LuminAIRe: Illumination-Aware Conditional Image Repainting for Lighting-Realistic Generation

Jiajun Tang[1,2] **Haofeng Zhong**[1,2] **Shuchen Weng**[1,2] **Boxin Shi**[1,2*]
[1]National Key Laboratory for Multimedia Information Processing
[2]National Engineering Research Center of Visual Technology
School of Computer Science, Peking University
{jiajun.tang, hfzhong, shuchenweng, shiboxin}@pku.edu.cn

Abstract

We present the il**Lumin**ation-**A**ware conditional **I**mage **Re**painting (LuminAIRe) task to address the unrealistic lighting effects in recent conditional image repainting (CIR) methods. The environment lighting and 3D geometry conditions are explicitly estimated from given background images and parsing masks using a parametric lighting representation and learning-based priors. These 3D conditions are then converted into illumination images through the proposed physically-based illumination rendering and illumination attention module. With the injection of illumination images, physically-correct lighting information is fed into the lighting-realistic generation process and repainted images with harmonized lighting effects in both foreground and background regions can be acquired, whose superiority over the results of state-of-the-art methods is confirmed through extensive experiments. For facilitating and validating the LuminAIRe task, a new dataset CAR-LUMINAIRE with lighting annotations and rich appearance variants is collected.

1 Introduction

Advanced image editing is in high demand across a multitude of applications, *e.g.*, old photo colorization [78, 32, 68], damaged image restoration [48, 73, 72], and artistic style transfer [22, 35, 70]. Recently, conditional image repainting (CIR) [67, 66, 58] has emerged as an innovative research topic, proven effective in controllable image editing while "freeing" users from the necessity of expert proficiency and retaining the "freedom" to actualize their creative visions for image modification. By utilizing provided attributes or textual descriptions, fine-grained strokes, and Gaussian noise to separately represent colors, contours, and texture conditions, users could insert generative objects with desired appearances in specified image positions, as shown in the blue line of Fig. 1.

Although CIR methods have made great progress in synthesizing photo-realistic and visually-pleasing conditional images by avoiding gradient vanishing pitfall [67], adopting flexible condition representation [66], and designing condition fusion modules [58], there is still a crucial element missing from the CIR task: making the synthesized results harmonized with the illumination of the scene, *e.g.*, spatially-varying dark and bright regions in accordance to the lighting condition in the background, physically-accurate highlight effects for highly-specular surfaces (shining objects), and perceptually-realistic shadow avoiding "floating objects" artifacts, as shown in the lower right example of Fig. 1.

Specifically, existing CIR methods handle image harmonization purely in 2D image space by estimating a pixel-wise color tone transformation of the repainted regions from the background regions. Current approaches use semantic parsing maps as "geometry" representations and do not exploit the lighting information contained in given background images, which prevents them from having

*Corresponding author.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

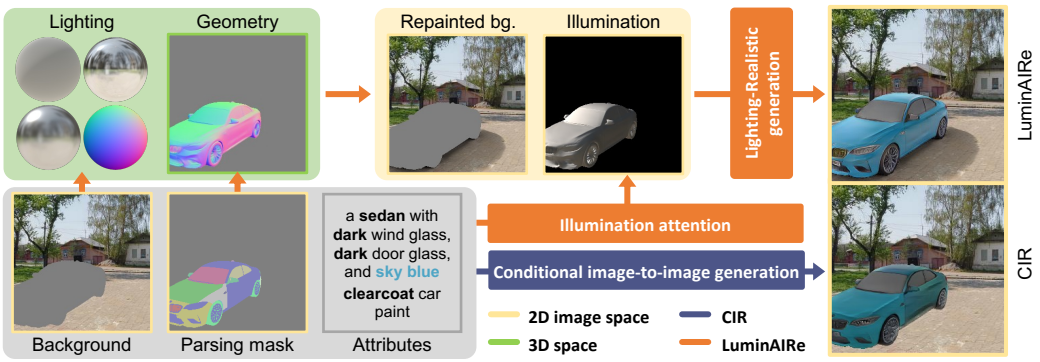


Figure 1: Illustration of proposed LuminAIRe task and result. Compared with the previous CIR task [58] (blue line) which takes all condition inputs[2] at once conducting a conditional image-to-image generation purely in 2D image space, LuminAIRe (orange line) exploits 3D lighting and geometry information and repaints both foreground (fg.) and background (bg.) regions via a lighting-realistic generation process. The 3D information is transformed back to 2D image space in the form of an illumination image, with the desired reflective properties obtained from an illumination attention module. LuminAIRe handles *(i)* surface shading, *(ii)* highlight effects, and *(iii)* realistic shadow in the repainted image (top right).

awareness of physically-based lighting in 3D space. To introduce physically-correct 3D lighting instead of hallucinated lighting effects into CIR results, there remain some major challenges: *(i)* The lighting condition in 3D space should be extracted from the limited field of view (limited-FoV) 2D LDR images; *(ii)* the lighting condition should be physically-correctly transformed back into 2D image space; *(iii)* a dataset suitable for learning-based solutions to the proposed task is needed.

To achieve *lighting-realistic generation* within the CIR pipeline in an illumination-aware manner, we hereby propose the task of il**Lumin**ation-**A**ware conditional **I**mage **Re**painting, denoted as **LuminAIRe**. We first lift geometry conditions from 2D parsing maps to 3D[1] normal maps using learning-based shape priors and estimate lighting conditions from limited-FoV LDR background images by designing a proper parametric representation. Then, we use physically-based reflection models to render *illumination candidate images* to capture possible lighting effects in 2D image space. With the help of *illumination attention module*, surface regions with different reflective properties are learned to adopt correct lighting effects in the resulting appearance. A dataset containing rich geometry and lighting annotations with abundant object variants is collected to facilitate the learning-based solution of the LuminAIRe task. As far as we know, we are the first to emphasize illumination-awareness in the image editing task of conditional image repainting.

Our contributions can be summarized as follows:

- introducing a new task of il**Lumin**ation-**A**ware conditional **I**mage **Re**painting (**LuminAIRe**) by exploiting the lighting information from background images;
- designing a full LuminAIRe pipeline that represents, extracts, converts, and injects lighting information to acquire more realistically repainted results; and
- collecting a new dataset CAR-LUMINAIRE with rich material and lighting condition variants for facilitating and validating the LuminAIRe task.

2 Related Work

Our method aims at introducing physical lighting constraints into generative image synthesis pipelines. In this section, we briefly review relevant works first and then discuss the relationships to our task.

Controllable image synthesis. Researchers have presented numerous works to synthesize images under the guidance of diverse user-provided conditions, *e.g.*, synthesizing specific object with category label [10, 43, 45, 75], transferring the texture from paintings to daily photos [22, 16, 35, 70], restoring

[1]Strictly speaking, the normal maps are in 2.5D. Here we use 3D to simply distinguish it from 2D.
[2]In this paper, attributes are shown in templated sentences for formatting, and texture is omitted for simplicity.

the colors of old photos [11, 12, 69, 68], and directly generating images from text descriptions [51, 50, 55, 56]. Recently, with the development of the condition injection mechanism [31, 47, 80, 34], researchers explore to control synthesized images with multiple cross-modality conditions, *e.g*., condition guided image inpainting [44, 55], controllable person synthesis [54, 65], and inversion-based style transfer [79]. However, few works focus on synthesizing images strictly following lighting conditions. Following DIH-GAN [6] that considers introducing illumination estimation in harmonization task that adjusts the highlight of the inserted given object, we further explore the lighting condition in synthesizing illumination-consistent objects under the guidance of multiple cross-modality conditions.

