Neural Illumination: Lighting Prediction for Indoor Environments

Authors: Shuran Song, Thomas Funkhouser
Presenters: Clarice Roo, Shivang Soni, Nicolas Buxbaum
Goal:

Estimate a high dynamic range panoramic illumination map of the entire scene from an input image and chosen locale.
Background

- Illumination map - a map that encodes the incident radiance arriving from every direction at the 3D location associated with the selected pixel
- Dynamic range is the ratio of the highest value to lowest value of the pixels in an image
- Low dynamic range (LDR) - dynamic range 1:255
- High dynamic range (HDR) - dynamic range 1:70,000
Motivations and Challenges

- Used to improve lighting in rendering
- Requires comprehensive understanding of the lighting environment
  - 3D location of selected pixel
  - 3D scene geometry to fill in occlusions
  - Distribution of unobserved light sources
  - Missing high dynamic range information

(a) Input: Image+locale
(b) Ground truth: Illumination map
Capture Based Methods for obtaining illumination of an environment

● Physical probe

https://www.pauldebevec.com/Probes/
Related Work - Optimization Based Methods

“Rendering synthetic objects into Legacy Photographs”
Related Work - Learning Based Methods

- Ground truth
- Gardner

- shiny
- rough
- lighting
Problem Formulation

- 3 network method:
  - A geometry estimation network (via depth estimation) (this creates the warped image centered at chosen point)
  - An LDR completion map network (via an understanding of scene illumination and geometry)
  - LDR to HDR network (for improved accuracy)
Training Dataset Generation

Physical Probes $\rightarrow$ Costly and time consuming

Panoramic Datasets: illumination data only at point of capture $\rightarrow$ Limits data quantity

The authors leverage a RGB-D data sets (Matterport3D) to generate ground truth for any locale in the dataset!
Training Dataset Generation

- Matterport3D contains panoramas composed of many densely acquired images
- Illumination maps can be generated at any locale by warping and compositing nearby panoramas
Training Dataset Generation: Selective Locales

- An application is virtual object placement, so locales are chosen according to where a “real” virtual object might logically be placed
  - Densely sample 10 cm above surface mesh
  - Criteria: horizontal surface ($\theta < \cos(\pi/8)$), semantic label “floor” or “furniture”, 10 cm object clearance, 50 cm minimum distance from previous locale
Training Dataset Generation: Forward Mapping

- For each locale, the distance to the closest surface in every direction is estimated
  - This is done using a forward mapping of every image in the panorama to the locale
Training Dataset Generation: Reverse Mapping

- Reconstruct illumination map by resampling input images via reverse mapping
  - Sample pixel values are blended proportionally to their distance from the locale
Training Dataset Generation: Advantages

1. Large variety of sampling sources gives varying illumination environments

1. Multiple illumination maps are generated for a single input image
   a. Model learns spatial dependencies between pixel selections and generated illumination maps
Network Architecture

(a) Prior Work

(b) Our Network Architecture

Geometry Estimation
Network sec. 5.1

Differentiable warping sec. 5.2

Cosine (normal)
L1 loss (plane offset)

LDR completion
Network sec. 5.3

L2 loss
Adversarial loss

LDR to HDR
Network sec. 5.4

L2 loss
Diffuse conv loss

Single LDR image + locale

Geometry Est.

Warped LDR observation

LDR completion

HDR illumination estimation
Geometry Estimation

- This module generates a pixel-wise prediction of geometry represented as a plane equation: \( aX + bY + cZ = d \)
- Well suited for representing the large planar surfaces of indoor environments compared with raw depth values
Geometry Estimation: U-Net Model

- Color image as input
- Surface normal and distance-to-origin plane distance as supervision
  - Calculated directly from Matterport3D depth images
Geometry Estimation: U-Net Model

- Surface normal predictions via a cosine loss
  - Angle between predicted and GT normals
- Plane offset predictions via an l1 loss
  - Difference between predicted and GT plane distance

