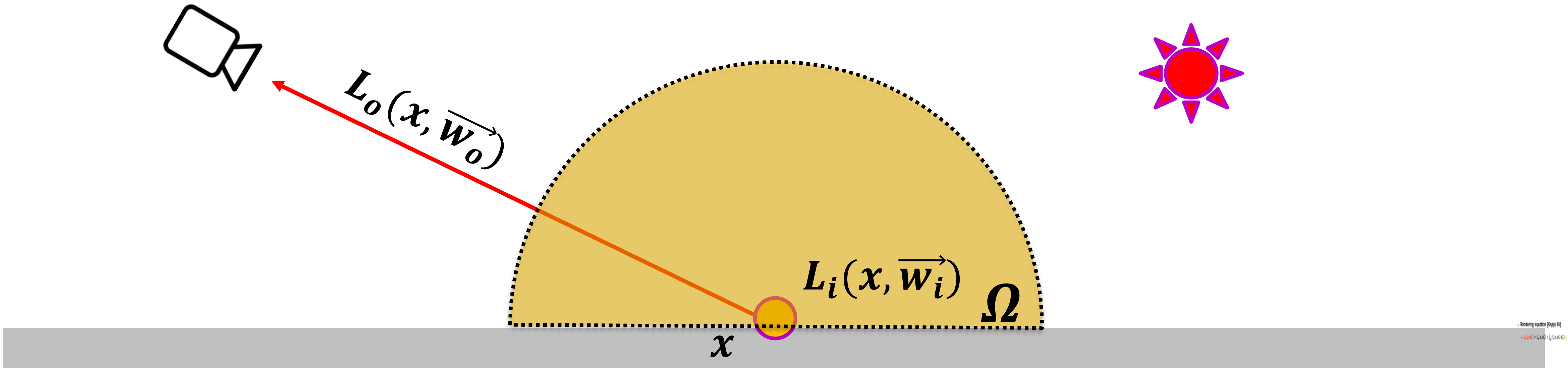
Kyubeom Han (TA)

Contents

- Review – Rendering Equation to Path Tracing
- Monte Carlo Noise and Denoising
- Classical methods for Monte Carlo Denoising
- Deep learning based Monte Carlo Denoising

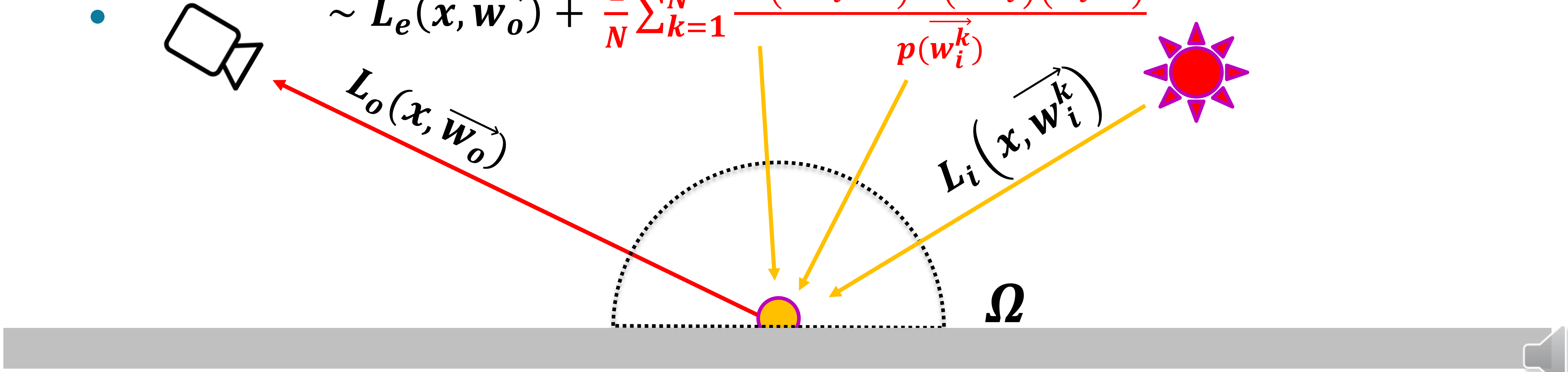
Review - Rendering Equation

- **Rendering equation [Kajiya 86]**
  - $L_o(x,\overrightarrow{w_o}) = L_e(x,\overrightarrow{w_o}) + \int_\Omega f_r(x,\overrightarrow{w_i},\overrightarrow{w_o}) L_i(x,\overrightarrow{w_i})(\overrightarrow{w_i}\cdot\vec{n}\,)d\overrightarrow{w_i}$



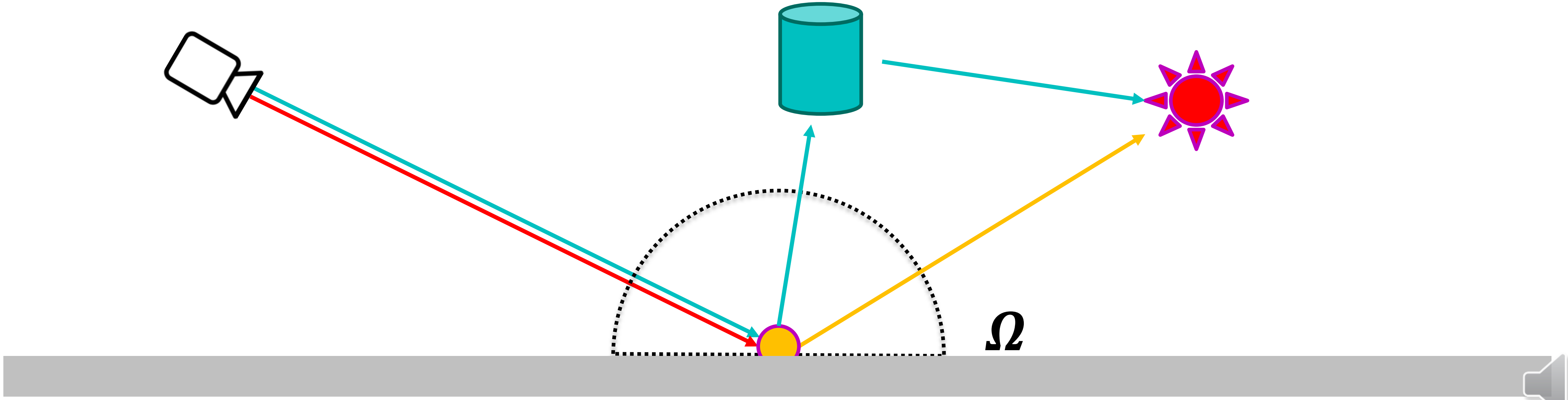
Review – Monte Carlo Integration

- **Monte Carlo Ray Tracing**
  - $L_o(x,\overrightarrow{w_o}) = L_e(x,\overrightarrow{w_o}) + \int_\Omega f_r(x,\overrightarrow{w_i},\overrightarrow{w_o}) L_i(x,\overrightarrow{w_i})(\overrightarrow{w_i}\cdot\vec{n})\,d\overrightarrow{w_i}$
  - $\sim L_e(x,\overrightarrow{w_o}) + \frac{1}{N}\sum_{k=1}^{N}\frac{f_r\left(x,\overrightarrow{w_i^k},\overrightarrow{w_o}\right)L_i\left(x,\overrightarrow{w_i^k}\right)\left(\overrightarrow{w_i^k}\cdot\vec{n}\right)}{p\left(\overrightarrow{w_i^k}\right)}$