Conditional image repainting and image harmonization. Conditional image repainting (CIR) aims at synthesizing reasonable visual content on an existing image, where the generated visual content should both meet the requirement of the user-provided conditions (*e.g*., color, geometry, and texture) and in harmonization with the existing background image. The first CIR task is proposed in MISC [67] for person image synthesis, where the foreground person image is synthesized first and then composited with the background. Weng *et al*. [66] design the semantic-bridge attention mechanism which allows more freely expressed color conditions by the users in text. UniCoRN [58] breaks the two-stage dependency and proposes a unified architecture that achieves more visually pleasing results. Despite recent achievements made by previous works in condition consistency, existing CIR models suffer from the issue of illumination inconsistency: although techniques such as color tone transform are applied, the lighting from the given background and on the generated visual contents often differ a lot, making lighting effects in the image rather unrealistic, such as incorrect shading, highlights, and shadows. In this paper, we address this issue by exploiting lighting and shape constraints in 3D space, which allows a more physically-correct rendering processing for generating lighting effects. Image harmonization methods [23, 24, 25, 14, 42, 62, 59], with a similar goal of CIR to realistically composite image foreground and background regions, have focused on illumination harmonization recently [6, 8]. However, this thread of works has poor control of visual content in foreground regions and may fail to preserve the color tone in background regions as they were.

Lighting representation and estimation. Achieving illumination-aware synthesis/repainting requires appropriate lighting representation and estimation from images. Lalonde *et al*. [37] is the first to use shadows, shading, and sky appearance variations observed in the image to infer outdoor lighting. A physics-based Hošek-Wilkie (HW) sky model [29, 30] is proposed to recover HDR parameters for deep outdoor lighting estimation [28]. A more weather-robust Lalonde-Matthews (LM) model [38, 77] is then proposed to cover more comprehensive lighting conditions in the outdoor environment. More recently, a learning-based lighting representation [27] is used on a large sky panorama dataset [36] with an autoencoder network. The encoder-decoder framework is further proposed [39] to estimates lighting as a spherical HDR lighting map. HDSky [74] and SOLD-Net [60] disentangle several physically meaningful attributes into separate learned latent spaces by hierarchical autoencoders and make the estimation editable. Parametric models such as spherical harmonic (SH) coefficients [7, 21] and spherical Gaussian (SG) [19, 40] are also widely used, especially in indoor scenes. Gardner *et al*. [20], NeurIllum [57], and SOLID-Net [81] design sophisticated networks to hallucinate the missing parts in the panoramic view and predict lighting as environment maps. 3D volumetric lighting representations are also widely used in recent works, which facilitate the lighting-realistic scene editing for indoor [41] and outdoor [64] scenes, however heavily require computation resources. Considering the demand for lighting-realistic generation, we propose a parametric lighting representation for outdoor scenes that is both easy to predict and simple to use.

3 Problem Formulation

For self-containedness, we briefly review the CIR formulation before introducing ours.

3.1 Preliminaries about CIR

The previous CIR tasks [66, 67, 58] aim at generating the repainted image y^{r} by repainting certain regions in an image $x \in \mathbb{R}^{3\times H\times W}$ according to user-specified conditions in different modalities: x^{g}, x^{p}, x^{c}, and x^{b} for the "geometry", "texture", "color", and background conditions respectively.

In their works, the "geometry" condition $x^{\mathrm{g}} \in \mathbb{L}^{N_{\mathrm{g}}\times H\times W}$ is a binary semantic parsing mask, where N_{g} is the number of possible parts of the visual content to be repainted and $\mathbb{L} = \{0, 1\}$; the "texture"

condition $x^{\mathrm{p}} \sim \mathcal{N}(0,1)$ is a Gaussian noise; the "color" condition can be represented as attributes $x^{\mathrm{c}} \in \mathbb{L}^{N_{\mathrm{c}} \times N_{\mathrm{v}}}$ or text descriptions $x^{\mathrm{c}} = \{x_t^{\mathrm{c}}\}_{t=1}^{N_{\mathrm{L}}}$, where N_{c}, N_{v}, and N_{L} represent the numbers of attributes and available choices, and the length of the user-inputted sentences, respectively; the background condition $x^{\mathrm{b}} \in \mathbb{R}^{3 \times H \times W}$ is the image of background region with respect to the repainted region as foreground region, *i.e.*, $x^{\mathrm{b}} = (1-m) \odot x$, where the binary mask m indicating foreground region can be directly acquired from the parsing mask x^{g}, as shown in lower left of Fig. 1.

The repainted image y^{r} can be further decomposed as a blending of repainted foreground image y^{f} and repainted background image y^{b}:

$$y^{\mathrm{r}} = m \odot y^{\mathrm{f}} + (1-m) \odot y^{\mathrm{b}}. \tag{1}$$

Previous works assume unchanged background region, *i.e.*, $y^{\mathrm{b}} = x^{\mathrm{b}}$, leaving the key question of CIR tasks as generating realistic foreground region y^{f} constrained by given conditions:

$$y^{\mathrm{f}} = \mathrm{F}^{\mathrm{G}}(x^{\mathrm{g}}, x^{\mathrm{p}}, x^{\mathrm{c}}, x^{\mathrm{b}}), \tag{2}$$

where previous works ignore clues in 3D space and implement the generation pipeline F^{G} as a *conditional image-to-image generation* purely in 2D image space. To make the repainted image harmonized as a whole, previous works [58, 67] design additional harmonization modules to adjust the color tone of intermediate repainting result based on clues in x^{b}.

3.2 Formulation of LuminAIRe

However, the image-based harmonization modules have limited representation ability for complex lighting effects (*e.g.*, varying shading and shiny surfaces) due to a lack of 3D representation. Besides, directly using x^{b} as y^{b} in Eq. (1) may neglect possible light transport effects (*e.g.*, shadows) introduced by the repainted region as its corresponding behaviors in the 3D real world might be.