![Diagram of U-Net Model](image-url)
Geometry Estimation: U-Net Model - PN Layer

- Output from the U-NET is passed to an additional PN layer that converts the normal and plane distances into pixel-wise prediction of 3D locations (via plane equation)
- This layer is fully differentiable and can be trained via an $l_1$ loss
- Enforces consistency between the normal and plane distance outputs
  - Reduces noise seen when reconstructing 3D surfaces

Camera intrinsics:
- $f = F/p$ where $F$ is focal length and $p$ is real pixel size
- $c$ is the optical center

$$\vec{P} = (x, y, z)$$

$$\vec{P} = -\frac{p}{\vec{v} \cdot \vec{n}} \vec{v}, \text{ where } \vec{v} = \left( \frac{x_i - c_x}{f_x}, \frac{y_i - c_y}{f_y}, 1 \right)$$
Geometry Estimation: Examples
Geometry-Aware Warping: Single Layer Module

- This maps the input image pixels to a spherical panoramic image, $h(\varphi, \theta)$, of the light arriving at $l$
- Pixels without a projected value are set to -1

Warped observation

Top: Ground Truth

$l$, the chosen point
Step 2: LDR Panorama Completion

- 2nd module of this system
- Fully Convolutional ResNet50
- Input: mapped observed pixels
- Outputs: dense pixel wise prediction of illumination
Distortion Aware Convolutional Filters
LDR Panorama Completion

- One of the biggest challenges: multi-model nature of the problem
- To address this: along with pixel wise supervision the module is trained with adversarial loss using a discriminator network
Step 3: LDR-to-HDR Estimation

- This module takes predicted LDR illumination as input and outputs a dense pixel-wise prediction of HDR illumination intensities.
LDR-to-HDR Estimation (Cont ..)

- The LDR-to-HDR module learns the mapping function for all pixels from the LDR space to the HDR space. The module is trained with supervision from: 1) a pixel-wise $l_2$ loss and 2) a diffuse convolutional loss $L$. 
1. Pixel-wise $l_2$ loss measures the visual error when re-lighting a perfectly specular surface.

$$L_{l_2} = \frac{1}{N} \sum_{i=1}^{N} (J(i) - J^*(i))$$

Notations:
- $J$: log-scaled image of the final light intensity.
- $J^*$: log-scaled ground truth image of the final light intensity.
- $H$: This is the output HDR illumination map.
- $i$: Target local or specified pixel in an image.
2. Diffuse convolutional loss measures the visual error when re-lighting a perfectly diffuse surface.

\[ L_d = \frac{1}{N} \sum_{i=1}^{N} (D(H(i)) - D(H^*(i))) \]

\[ D(H, i) = \frac{1}{K_i} \sum_{\omega \in \Omega_i} H(\omega) s(\omega)(\omega \cdot \vec{n}_i) \]

- **H**: Expected HDR illumination map produced by LDR-to-HDR module.
- **H^***: Ground truth HDR illumination map
- **D**: Diffuse Convolution function.
- **L_d**: Diffuse convolution loss.
- **\( \Omega_i \)**: hemisphere centered at pixel \( i \).
- **K_i**: the sum of solid angles on \( \Omega_i \).
- **\( \vec{n}_i \)**: the unit normal at pixel \( i \).
- **s(\omega)**: the solid angle for the pixel in the direction \( \omega \).
LDR-to-HDR Estimation (Cont .. )

- Add diffuse convolution loss and pixel-wise $l_2$ loss to compute final loss:

$$L = \lambda_1 L_{l_2} + \lambda_2 L_d$$

where,

$$\lambda_1 = 0.1 \text{ and } \lambda_2 = 0.05.$$
Evaluation:

- Matterport3D dataset of HDR RGB-D is leveraged to generate the training data for the arbitrary locale.

- Training and testing is done by using same train/test split provided in Matterport3D dataset.