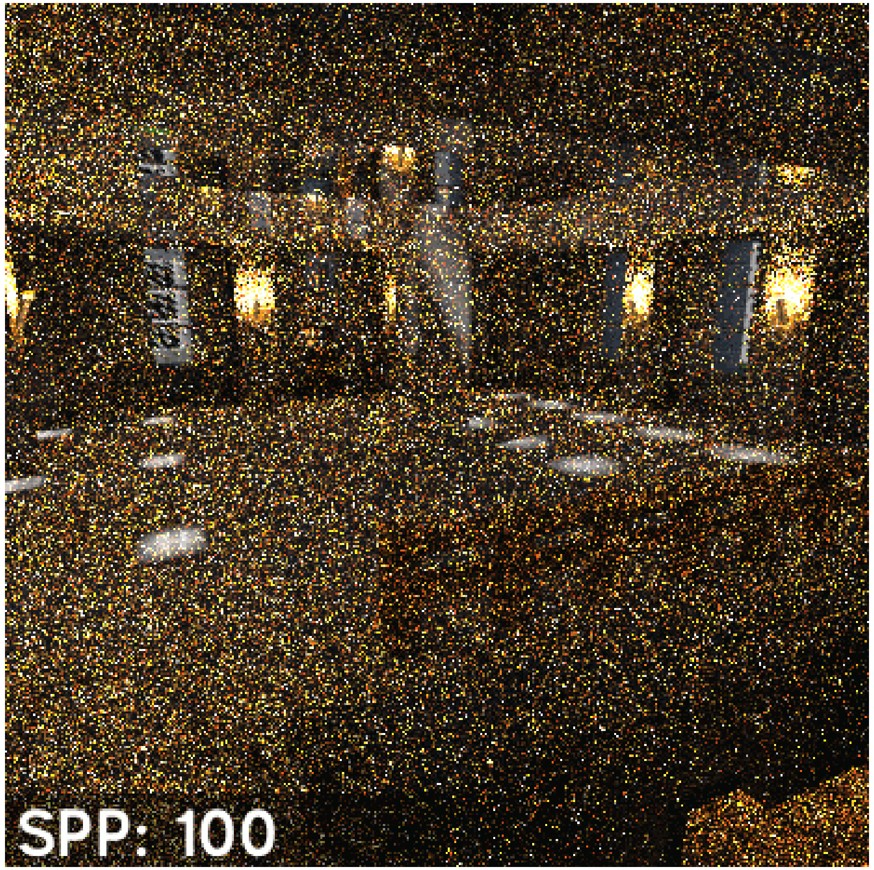
Review – Path Tracing

- Shoot ray from the camera
- Branch one secondary ray
- Recurse until it reaches a light source (or do Russian Roulette)



Monte Carlo Noise in Path Tracing

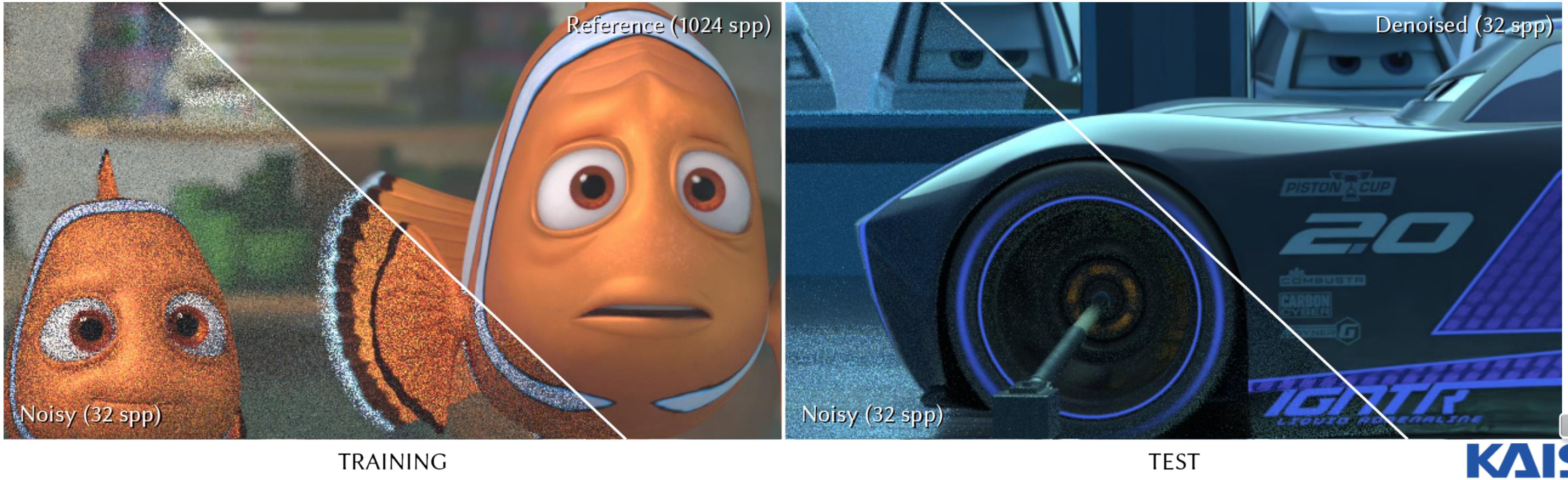
- Noise due to the discrepancy between real and sampling PDF
- Requires a lot of rays (10,000~100,000) per pixel to converge



https://chunky.llbit.se/path_tracing.html

Monte Carlo Denoising for Path Tracing

- Denoising to get a high-quality rendering with few samples
- Reduce rendering time cost by using few samples



Post-processing for MC Denoising

Path Tracer $\xrightarrow[t_{render}]{\text{10,000 spp}}$

$$t_{render} > t'_{render} + t_{denoise}$$

Path Tracer $\xrightarrow[t'_{render}]{\text{8 spp}}$

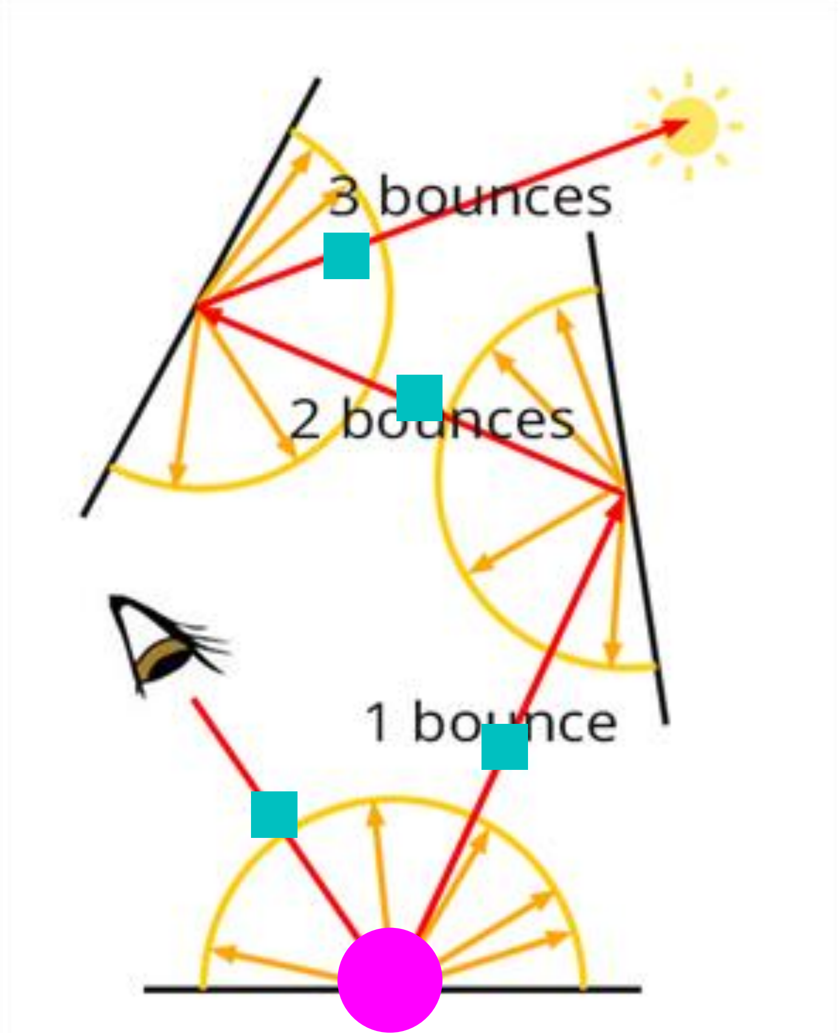
$\rightarrow$ Denoising Method ($t_{denoise}$) $\rightarrow$

Denoising for Path Tracing

- General images contains a random Gaussian Noise
  - Temperature instability of the camera sensors