As illustrated by the rendering equation [33], a physically-correct and -realistic appearance of an object is derived from its *geometry*, *reflective property*, and omnidirectional *environment lighting* in 3D space. Therefore, to make the repainted image y^{r} more *lighting-realistic*, the repainted foreground y^{f} should also be conditioned by the lighting condition L and geometry condition G in 3D space:

$$y^{\mathrm{f}} = \mathrm{F}^{\mathrm{F}}(x^{\mathrm{g}}, x^{\mathrm{p}}, x^{\mathrm{c}}, x^{\mathrm{b}}, L, G). \tag{3}$$

Given L and G in 3D space, a proper 2D representation x^{i} containing both the information from L and G should be derived for compatibility with current image generation architectures:

$$x^{\mathrm{i}} = \mathrm{R}^{\mathrm{i}}(L, G), \tag{4}$$

and then the *lighting-realistic generation* for foreground y^{f} can be rewritten as:

$$y^{\mathrm{f}} = \mathrm{F}^{\mathrm{F}}(x^{\mathrm{g}}, x^{\mathrm{p}}, x^{\mathrm{c}}, x^{\mathrm{b}}, x^{\mathrm{i}}). \tag{5}$$

Similarly, the repainted background y^{b} should also be conditioned on x^{i} to recover lighting effects:

$$y^{\mathrm{b}} = \mathrm{F}^{\mathrm{B}}(x^{\mathrm{b}}, x^{\mathrm{i}}). \tag{6}$$

The limited-FoV background image x^{b} itself is a partial observation of environment lighting and thus can provide clues about L. Therefore, the lighting condition L can be inferred in the form of:

$$L = \mathrm{F}^{\mathrm{L}}(x^{\mathrm{b}}). \tag{7}$$

Similarly, by finding the shape priors of certain types of objects, the 3D geometry condition G can be lifted from its "2D flattened version", *i.e.*, parsing mask x^{g}:

$$G = \mathrm{F}^{\mathrm{Geo}}(x^{\mathrm{g}}). \tag{8}$$

Moreover, in our LuminAIRe formulation, we extend the attributes x^{c} beyond colors, which allows the users to describe the *reflective property* and have control over the lighting effects of repainting results. A sample of attributes is shown as the **bold text** in the lower left of Fig. 1.

As aforementioned, both the repainted foreground y^{f} and background y^{b} are given by the *lighting-realistic generation* in our LuminAIRe formulation, which leads to more realistic and harmonized results than traditional CIR pipelines [58], as shown in Fig. 1.

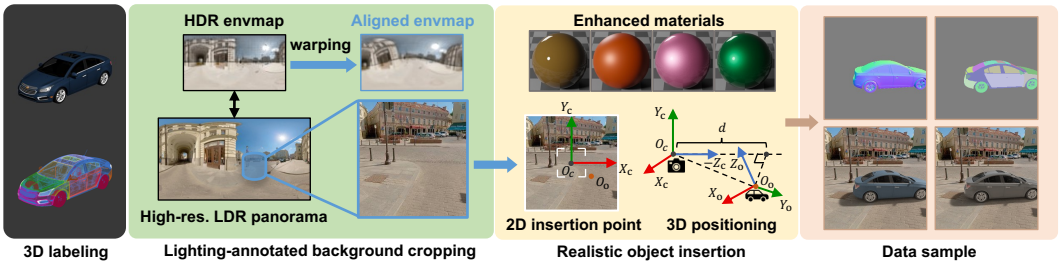


Figure 2: Data preparation process of the CAR-LUMINAIRE dataset.

4 Data Preparation

To tackle the data shortage issue, we create the first dataset suitable for the LuminAIRe task, named CAR-LUMINAIRE, with its data preparation process and data sample shown in Fig. 2.

3D car models with hierarchical semantic labeling. Collecting large-scale real data for learning-based LuminAIRe methods is infeasible since the geometry and lighting capture in 3D space requires specialized equipment and extensive human labor. Therefore, here we resort to computer graphics techniques to create photo-realistic synthetic data. The cars are chosen as the foreground objects for the obviousness of lighting effects and the availability of high-quality synthetic models. We collect 198 detailed 3D car models in 17 different categories from online model stores [2, 4] and then label the parts of the models in 3D space, which allows us to get the accurate parsing mask in 2D image space from any viewpoint. Following the common structure of vehicles, we divide the car models into 35 semantic part labels. The part labels are organized in a hierarchical way (*e.g.*, the *door window* is a sub-part of the *door*) to accommodate car models in different granularity. Besides 3D labeling, we manually adjust the scales of each model to fit the real-world dimensions.

Background images with lighting annotations. Then we prepare background images with known lighting annotations. Here we use the SUN360-HDR dataset [27, 76], which contains HDR panoramic environment maps (envmaps) corresponding to the LDR panoramas of outdoor scenes in the SUN360 dataset [71]. Limited field-of-view (limited-FoV) background images are cropped from the LDR panoramas with virtual cameras of randomized FoVs and camera poses. For each cropped background image, the corresponding HDR envmap in the SUN360-HDR dataset [27, 76] is warped to align with the viewing direction of the virtual camera. Background images unsuitable for realistic object insertion are manually filtered out, leaving 1,321 images of diverse scenes and lighting conditions.

Enhanced data rendering with realistic placement. For each background image, we randomly select insertion points within the central region of the "placeable flat ground" marked by an off-the-shelf segmentation toolbox [15]. Then, for each 2D insertion point in the image, we calculate the relative transformation from the camera coordination O_c to the local coordination of the object O_o from the depth d and the normal Z_o estimated by depth [52, 53] and normal [5] estimation methods. With the aligned envmaps and the ray-tracing based Blender [3] Cycles rendering engine, physically-correct lighting effects can be rendered into the composited images. In the rendering process, besides the original materials of the models, several physics-based rendering (PBR) car paint materials are randomly applied for more appearance variants, especially in lighting effects; besides, the inserted models are randomly rotated around Z_o axis for more geometry variants. The rendered images are filtered to ensure reasonable pixel portions of both foreground and background regions. At last, 52,581 composited images at the resolution of 256×256 are collected, accompanied by parsing mask and normal map annotations, as shown in the data sample of Figure 2.

5 Method

To realize the LuminAIRe formulation, we first estimate 3D lighting and geometry from background images and parsing masks (Sec. 5.1). Then the lighting information is injected into the lighting-realistic generation process as illumination images (Sec. 5.2). By further introducing hierarchical labeling enhancement (Sec. 5.3), our method can generate reasonable results even with coarse-level parsing masks. Our pipeline is shown in Fig. 3, with detailed network architectures and loss functions for network modules in supplementary materials.

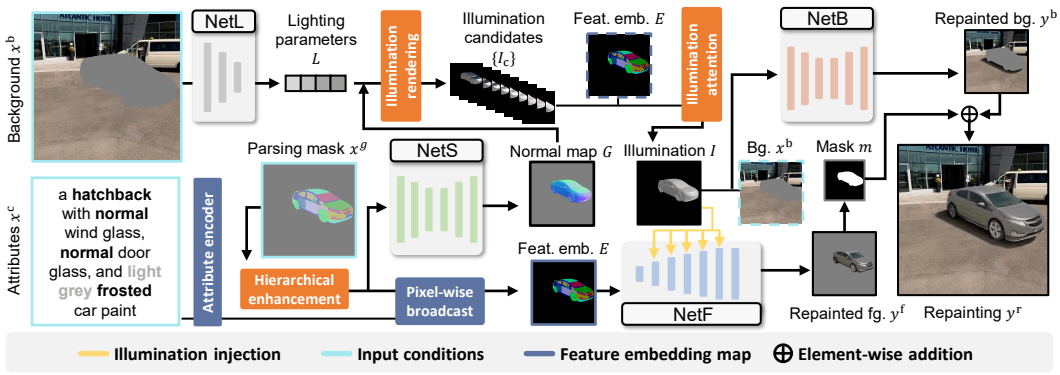


Figure 3: Overview of LuminAIRe pipeline.

Figure 4: Our lighting representation can capture most of the lighting effects in all weather conditions.