- The experiment makes quantitative and qualitative comparisons with the models proposed in the prior work.
Comparisons to state-of-the-art
Comparisons to state-of-the-art
Comparisons to state-of-the-art
Evaluation Metrics:

- **Pixel-wise $l_2$ distance error**: Sum of all the pixel-wise $l_2$ distances between the predicted $H_i$ and the ground truth $H_i^*$ illumination maps.

- **Pixel-wise diffuse convolution error**: Sum of all the pixel-wise $l_2$ distance between $D(H_i)$ and $D(H_i^*)$. 


Comparisons to state-of-the-art

<table>
<thead>
<tr>
<th>Method</th>
<th>$\ell_2$(log)</th>
<th>$\ell_2$</th>
<th>diffuse</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gardner et al. [7]</td>
<td>0.375</td>
<td>0.977</td>
<td>1.706</td>
</tr>
<tr>
<td>Im2Im network</td>
<td>0.229</td>
<td>0.369</td>
<td>0.927</td>
</tr>
<tr>
<td>Nearest Neighbour</td>
<td>0.296</td>
<td>0.647</td>
<td>1.679</td>
</tr>
<tr>
<td>Ours</td>
<td><strong>0.202</strong></td>
<td><strong>0.280</strong></td>
<td><strong>0.772</strong></td>
</tr>
</tbody>
</table>

Table 1. Comparing the quantitative performance of our method to that of Gardner et al. [7] and a nearest neighbour retrieval method.
Modularization v.s. Additional supervision:

<table>
<thead>
<tr>
<th></th>
<th>$\ell_2$ (log)</th>
<th>$\ell_2$</th>
<th>diffuse</th>
</tr>
</thead>
<tbody>
<tr>
<td>without</td>
<td>0.213</td>
<td>0.319</td>
<td>0.856</td>
</tr>
<tr>
<td>with (ours)</td>
<td><strong>0.202</strong></td>
<td><strong>0.280</strong></td>
<td><strong>0.772</strong></td>
</tr>
</tbody>
</table>

Table 2. Effects of modularization.
Comparisons to variants:

LDR + D → HDR (first two modules are omitted)

HDR(wrapped) + D → HDR (last modules are omitted)

<table>
<thead>
<tr>
<th>Method</th>
<th>$\ell_2$ (log)</th>
<th>$\ell_2$</th>
<th>diffuse</th>
</tr>
</thead>
<tbody>
<tr>
<td>LDR → HDR</td>
<td>0.202</td>
<td>0.280</td>
<td>0.772</td>
</tr>
<tr>
<td>LDR + D → HDR</td>
<td>0.188</td>
<td>0.269</td>
<td>0.761</td>
</tr>
<tr>
<td>HDR + D → HDR</td>
<td><strong>0.131</strong></td>
<td><strong>0.212</strong></td>
<td><strong>0.619</strong></td>
</tr>
</tbody>
</table>

Table 3. Comparisons to variants with oracles.
Effect of different losses:

<table>
<thead>
<tr>
<th>loss</th>
<th>$\ell_2$(log)</th>
<th>$\ell_2$</th>
<th>diffuse</th>
</tr>
</thead>
<tbody>
<tr>
<td>l2</td>
<td>0.116</td>
<td>0.235</td>
<td>0.691</td>
</tr>
<tr>
<td>l2+gan</td>
<td>0.224</td>
<td>0.275</td>
<td>0.713</td>
</tr>
<tr>
<td>l2+gan+df</td>
<td>0.131</td>
<td>0.212</td>
<td>0.619</td>
</tr>
</tbody>
</table>

Table 4. Effects of different losses.
Strengths and Weaknesses

Strengths:

- This model is separated into 3 separate modules which increases performance (3 more doable subtasks rather than one larger problem)
- Produces richer/sharper detailed estimations

Weaknesses:

- Produces plausible illumination maps rather than accurate ones when no lights are observed directly in the input
Extensions

- Future work:
  - Include explicit modeling of surface material and reflective properties
  - Explore alternative 3D geometric representations that facilitate out-of-view illumination estimation through whole scene understanding.
Thank You For Listening!