- Denoising for Path Tracing
  - Discrepancy between two PDFs
  - Scene geometry

Auxiliary Features for Denoising



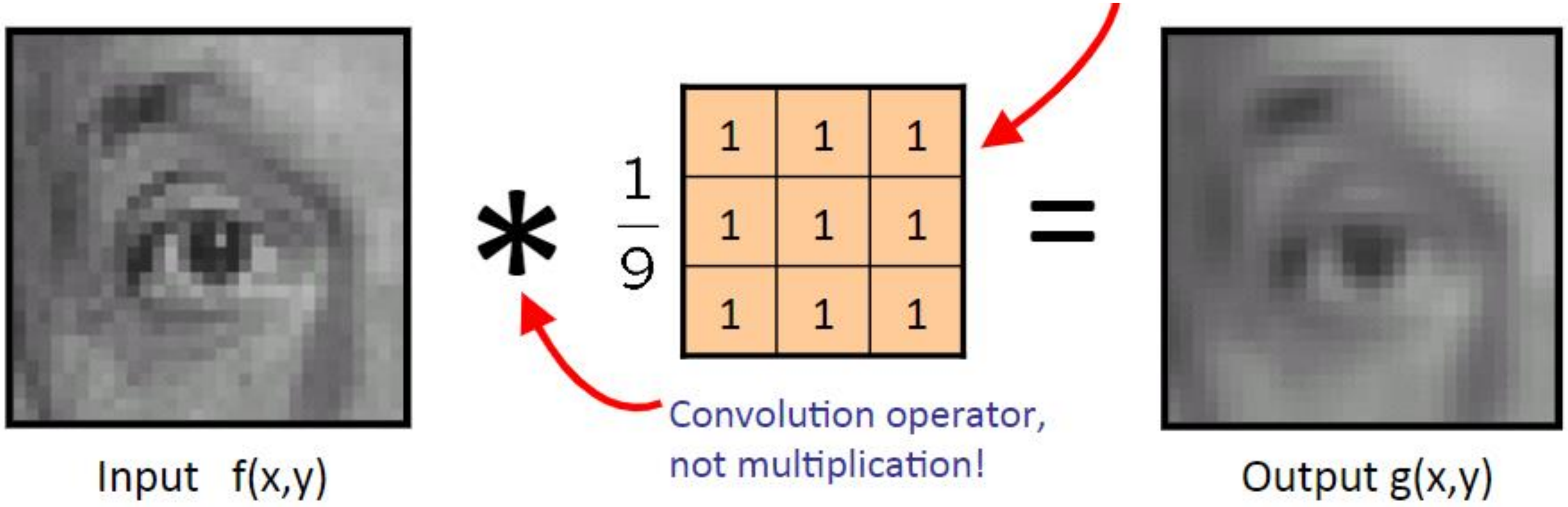
Geometry Features
Albedo (Texture), Normal, Depth

Sample Features
Radiance of sample

Path Features
Radiance of each path
Direction of incoming light
Material properties of vertex
Etc…

Classical Denoising Methods

- **General Image Denoising Methods**
  - **Deriving appropriate weights of nearby pixels**



https://medium.com/@boelsmaxence/introduction-to-image-processing-filters-179607f9824a

Classical Denoising Methods

- General Image Denoising Methods
  - Deriving appropriate weights of nearby pixels
    - Gaussian filtering
    - Bilateral filtering
    - Non-local means filtering
    - Wavelet-based filtering
    - Etc...

Classical Denoising Methods

- **E.g. Cross-bilateral filter**

$$\hat{c}_i = \frac{\sum_{j \in N(i)} w_{ij} \bar{c}_j}{\sum_{j \in N(i)} w_{ij}}$$

Noisy pixel

Neighbor pixels

Contribution of pixels

$$w_{ij} = \exp\left[-\frac{1}{2\sigma_p^2} \sum_{1 \le k \le 2} (\bar{p}_{i,k} - \bar{p}_{j,k})^2\right] \times \exp\left[-\frac{1}{2\sigma_c^2} \sum_{1 \le k \le 3} \alpha_k (\bar{c}_{i,k} - \bar{c}_{j,k})^2\right] \times \exp\left[-\frac{1}{2\sigma_f^2} \sum_{1 \le k \le m} \alpha_k (\bar{f}_{i,k} - \bar{f}_{j,k})^2\right]$$

On Filtering the Noise from the Random Parameters in Monte Carlo Rendering, Sen, Darabi et al. 2012

Classic Denoising Methods

- **Human design leads to biases…**

Winer filtering

Bilateral filtering

PCA method

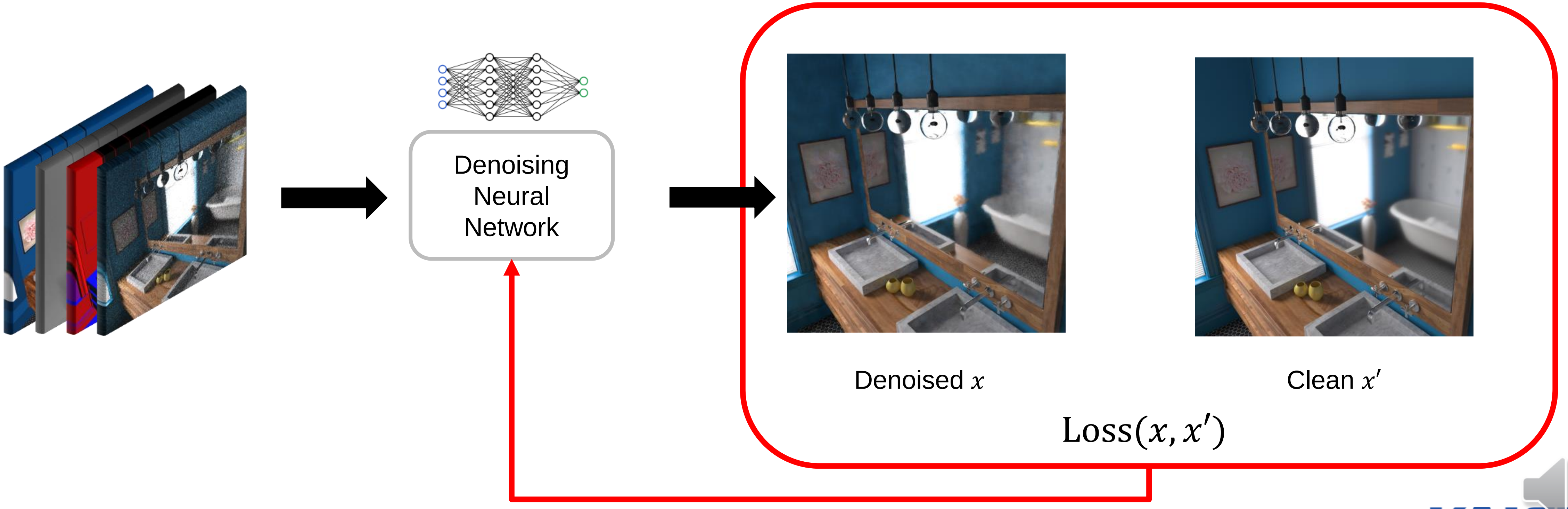
Wavelet filtering

Collaborative filtering

Brief Review of Image Denoising Techniques, Fan et al. 2019

Deep Learning for Denoising

- **Training a neural network to detect and remove the noise**



Deep Learning for Denoising

- Various choices of neural network
  - Multi-layer perceptron
  - Convolutional Network
  - Self attention
  - Etc…
- I assume you know basics of deep learning

Three-Scale DL-based Denoisers

- Geometric features
  - Kernel Predicting Convolutional Networks (Bako et al., 2017)
- Sample features
  - Sample-based Monte Carlo Denoising (Gharbi et al., 2019)
- Path features
  - Weakly-supervised Contrastive Learning in Path Manifold (Cho et al., 2021)