5.1 Estimating 3D Information with Learning-based Priors

Our CAR-LUMINAIRE dataset consists of outdoor scene images, where the lighting can be approximately decomposed into the high-frequency sunlight and the low-frequency ambient light [60]. Accordingly, we model the lighting condition L as the addition of a directional light and a 2-nd order spherical harmonics (SH) lighting, which can be specifically described as lighting parameters:

$$L = \{z_{\text{vis}}, z_{\text{int}}, z_{\text{ang}}, c_{\text{sun}}, l_{\text{sun}}, \sigma_{\text{SH}}\}, \tag{9}$$

where $z_{\text{vis}} \in \{0, 1\}$ is the sun visibility, z_{int} is the intensity of sunlight, z_{ang} describes the "size" of the sun (in solid angle formally), $c_{\text{sun}} \in \mathbb{R}^3$ is the normalized sun color in RGB channels, $l_{\text{sun}} \in \mathbb{R}^2$ indicates the sun position, and $\sigma_{\text{SH}} \in \mathbb{R}^{3\times 9}$ is the 2-nd order SH coefficients for RGB channels.

As shown in Fig. 4, the parametric representation[3] in Eq. (9) can well fit real-world lighting in sunny, cloudy, and low light conditions. On the other hand, the proposed parametric lighting representation is convenient for network prediction. Here we design a *NetL* to serve as F^{L} in Eq. (7), where l_{sun} is estimated by a classification task and other parameters are estimated by regression tasks. To apply our method to other types of background scenes, specifically tailored lighting representations can be directly adopted, without modification to our underlying formulation of LuminAIRe.

For 3D geometry, we use the normal map $G \in \mathbb{R}^{3\times H\times W}$ as the representation where each pixel indicates the surface normal direction $\boldsymbol{n}$ at that surface point in 3D space. For certain types of objects, there exist strong shape priors (such as sedans and hatchbacks), which can be learned in a supervised way. Similarly, a *NetS* of encoder-decoder structure is further proposed to serve as F^{Geo} in Eq. (8).

5.2 Injecting Lighting Information using Illumination Images

To bridge the 3D lighting and geometry with 2D images, the rendering equation [33] is a handy tool to serve as R^{i} in Eq. (4), which physically models the image formation process as the light reflection:

$$L_{\text{o}}(\boldsymbol{\omega}_{\text{r}}) = \int_{\Omega_{\boldsymbol{n}}} L_{\text{i}}(\boldsymbol{\omega}_{\text{i}}) f_{\text{r}}(\boldsymbol{\omega}_{\text{i}}, \boldsymbol{\omega}_{\text{r}})(\boldsymbol{n} \cdot \boldsymbol{\omega}_{\text{i}}) \text{d}\boldsymbol{\omega}_{\text{i}}, \tag{10}$$

where $L_{\text{i}}(\boldsymbol{\omega}_{\text{i}})$ is the environment lighting from direction $\boldsymbol{\omega}_{\text{i}}$, $L_{\text{o}}(\boldsymbol{\omega}_{\text{r}})$ is the reflected lighting toward direction $\boldsymbol{\omega}_{\text{r}}$, $\Omega_{\boldsymbol{n}}$ is the visible hemisphere determined by surface normal $\boldsymbol{n}$, and $f_{\text{r}}(\boldsymbol{\omega}_{\text{i}}, \boldsymbol{\omega}_{\text{r}})$ describes the reflective properties of all possible combination of incoming and outgoing directions.

[3]Lighting parameters are converted back to tone-mapped HDR environment maps for visualization.

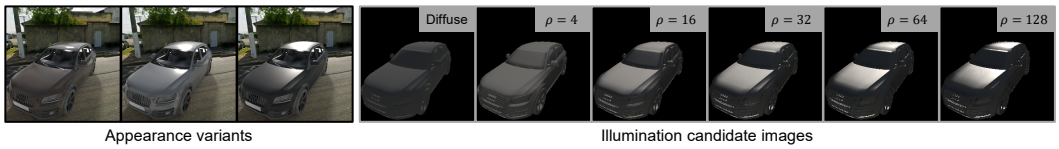


Figure 5: Our illumination candidate images can cover realistic lighting effects in appearance variants.

For a certain image pixel with the known camera viewing direction $\boldsymbol{v}$, ideally, the pixel intensity can be calculated as $L_{\mathrm{o}}(-\boldsymbol{v})$, and accurate lighting effects can be calculated as *illumination images*.

However, with L and G estimated from input conditions, $f_{\mathrm{r}}(\boldsymbol{\omega}_{\mathrm{i}}, \boldsymbol{\omega}_{\mathrm{r}})$ still remains unknown. Therefore, in a similar spirit to Gao *et al*. [18] and Pandey *et al*. [46], instead of directly calculating the actual illumination image, we use a set of uncolored "standard materials" as f_{r} in Eq. (10) and render corresponding *illumination candidate images* $\{I_{\mathrm{c}}\}$. For the physics-based rendering of $\{I_{\mathrm{c}}\}$, we use the Lambertian refectance model $f_{\mathrm{diff}}(\boldsymbol{\omega}_{\mathrm{i}}, \boldsymbol{\omega}_{\mathrm{r}}) = {}^{1}/_{\pi}$ and normalized Blinn-Phong model [9] $f_{\mathrm{spec}}(\boldsymbol{\omega}_{\mathrm{i}}, \boldsymbol{\omega}_{\mathrm{r}}) = {}^{(\rho+4)(\boldsymbol{n}\cdot\boldsymbol{h})^{\rho}}/_{8\pi}$ with M different values of roughness ρ, where $\boldsymbol{h} = {}^{\boldsymbol{\omega}_{\mathrm{i}}+\boldsymbol{\omega}_{\mathrm{r}}}/_{||\boldsymbol{\omega}_{\mathrm{i}}+\boldsymbol{\omega}_{\mathrm{r}}||}$ is the half vector. At last, we have $\{I_{\mathrm{c}}\} = \{I_{\mathrm{diff}}\} \cup \{I^{\rho_i}_{\mathrm{spec}}\}_{i=1}^{M}$.

As shown in Fig. 5, most lighting effects in different appearance variants can be covered by the linear combinations of the pre-computed $\{I_{\mathrm{c}}\}$. However, it's worth noting that the correspondence of the appearance image and $\{I_{\mathrm{c}}\}$ may vary pixel-wisely (*e.g*., the tires, hood, and windshield have different reflective properties thus different lighting effects). Accordingly, we design an *illumination attention* module $\mathrm{A^I}$ to estimate the combination coefficient maps $C_{\mathrm{I}} = \mathrm{A^I}(E)$ for each image pixel, where E is the feature embedding map containing information of both part labels and part-associated attributes in a pixel-aligned way. After the illumination image I derived as $I = \sum_{i=1}^{M+1} C^i_{\mathrm{I}} \odot I^i_{\mathrm{c}}$, which covers lighting effects of parts with different materials , we use I as x^{i} in Eq. (5) and conduct lighting-realistic generations of foreground and background regions using our proposed *NetF* and *NetB* respectively. For *NetF*, we adopt the network backbone of $\mathrm{F^G}$ in UniCoRN [58], and the illumination image I is injected in a similar way as other conditions in 2D image space at different resolutions. The *NetB* is also an encoder-decoder architecture, serving as $\mathrm{F^B}$ in Eq. (6). We adopt the same loss functions for *NetF* as used in UniCoRN [58].

