Joint SVBRDF Recovery and Synthesis From a Single Image using an Unsupervised Generative Adversarial Network

Yezi Zhao\textsuperscript{1}, Beibei Wang\textsuperscript{2}, Yanning Xu\textsuperscript{1}, Zheng Zeng\textsuperscript{1}, Lu Wang\textsuperscript{1} and Nicolas Holzschuch\textsuperscript{3}

\textsuperscript{1} School of Software, Shandong University
\textsuperscript{2} School of Computer Science and Engineering, Nanjing University of Science and Technology
\textsuperscript{3} Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LJK
Motivation

normal
diffuse
roughness
specular

Substance by Adobe
Our Goal

A lightweight method for recovering real-world material

normal  diffuse
roughness  specular
State of the art

Aittala et al., SIGGRAPH 2016
Reflectance Modeling by Neural Texture Synthesis

Li et al., SIGGRAPH 2017
Modeling surface appearance from a single photograph using self-augmented convolutional neural networks

Deschaintre et al., SIGGRAPH 2018
Single-Image SVBRDF Capture with a Rendering-Aware Deep Network
State of the art

Aittala et al., SIGGRAPH 2016
Reflectance Modeling by Neural Texture Synthesis

Li et al., SIGGRAPH 2017
Modeling surface appearance from a single photograph using self-augmented convolutional neural networks

Deschaintre et al., SIGGRAPH 2018
Single-Image SVBRDF Capture with a Rendering-Aware Deep Network
State of the art

Aittala et al., SIGGRAPH 2016
Reflectance Modeling by Neural Texture Synthesis

Li et al., SIGGRAPH 2017
Modeling surface appearance from a single photograph using self-augmented convolutional neural networks

Deschaintre et al., SIGGRAPH 2018
Single-Image SVBRDF Capture with a Rendering-Aware Deep Network
State of the art

Aittala et al., SIGGRAPH 2016
Reflectance Modeling by Neural Texture Synthesis

Li et al., SIGGRAPH 2017
Modeling surface appearance from a single photograph using self-augmented convolutional neural networks

Deschaintre et al., SIGGRAPH 2018
Single-Image SVBRDF Capture with a Rendering-Aware Deep Network
State of the art

Aittala et al., SIGGRAPH 2016
Reflectance Modeling by Neural Texture Synthesis

Li et al., SIGGRAPH 2017
Modeling surface appearance from a single photograph using self-augmented convolutional neural networks

Deschaintre et al., SIGGRAPH 2018
Single-Image SVBRDF Capture with a Rendering-Aware Deep Network
State of the art

Gao et al., SIGGRAPH 2019
Deep Inverse Rendering for High-resolution SVBRDF Estimation from an Arbitrary Number of Images
State of the art

Gao et al., SIGGRAPH 2019
Deep Inverse Rendering for High-resolution SVBRDF Estimation from an Arbitrary Number of Images

Rely on a plausible starting point of optimizing 😞
State of the art

Zhou et al., SIGGRAPH 2018
Non-stationary texture synthesis by adversarial expansion

Synthesize

[Zhou et al. 2018]
State of the art

Zhou et al., SIGGRAPH 2018
Non-stationary texture synthesis by adversarial expansion

Synthesize Separately

[Zhou et al. 2018]

Zoom-in
State of the art

Zhou et al., SIGGRAPH 2018
Non-stationary texture synthesis by adversarial expansion

Inconsistency between SVBRDF maps 😞
Method overview

Captured image

Our method

normal

diffuse

roughness

specular

Reference

Render

Our method
Imaging setup

- light dir.
- view dir.
- flash
- camera
- mobile phone

For points in tile, light dir. ≈ view dir. ≈ e

Image tile $\mathbf{x}$

Captured image $\mathbf{x}_c$
Generative Adversarial Network (GAN)
\( x \)

Captured image

Generator

SVBRDF maps

Discriminator
Generator

Captured image

$\mathcal{X}$

$\mathcal{X}_c$

Generator

SVBRDF maps

Discriminator

EGSR 2020
Untrained encoder

Visualization of the first 4 layers latent vector from encoder
Generator

Captured Image

SVBRDF maps

Two decoders

normal roughness

diffuse specular

$\mathbf{x}$ $\mathbf{x}_{\text{gt}}$

 latent z

$D_{\rho_{d},\rho_{s}}$

$D_{e_{n,\alpha}}$

latent z

EGSR 2020
Discriminator

- "Real data"
- "Fake data"