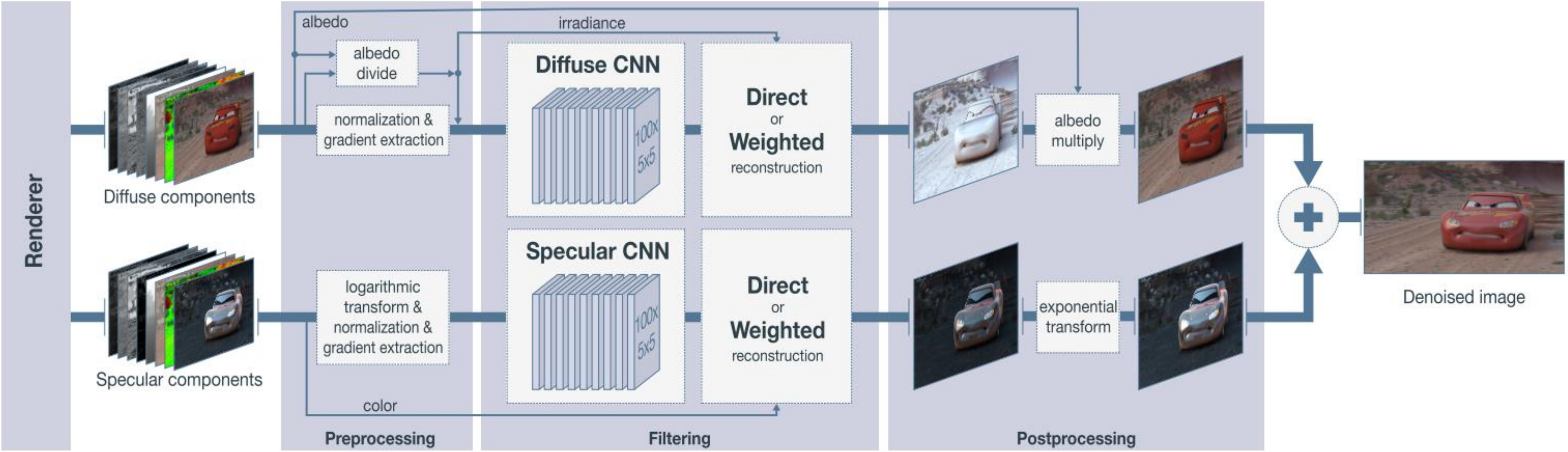
KAIST

Kernel Predicting Convolutional Networks

- A **two-stream convolutional network** that predicts the weights of the nearby pixels.
- Makes prediction from the **noisy input and the geometric features**
  - Geometric features: normal, albedo, depth and their gradients & variance

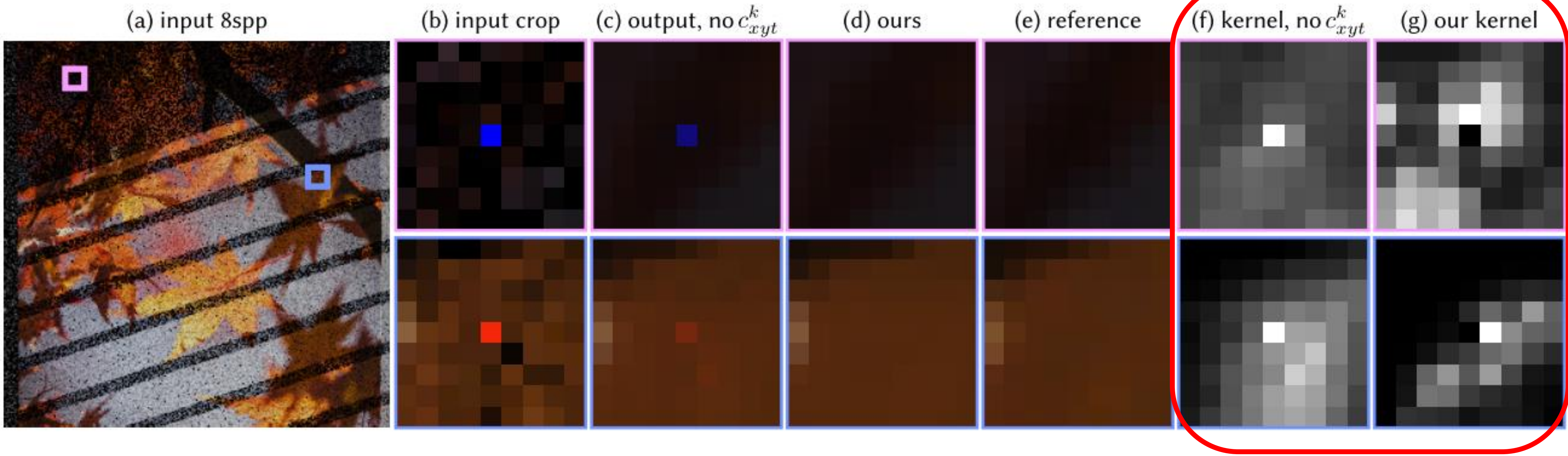
Kernel-Predicting Convolutional Networks for Denoising Monte Carlo Renderings, Bako et al, 2017

Kernel Predicting Convolutional Networks



Kernel-Predicting Convolutional Networks for Denoising Monte Carlo Renderings, Bako et al, 2017

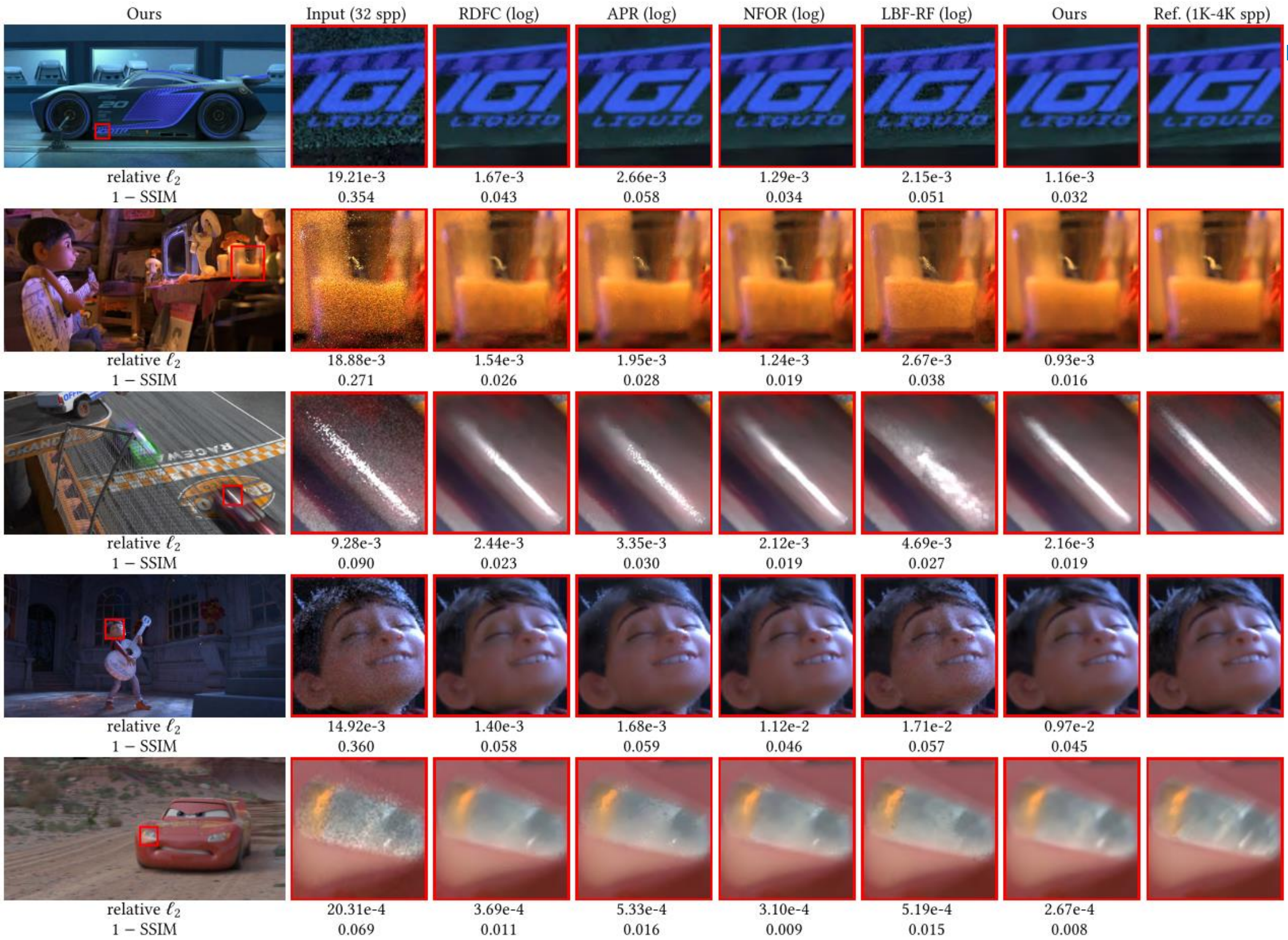
Kernel Predicting Convolutional Networks