5.3 Generating Realistic Results from Coarse Parsing Masks

As mentioned in Sec. 4, the parsing masks in our CAR-LUMINAIRE dataset can be very coarse, which also reflects the possible application scenarios when the user only specifies interested parts. Previous CIR formulations may fail to generate realistic results in regions without fine-grained labels since their generation follows a strictly pixel-wise semantic mapping between labels and images. We hereby introduce a *hierarchical labeling enhancement*: randomly coarsening the input parsing mask at training time (*e.g*., *door glass* label becomes *door* label) and encouraging the fine-grained parts (*door glass*) to be generated. Besides, the part-associated attributes of lower-level parts (*door glass*) should be also associated with their upper-level parts (*door*) to avoid loss of condition in attributes x^{c}, which can be done by modifying the association matrix [58] $A \in \mathbb{L}^{N_{\mathrm{c}} \times N_{\mathrm{g}}}$ accordingly.

6 Experiments

In this section, we conduct comparisons with state-of-the-art methods and validate our design with an ablation study and a robustness test. Please see supplementary materials for implementation details.

6.1 Comparison with State-of-the-art Methods

Baseline methods. We conduct quantitative and qualitative comparisons with three state-of-the-art CIR methods (**UniCoRN** [58], **Weng *et al*.** [66], and **MISC** [67]) and a most-relevant conditional image generation method (**Pavllo *et al*.** [49]). Among them, modifications are made for **Pavllo *et al*.** [49] and **Weng *et al*.** [66] to accept conditions represented as attributes.

Quantitative metrics. Following previous work [58], we adopt Fréchet inception distance (FID) [26] for assessment of perception quality, R-precision [66] for assessment of alignment between generated

images y^{r} and given attributes x^{c}, and M-score [61] for assessment of authenticity. We use the latest manipulation detection model [17, 13] for calculating the M-score [61]. We also report the structural similarity index (SSIM) [63] for comparing the major image structure with the reference image.

Table 1: Comparison with the state-of-the-art methods and variants of our proposed method. Quantitative evaluation scores and user study results are shown. ↑ (↓) means higher (lower) is better. "Real." and "Har." are abbreviations of "Realistic" and "Harmonized".

Method	Quantitative Evaluation				User Study	
	FID ↓	R-prcn ↑	M-score ↓	SSIM ↑	Real. ↑	Har. ↑
MISC [67]	53.84	34.94%	31.23	0.6660	0.25%	0.28%
Weng *et al*. [66]	38.12	46.66%	30.84	0.6697	0.85%	0.85%
Pavllo *et al*. [49]	9.29	56.98%	36.77	0.7050	43.00%	36.72%
UniCoRN [58]	11.55	62.13%	29.72	0.6940	7.78%	9.90%
LuminAIRe (Ours)	**4.62**	**74.13%**	**13.68**	**0.7211**	**48.12%**	**52.25%**
Ours-H	5.83	63.27%	13.97	0.7163	—	—
Ours-HA	6.31	63.94%	13.95	**0.7214**	—	—
Ours-HAI	8.00	62.13%	15.83	0.7054	—	—

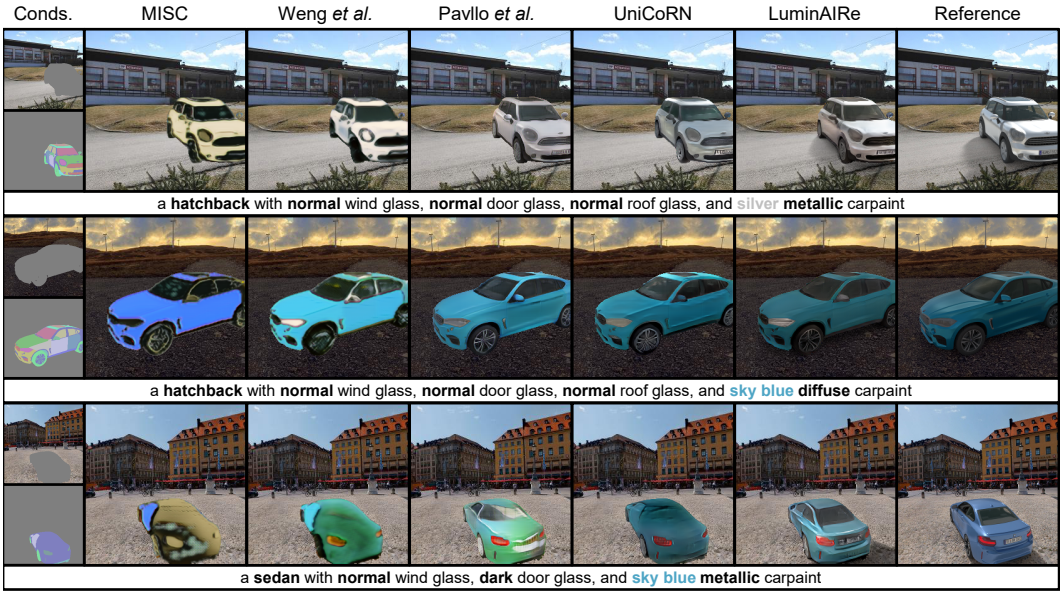


Figure 6: Qualitative comparison with the state-of-the-art methods, with given conditions (conds.).

The scores in Tab. 1 and the second and the third columns of Fig. 6 show that results of **MISC** [67] and **Weng *et al.*** [66] are far from lighting-realistic with "crayon-drawing-like" appearances, since the color tone transform is not applied [66], or conducted in a two-phase manner [67]. As shown in the fourth column of Fig. 6, **Pavllo *et al.*** [49] tend to generate foreground regions in flat shadings with fewer texture patterns, which makes its results generally look reasonable when only focusing on foreground regions or in low light or cloudy scenes (as indicated by the FID and user study results), but computer vision models can easily find the disharmony due the sharp boundaries between foreground and background regions [58], as also indicated by the worst M-score. **UniCoRN** [58] fails to generate correct lighting effects from its unified color tone transform (the first row), therefore tends to hallucinate highlights at the top of cars regardless of lighting in background regions (the second row). The hallucinated lighting effects along with the undesired texture pattern on car bodies drastically damage the perceptual preferences, as confirmed by the FID score and user study results in Tab. 1. **LuminAIRe** generates realistic lighting effects close to the reference images in both sunny (the first and the third rows) and cloudy (the second row) scenes of specified materials and even when a coarse-level parsing mask is given (the third row), with a large margin in all quantitative metrics compared with baseline methods. **LuminAIRe** also learns to avoid the undesired texture pattern with the hints of the smoothly varied shading in the illumination images (Fig. 7).