Diagram showing a process involving a generator and a discriminator, with inputs and outputs labeled as follows:
- Input: \( x \)
- Output: \( y \)
- "Real data" and "Fake data" labels are shown on the right side of the diagram.
Loss function
Guessed diffuse map

Input image

Guessed diffuse map

Computed as in [AAL16]

[AAL16] Aittala et al.
Reflectance modeling by neural texture synthesis.
Loss function

\[ L_{final} = \lambda L_{GAN}(G, D) + L_d(G), \]

**Adversarial loss**

\[ L_{GAN}(G, D) = \mathbb{E}[\log \text{Dis}(x)] + \mathbb{E}[\log(1 - \text{Dis}(y))], \]

**L1 loss**

\[ L_d(G) = \mathbb{E} [\|\tilde{\rho}_d - \rho_d\|_1]. \]
Loss function

$$\mathcal{L}_{\text{final}} = \lambda \mathcal{L}_{\text{GAN}}(G, D) + \mathcal{L}_{\text{render}}(G)$$

**Adversarial loss**

$$\mathcal{L}_{\text{GAN}}(G, D) = \mathbb{E}[\log \text{Dis}(x)] + \mathbb{E}[\log(1 - \text{Dis}(y))]$$

**L1 loss**

$$\mathcal{L}_{\text{render}}(G) = \mathbb{E}[\|x - y\|_1]$$
\[ \mathcal{L}_{final} = \lambda \mathcal{L}_{GAN}(G,D) + \mathcal{L}_d(G), \]

\[ \mathcal{L}_{final} = \lambda \mathcal{L}_{GAN}(G,D) + \mathcal{L}_{render}(G) \]
Results

Input image: 1632 × 1224
SVBRDF & Render: 3264 × 2448
Results

Input image: 1632 × 1224
SVBRDF & Render: 3264 × 2448
Results

Input image: 1632 × 1224
SVBRDF & Render: 3264 × 2448
Results

512 × 512 Input

1024×1024 SVBRDF maps

normal
diffuse
roughness specular

1024×1024 Render
Results

512 × 512 Input

1024×1024 SVBRDF maps

normal

diffuse

roughness

specular

1024×1024 Render
Results

2048×2048 SVBRDF maps

normal  diffuse  roughness  specular

1024×1024 Render
Results

2048×2048 SVBRDF maps

2048×2048 Render (omit)
Results

2048×2048 Render (omit)

normal  diffuse  roughness  specular

4096×4096 SVBRDF maps
Results

Captured images

Rendered with Arnold
Network Analysis – loss function

![Image showing network analysis and loss functions](image-url)
Network Analysis – generator

- Normal
- Diffuse
- Roughness
- Specular
- Render

**Reference**
- MSE: 0.0046
- MSE: 0.0034
- MSE: 0.0171
- MSE: 0.0118
- MSE: 0.0065

**Our encoder** (untrained encoder)
- MSE: 0.0042
- MSE: 0.0035
- MSE: 0.0241
- MSE: 0.0095
- MSE: 0.0054

**Trained encoder**
- Training time: 1h33m31s
- Training time: 2h0m45s
Limitation

- Our method failed to synthesize textures when the input image has a global structure.
Limitation

- Our method failed to synthesis textures when the input image has a global structure.

- Each input for our method requires individual training, which costs about 3 hours.
Conclusion and Future work

Conclusion

• An unsupervised GAN for joint SVBRDF recovery and synthesis without a large training dataset.
• A two-stream generator to enhance specular component.
• A novel joint loss function for high-quality novel view renderings.

Future work

• Introduce existing knowledge about the material.
Thank you for your attention

The code is available: https://github.com/mengshu1996/SVBRDF-GAN

Yezi Zhao, Beibei Wang, Yanning Xu, Zheng Zeng, Lu Wang and Nicolas Holzschuch