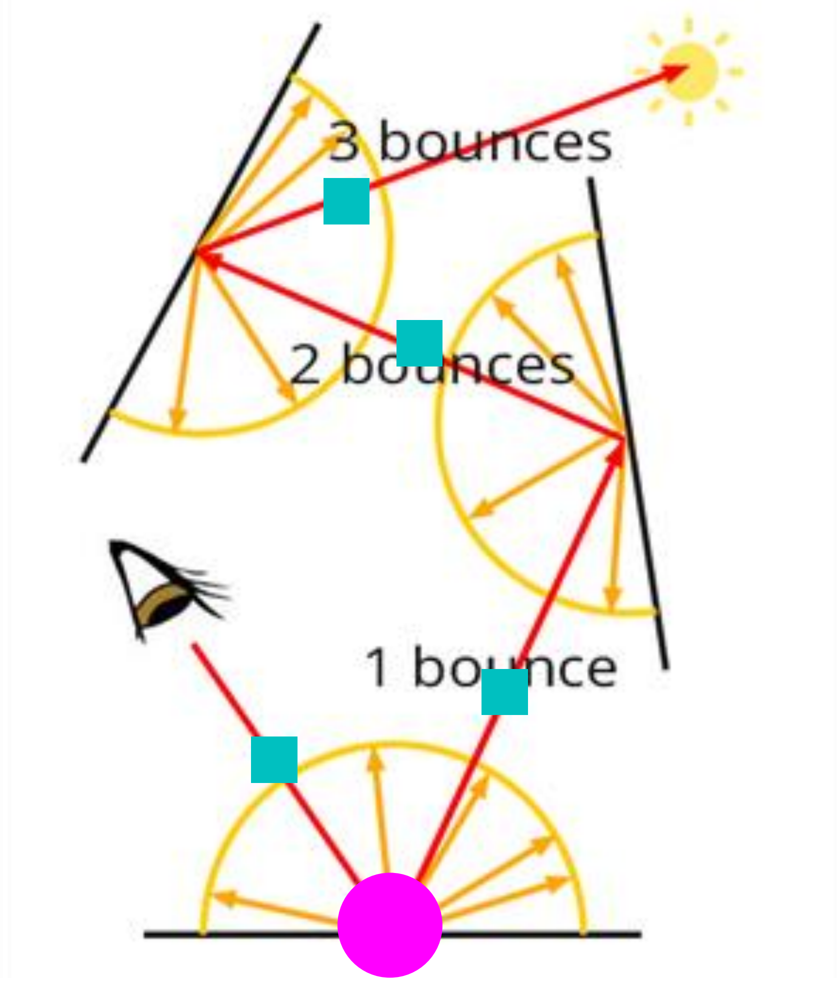
Interactive Monte Carlo Denoising using Affinity of Neural Features, Isik et al, 2021

Kernel Predicting Convolutional Networks

Kernel Predicting Convolutional Networks



Richer Auxiliary Features for Denoising



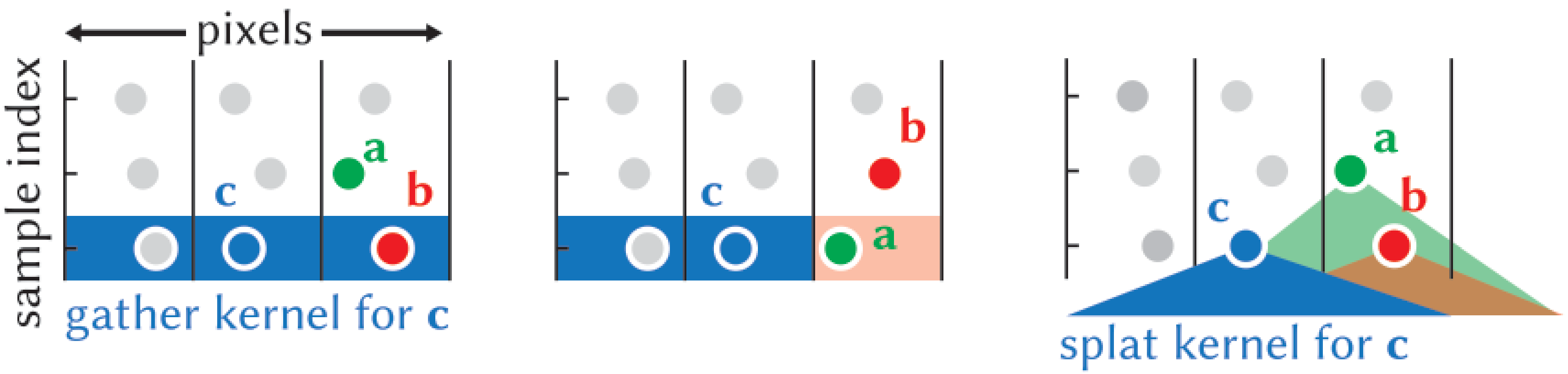
Geometry Features
Albedo (Texture), Normal, Depth

Sample Features
Radiance of sample

Path Features
Radiance of each path
Direction of incoming light
Material properties of vertex
Etc…