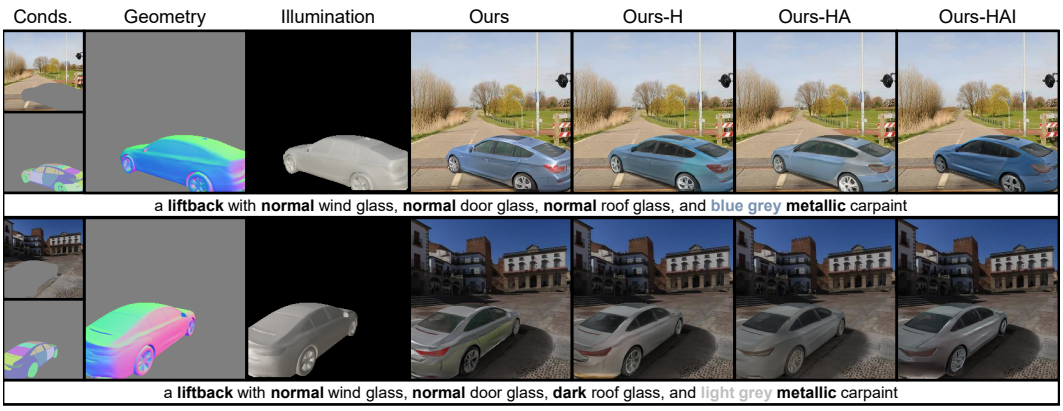


Figure 7: Ablation study for three variants of our proposed method, with given conditions (conds.).

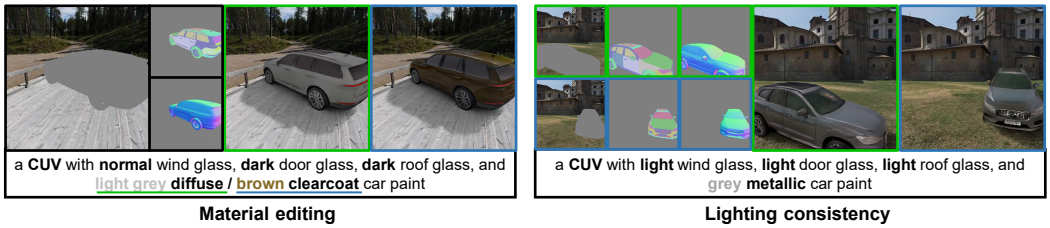


Figure 8: Our method can generate realistic lighting effects with given materials (left), which are consistent across different geometry conditions (right). Green and blue boxes mark individual cases.

6.2 Evaluations

Ablation study. We conduct an ablation study with three variants of **Ours**: *(i)* **Ours-H**, *(ii)* **Ours-HA**, and *(iii)* **Ours-HAI**, where "-H", "-A", and "-I" mean disabling the hierarchical labeling enhancement, the illumination attention, and the illumination injection for the foreground, respectively.

The hierarchical labeling enhancement is confirmed helpful in generating realistic results with coarse-level parsing masks, as shown in Fig. 7 and the third row of Fig. 6, where **Ours** generates more consistent and better repaintings at regions with no specified part labels (marked in blue purple), which is also demonstrated by the FID and R-prcn score in Tab. 1. The second row of Fig. 7 shows an example where the lack of illumination attention module wrongly renders a diffuse appearance, with further evidence from the drop of FID from **Ours-H** to **Ours-HA** in Tab. 1. It's quite obvious from Tab. 1 and Fig. 7 that the illumination injection helps foreground generation by comparaing **Ours-HA** and **Ours-HAI**. From **UniCoRN** to **Ours-HAI**, the improvements in FID score and M-score validate the contribution of the lighting-realistically generated background.

Besides, **Ours-HA** gets an unexpectedly good SSIM score. It's possibly because a slight misalignment of lighting effects (especially highlights) due to errors in lighting or geometry estimation would lead to a considerable drop in the SSIM score (which honestly measures the pixel-wise difference) but with very little harm to the lighting-realistic perception (as indicated by the FID and M-score).

User study. We also conduct a user study with 20 volunteers on the Amazon Mechanical Turk [1] platform, where 200 sets of results randomly drawn from the test set are shown and volunteers are asked to choose one in each set with *(i)* the most *realistic* foreground and *(ii)* the most *harmonized* lighting. The results of the user study in Tab. 1 are basically aligned with the trending of FID and SSIM scores in quantitative evaluation, showing that repainting results of our **LuminAIRe** are most favored subjectively, with a greater lead in realistic and harmonized lighting perception.

Robustness Test. Fig. 8 shows the robustness of our method to varying materials and geometry conditions, where different materials and geometry conditions are correctly handled with realistic lighting effects accordingly and consistently generated. To test the robustness of our method to varying parsing masks (*e.g.*, casually-drawn parsing masks), we compare in Fig. 9 the repainting

results of from the input parsing masks before and after the disturbing, where the boarders are randomly extended and the inner structures are coarsened. To test the robustness of our method to varying lighting conditions, we conduct an experiment where the estimated lighting conditions are rotated clockwise while all other conditions are left unchanged. The results in Fig. 10 show that our method correctly handles most of the lighting rotations in the sense of the lighting effects on the foreground objects and the shadow effects in the background regions. The repainting results in the second column with no lighting conditions given ("No light") further validate the effectiveness of our illumination injection module. To test the robustness of our method to varying background conditions, we also show the results of in-the-wild examples in the supplemental material.

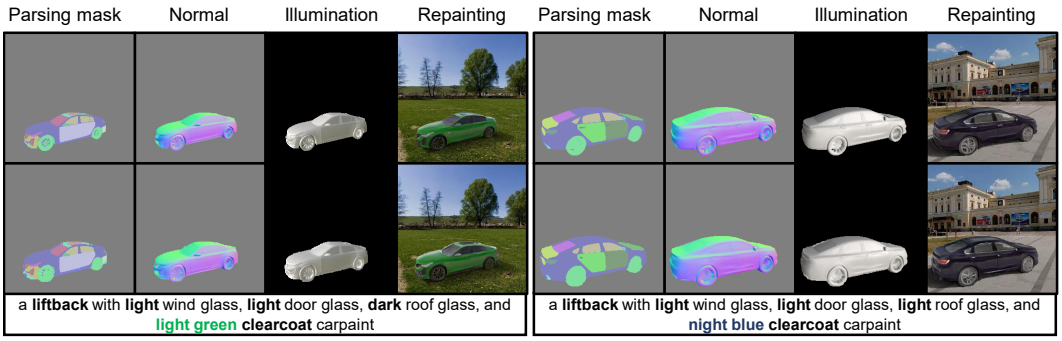


Figure 9: Qualitative results of normal maps, illumination images and repaintings using original (first row) and disturbed (second row) parsing masks as input conditions. Backgrounds are omitted here.

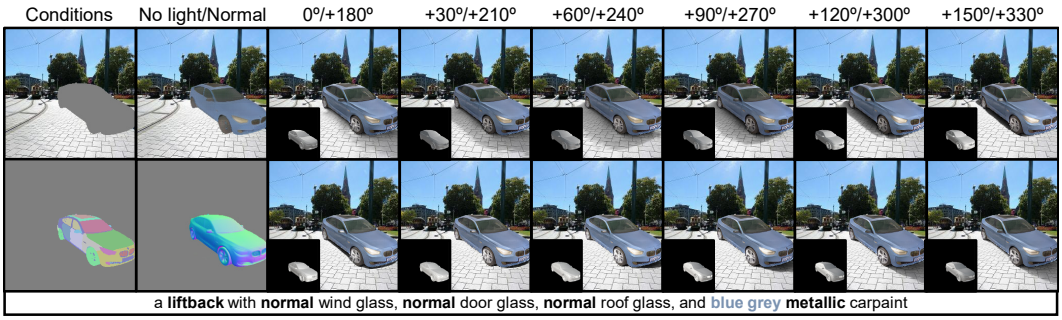


Figure 10: Qualitative results of repaintings and illumination images as the estimated lighting rotates.