Sample-based Monte Carlo Denoising

- **Predict contribution of each sample**

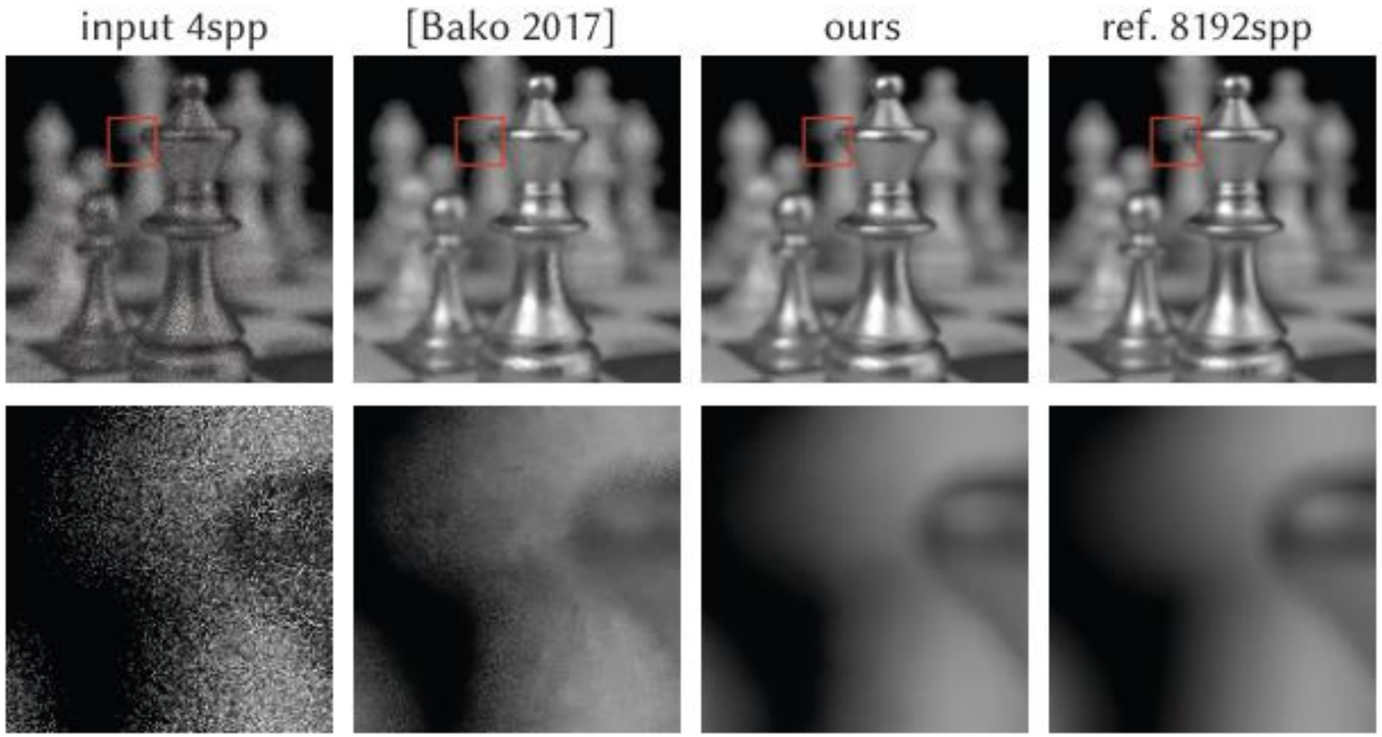


(a) gather (b) invalid gather after permutation of **a**, **b** (c) splat

Sample-based Monte Carlo Denoising using a Kernel Splatting Network, Gharbi et al, 2019

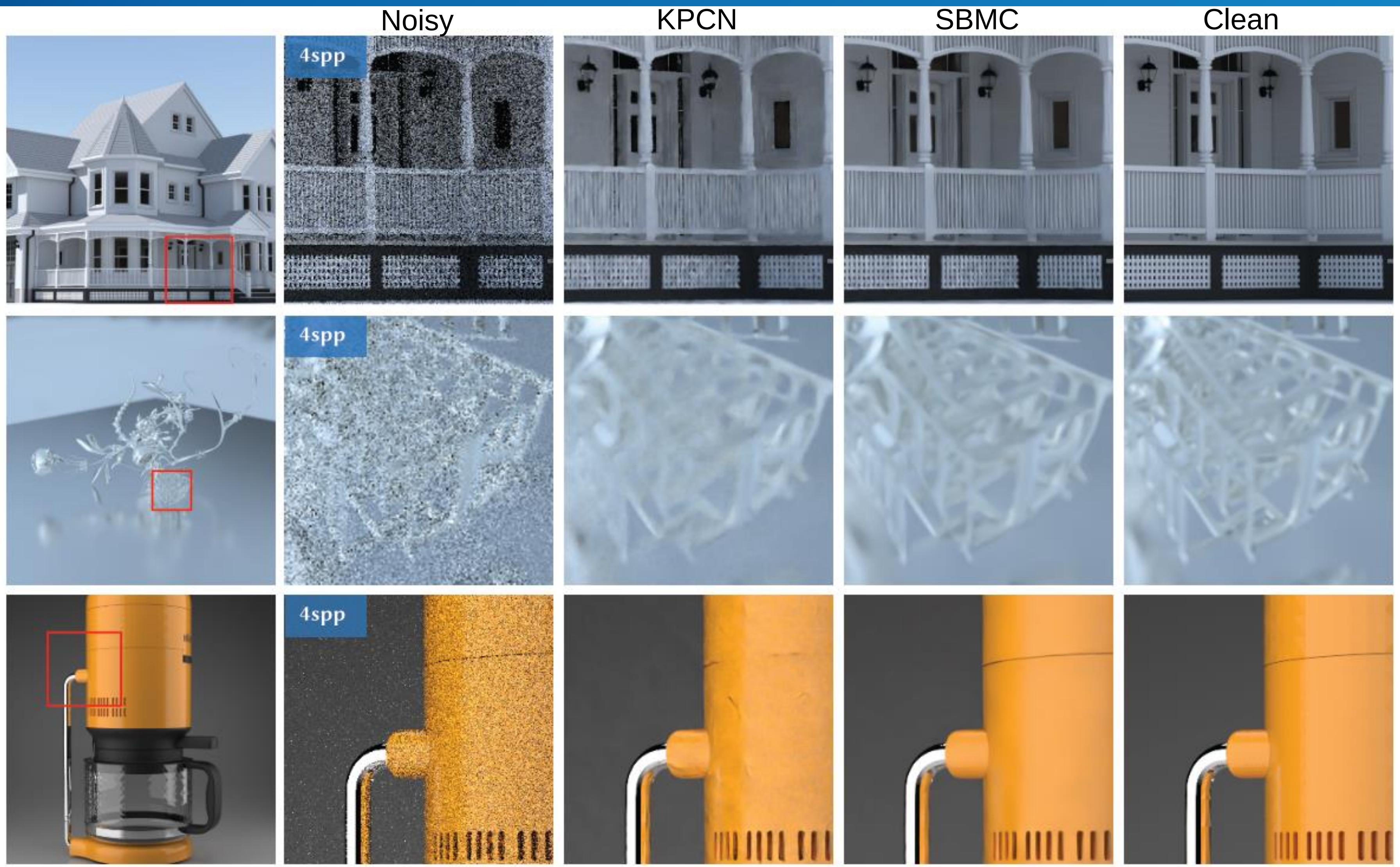
Sample-based Monte Carlo Denoising

- **Predict contribution of each sample**

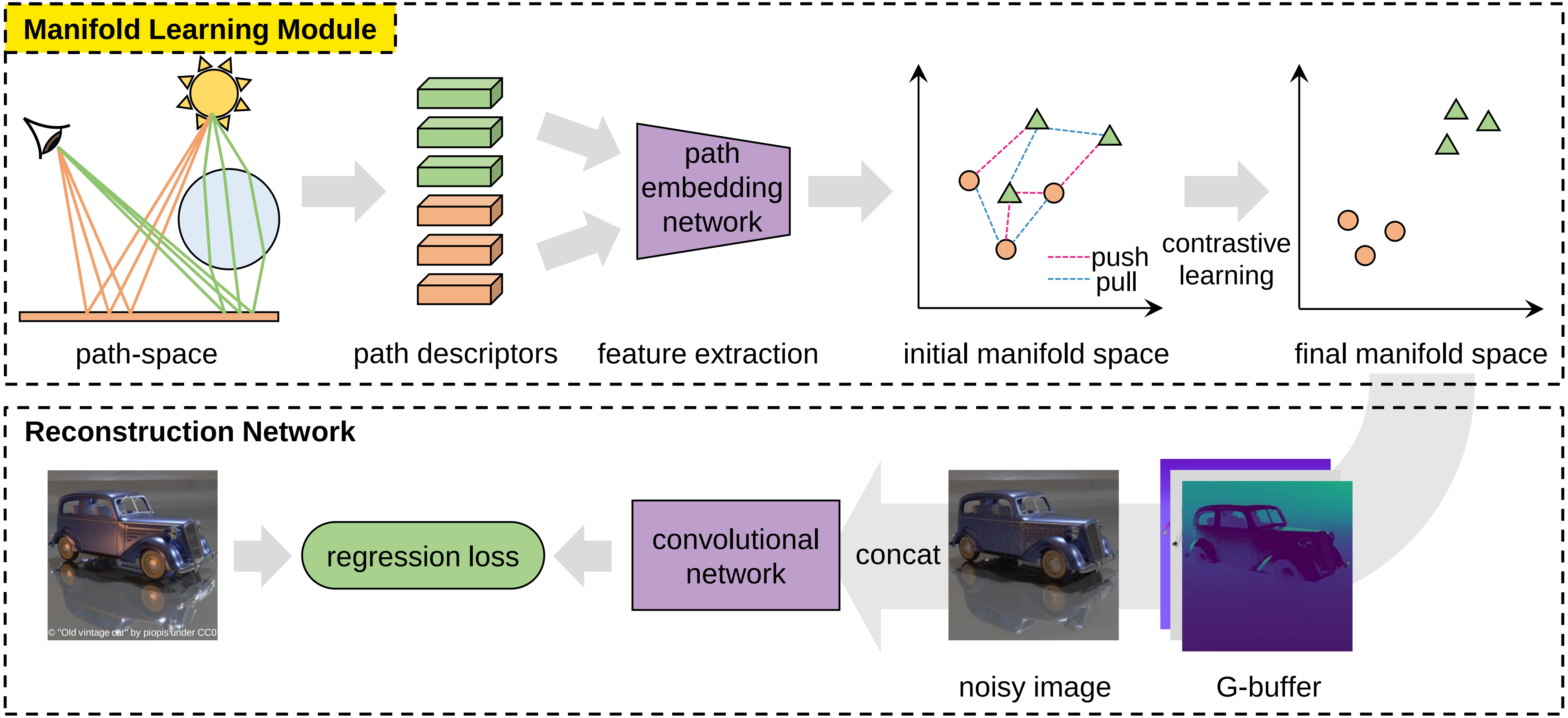


Sample-based Monte Carlo Denoising using a Kernel Splatting Network, Gharbi et al, 2019