7 Conclusion

In this paper, we introduce the task of LuminAIRe for the realistic generation of lighting effects. The synthetic CAR-LUMINAIRE dataset is collected for the newly proposed task. Extensive experiments and the user study confirm that our method achieves perceptually more lighting-realistic and harmonized repainting results compared with the state-of-the-art methods. The effectiveness and consistency of our illumination-aware design are shown in the robustness test.

Limitations and future works. In this paper, only the results of cars as foreground objects are shown, resulting from the inadequate feasibility of data collection. Besides, our model can not handle complex thin structures and some translucent glass materials very well, which are not well covered by our synthetic data for now. As a single-image-based method for generic outdoor scenes, our method currently ignores the non-local inter-reflections with other objects and focuses on the shadows cast directly on the ground. Therefore, datasets of richer object categories and finer details will be helpful to boost the training of learning-based methods. Combining the lighting constraints with the newly emerged latent diffusion models [55] would also be an interesting direction for our future work.

Acknowledgement. This work is supported by the National Natural Science Foundation of China under Grant No. 62136001, 62088102.

References

[1] Amazon Mechanical Turk. https://www.mturk.com.

[2] Best 3D models of Cars, Objects and more - Hum3D store. https://hum3d.com.

[3] Blender. https://www.blender.org.

[4] Squir Home. https://squir.com.

[5] G. Bae, I. Budvytis, and R. Cipolla. Estimating and exploiting the aleatoric uncertainty in surface normal estimation. In *Proc. of International Conference on Computer Vision*, 2021.

[6] Z. Bao, C. Long, G. Fu, D. Liu, Y. Li, J. Wu, and C. Xiao. Deep image-based illumination harmonization. In *Proc. of Computer Vision and Pattern Recognition*, 2022.

[7] J. T. Barron and J. Malik. Intrinsic scene properties from a single RGB-D image. In *Proc. of Computer Vision and Pattern Recognition*, 2013.

[8] A. Bhattad and D. Forsyth. Cut-and-paste object insertion by enabling deep image prior for reshading. In *Proc. of International Conference on 3D Vision*, 2022.

[9] J. F. Blinn. Models of light reflection for computer synthesized pictures. *SIGGRAPH Comput. Graph.*, 11(2):192–198, 1977.

[10] A. Brock, J. Donahue, and K. Simonyan. Large scale GAN training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

[11] Z. Chang, S. Weng, Y. Li, S. Li, and B. Shi. L-CoDer: Language-based colorization with color-object decoupling transformer. In *Proc. of European Conference on Computer Vision*, 2022.

[12] Z. Chang, S. Weng, P. Zhang, Y. Li, S. Li, and B. Shi. L-CoIns: Language-based colorization with instance awareness. In *Proc. of Computer Vision and Pattern Recognition*, 2023.

[13] X. Chen, C. Dong, J. Ji, j. Cao, and X. Li. Image manipulation detection by multi-view multi-scale supervision. In *Proc. of International Conference on Computer Vision*, 2021.

[14] W. Cong, J. Zhang, L. Niu, L. Liu, Z. Ling, W. Li, and L. Zhang. DoveNet: Deep image harmonization via domain verification. In *Proc. of Computer Vision and Pattern Recognition*, 2020.

[15] M. Contributors. MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark. https://github.com/open-mmlab/mmsegmentation, 2020.

[16] Y. Deng, F. Tang, W. Dong, C. Ma, X. Pan, L. Wang, and C. Xu. StyTr2: Image style transfer with transformers. In *Proc. of Computer Vision and Pattern Recognition*, 2022.

[17] C. Dong, X. Chen, R. Hu, J. Cao, and X. Li. MVSS-Net: Multi-view multi-scale supervised networks for image manipulation detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):3539–3553, 2023.

[18] D. Gao, G. Chen, Y. Dong, P. Peers, K. Xu, and X. Tong. Deferred neural lighting: Free-viewpoint relighting from unstructured photographs. *ACM Transactions on Graphics (Proc. of ACM SIGGRAPH Asia)*, 39(6):1–15, 2020.

[19] M.-A. Gardner, Y. Hold-Geoffroy, K. Sunkavalli, C. Gagné, and J.-F. Lalonde. Deep parametric indoor lighting estimation. In *Proc. of International Conference on Computer Vision*, 2019.

[20] M.-A. Gardner, K. Sunkavalli, E. Yumer, X. Shen, E. Gambaretto, C. Gagné, and J.-F. Lalonde. Learning to predict indoor illumination from a single image. In *Proc. of ACM SIGGRAPH Asia*, 2017.

[21] M. Garon, K. Sunkavalli, S. Hadap, N. Carr, and J.-F. Lalonde. Fast spatially-varying indoor lighting estimation. In *Proc. of Computer Vision and Pattern Recognition*, 2019.

[22] L. A. Gatys, A. S. Ecker, and M. Bethge. Image style transfer using convolutional neural networks. In *Proc. of Computer Vision and Pattern Recognition*, 2016.

[23] J. J. A. Guerreiro, M. Nakazawa, and B. Stenger. PCT-Net: Full resolution image harmonization using pixel-wise color transformations. In *Proc. of Computer Vision and Pattern Recognition*, 2023.

[24] Z. Guo, Z. Gu, B. Zheng, J. Dong, and H. Zheng. Transformer for image harmonization and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(11):12960–12977, 2023.

[25] Z. Guo, D. Guo, H. Zheng, Z. Gu, B. Zheng, and J. Dong. Image harmonization with transformer. In *Proc. of International Conference on Computer Vision*, 2021.

[26] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, G. Klambauer, and S. Hochreiter. GANs trained by a two time-scale update rule converge to a nash equilibrium. In *Proc. of Advances in Neural Information Processing Systems*, 2017.

[27] Y. Hold-Geoffroy, A. Athawale, and J.-F. Lalonde. Deep sky modeling for single image outdoor lighting estimation. In *Proc. of Computer Vision and Pattern Recognition*, 2019.

[28] Y. Hold-Geoffroy, K. Sunkavalli, S. Hadap, E. Gambaretto, and J.-F. Lalonde. Deep outdoor illumination estimation. In *Proc. of Computer Vision and Pattern Recognition*, 2017.

[29] L. Hosek and A. Wilkie. An analytic model for full spectral sky-dome radiance. *ACM Trans. on Graphics (TOG)*, 31(4), 2012.

[30] L. Hosek and A. Wilkie. Adding a solar-radiance function to the Hošek-Wilkie skylight model. *IEEE Computer Graphics and Applications*, 33(3):44–52, 2013.

[31] X. Huang and S. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proc. of International Conference on Computer Vision*, 2017.

[32] Z. Huang, N. Zhao, and J. Liao. UniColor: A unified framework for multi-modal colorization with transformer. In *Proc. of ACM SIGGRAPH Asia*, 2022.

[33] J. T. Kajiya. The rendering equation. In *Proc. of ACM SIGGRAPH*, 1986.