Sample-based Monte Carlo Denoising

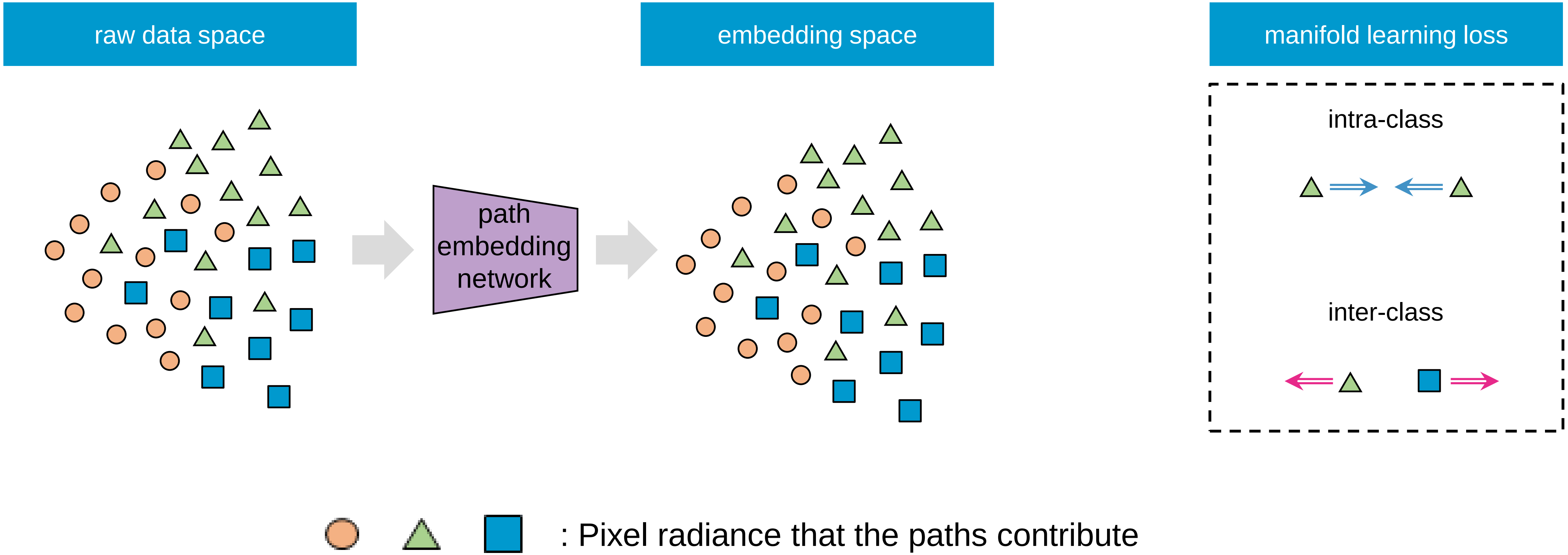


Path-based Monte Carlo Denoising

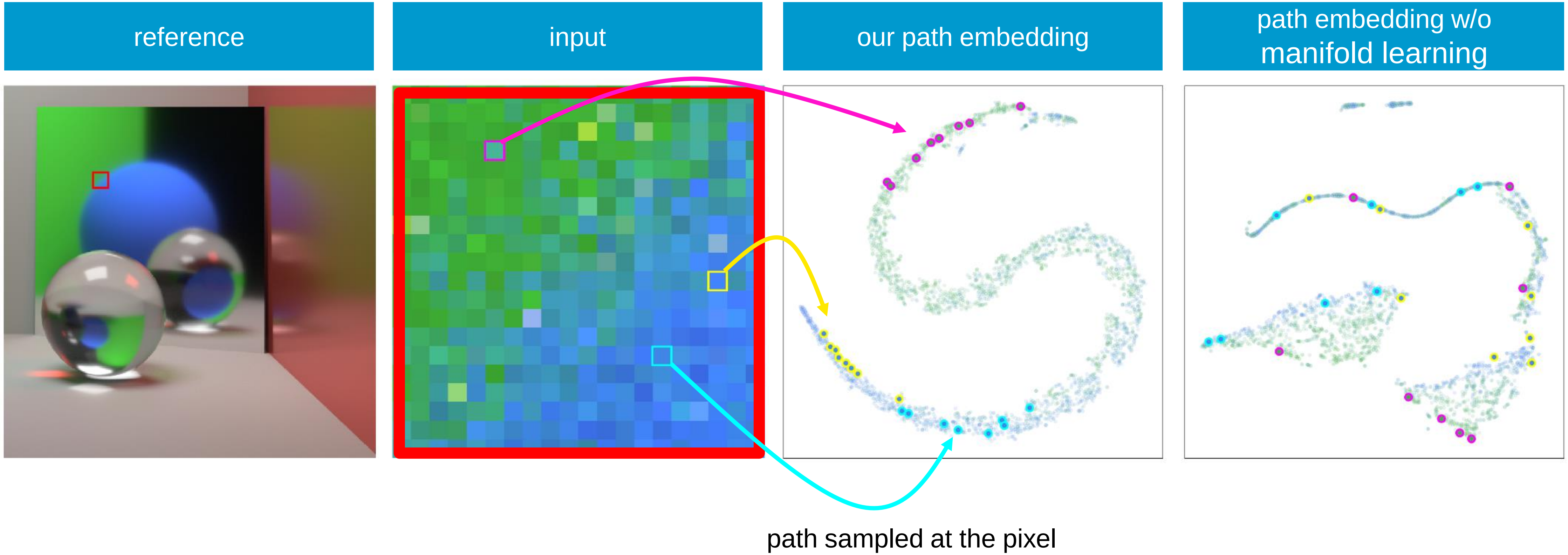


Weakly-supervised Contrastive Learning in Path Manifold for Monte Carlo Image Reconstruction, Cho et al, 2021

Path-based Monte Carlo Denoising

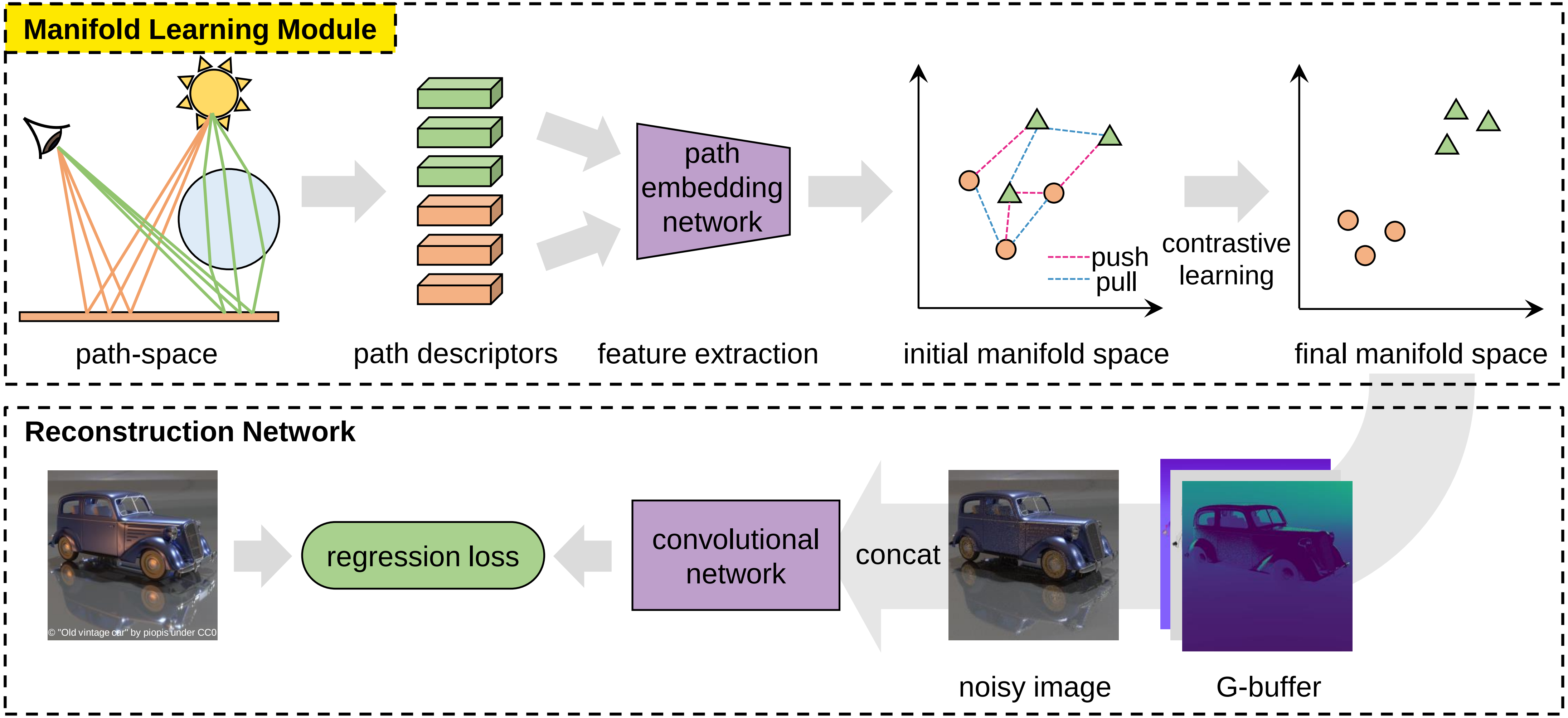


Path-based Monte Carlo Denoising



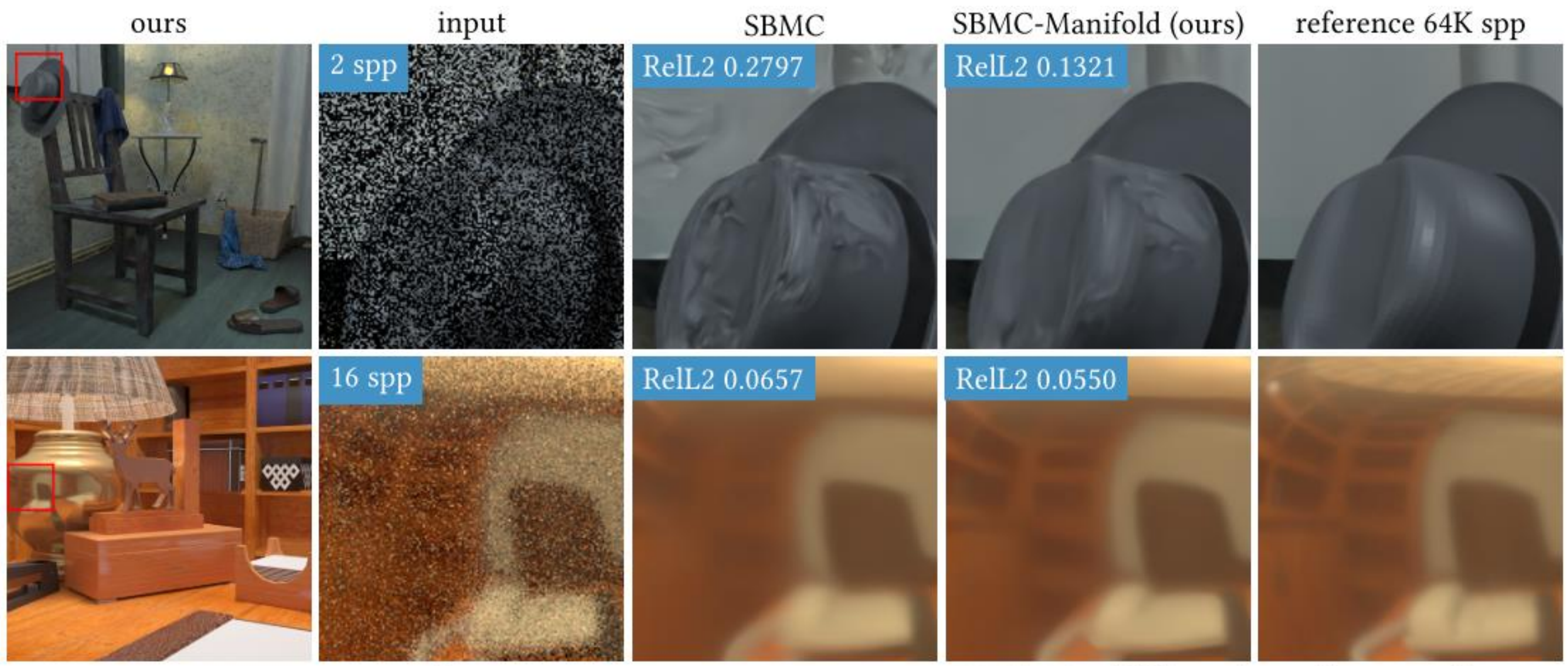
Weakly-supervised Contrastive Learning in Path Manifold for Monte Carlo Image Reconstruction, Cho et al, 2021

Path-based Monte Carlo Denoising



Weakly-supervised Contrastive Learning in Path Manifold for Monte Carlo Image Reconstruction, Cho et al, 2021

Path-based Monte Carlo Denoising



Weakly-supervised Contrastive Learning in Path Manifold for Monte Carlo Image Reconstruction, Cho et al, 2021

Downside of using Richer Features

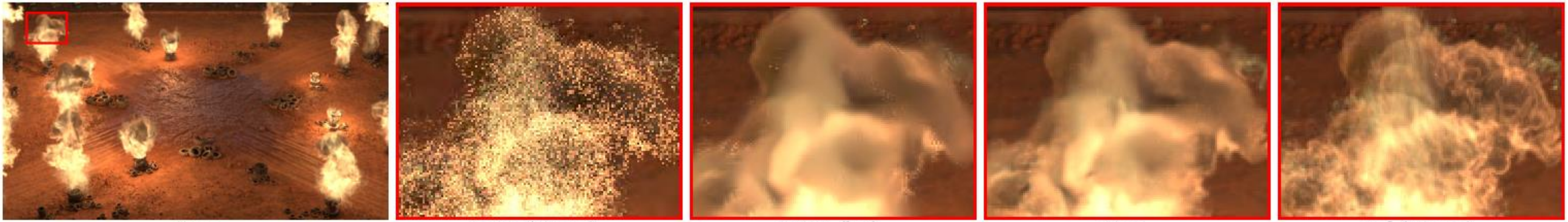
- Using rich features…
  - Gives better denoising result
  - is time & memory consuming

spp	2	4	8	16	32	64
KPCN (geometric)	1.6	1.6	1.6	1.6	1.6	1.6
SBMC (sample)	4.9	6.1	8.6	13.7	24	43.5
Path embed. (additional overhead)	0.64	0.82	1.19	1.99	3.53	6.6

Weakly-supervised Contrastive Learning in Path Manifold for Monte Carlo Image Reconstruction, Cho et al, 2021

Limitations of DL-based Denoising

- Highly dependent on training set
  - Effects not in the training set cannot be well denoised


Ours | Input (32 spp) | NFOR (log) | Ours | Ref. (8K spp)

Further Advancements for Denoising

- **Direct Estimation with Adversarial Training**

- **Temporal Extension**
  - **Using temporal information for denoising**

- **Adaptive sampling with denoising**
  - **Shoot more rays to pixels where it needs to be denoised**

Summary

- Denoising methods for Path Tracing
- Recent Deep learning-based denoising based on Kernel Prediction
  - Using auxiliary features in three-scale (geometry, sample, and path)