[34] W. Kim, B. Son, and I. Kim. ViLT: Vision-and-language transformer without convolution or region supervision. In *Proc. of International Conference on Machine Learning*, 2021.

[35] G. Kwon and J. C. Ye. CLIPstyler: Image style transfer with a single text condition. In *Proc. of Computer Vision and Pattern Recognition*, 2022.

[36] J.-F. Lalonde, L.-P. Asselin, J. Becirovski, Y. Hold-Geoffroy, M. Garon, M.-A. Gardner, and J. Zhang. The laval HDR sky database., 2016. `http://sky.hdrdb.com`.

[37] J.-F. Lalonde, A. A. Efros, and S. G. Narasimhan. Estimating the natural illumination conditions from a single outdoor image. *International Journal of Computer Vision*, 2012.

[38] J.-F. Lalonde and I. Matthews. Lighting estimation in outdoor image collections. In *Proc. of International Conference on 3D Vision*, 2014.

[39] C. LeGendre, W.-C. Ma, G. Fyffe, J. Flynn, L. Charbonnel, J. Busch, and P. Debevec. DeepLight: Learning illumination for unconstrained mobile mixed reality. In *Proc. of Computer Vision and Pattern Recognition*, 2019.

[40] Z. Li, M. Shafiei, R. Ramamoorthi, K. Sunkavalli, and M. Chandraker. Inverse rendering for complex indoor scenes: Shape, spatially-varying lighting and SVBRDF from a single image. In *Proc. of Computer Vision and Pattern Recognition*, 2020.

[41] Z. Li, L. Yu, M. Okunev, M. Chandraker, and Z. Dong. Spatiotemporally consistent hdr indoor lighting estimation. *ACM Trans. on Graphics (TOG)*, 42(3):1–15, 2023.

[42] J. Ling, H. Xue, L. Song, R. Xie, and X. Gu. Region-aware adaptive instance normalization for image harmonization. In *Proc. of Computer Vision and Pattern Recognition*, 2021.

[43] T. Miyato and M. Koyama. cGANs with projection discriminator. *arXiv preprint arXiv:1802.05637*, 2018.

[44] A. Q. Nichol, P. Dhariwal, A. Ramesh, P. Shyam, P. Mishkin, B. Mcgrew, I. Sutskever, and M. Chen. GLIDE: Towards photorealistic image generation and editing with text-guided diffusion models. In *Proc. of International Conference on Machine Learning*, 2022.

[45] A. Odena, C. Olah, and J. Shlens. Conditional image synthesis with auxiliary classifier GANs. In *Proc. of International Conference on Machine Learning*, 2017.

[46] R. Pandey, S. O. Escolano, C. Legendre, C. Haene, S. Bouaziz, C. Rhemann, P. Debevec, and S. Fanello. Total relighting: learning to relight portraits for background replacement. *ACM Transactions on Graphics (Proc. of ACM SIGGRAPH)*, 40(4):1–21, 2021.

[47] T. Park, M.-Y. Liu, T.-C. Wang, and J.-Y. Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proc. of Computer Vision and Pattern Recognition*, 2019.

[48] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros. Context encoders: Feature learning by inpainting. In *Proc. of Computer Vision and Pattern Recognition*, 2016.

[49] D. Pavllo, A. Lucchi, and T. Hofmann. Controlling style and semantics in weakly-supervised image generation. In *Proc. of European Conference on Computer Vision*, 2020.

[50] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen. Hierarchical text-conditional image generation with CLIP latents. *arXiv preprint arXiv:2204.06125*, 2022.

[51] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever. Zero-shot text-to-image generation. In *Proc. of International Conference on Machine Learning*, 2021.

[52] R. Ranftl, A. Bochkovskiy, and V. Koltun. Vision transformers for dense prediction. In *Proc. of International Conference on Computer Vision*, 2021.

[53] R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, and V. Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(3):1623–1637, 2020.

[54] Y. Ren, X. Fan, G. Li, S. Liu, and T. H. Li. Neural texture extraction and distribution for controllable person image synthesis. In *Proc. of Computer Vision and Pattern Recognition*, 2022.

[55] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models. In *Proc. of Computer Vision and Pattern Recognition*, 2022.

[56] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. L. Denton, K. Ghasemipour, R. Gontijo Lopes, B. Karagol Ayan, T. Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Proc. of Advances in Neural Information Processing Systems*, 2022.

[57] S. Song and T. Funkhouser. Neural Illumination: Lighting prediction for indoor environments. In *Proc. of Computer Vision and Pattern Recognition*, 2019.

[58] J. Sun, S. Weng, Z. Chang, S. Li, and B. Shi. UniCoRN: A unified conditional image repainting network. In *Proc. of Computer Vision and Pattern Recognition*, 2022.

[59] K. Sunkavalli, M. K. Johnson, W. Matusik, and H. Pfister. Multi-scale image harmonization. *ACM Transactions on Graphics (Proc. of ACM SIGGRAPH)*, 29(4):1–10, 2010.

[60] J. Tang, Y. Zhu, H. Wang, J.-H. Chan, S. Li, and B. Shi. Estimating spatially-varying lighting in urban scenes with disentangled representation. In *Proc. of European Conference on Computer Vision*, 2022.

[61] S. Tripathi, S. Chandra, A. Agrawal, A. Tyagi, J. M. Rehg, and V. Chari. Learning to generate synthetic data via compositing. In *Proc. of Computer Vision and Pattern Recognition*, 2019.

[62] Y.-H. Tsai, X. Shen, Z. Lin, K. Sunkavalli, X. Lu, and M.-H. Yang. Deep image harmonization. In *Proc. of Computer Vision and Pattern Recognition*, 2017.

[63] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[64] Z. Wang, W. Chen, D. Acuna, J. Kautz, and S. Fidler. Neural light field estimation for street scenes with differentiable virtual object insertion. In *Proc. of European Conference on Computer Vision*, 2022.

[65] Z. Wang, X. Qi, K. Yuan, and M. Sun. Self-supervised correlation mining network for person image generation. In *Proc. of Computer Vision and Pattern Recognition*, 2022.

[66] S. Weng, W. Li, D. Li, H. Jin, and B. Shi. Conditional image repainting via semantic bridge and piecewise value function. In *Proc. of European Conference on Computer Vision*, 2020.

[67] S. Weng, W. Li, D. Li, H. Jin, and B. Shi. MISC: Multi-condition injection and spatially-adaptive compositing for conditional person image synthesis. In *Proc. of Computer Vision and Pattern Recognition*, 2020.

[68] S. Weng, J. Sun, Y. Li, S. Li, and B. Shi. CT^2: Colorization transformer via color tokens. In *Proc. of European Conference on Computer Vision*, 2022.

[69] S. Weng, H. Wu, Z. Chang, J. Tang, S. Li, and B. Shi. L-code: Language-based colorization using color-object decoupled conditions. In *Proc. of the AAAI Conference on Artificial Intelligence*, 2022.

[70] S. Weng, P. Zhang, Z. Chang, X. Wang, S. Li, and B. Shi. Affective image filter: Reflecting emotions from text to images. In *Proc. of International Conference on Computer Vision*, 2023.

[71] J. Xiao, K. A. Ehinger, A. Oliva, and A. Torralba. Recognizing scene viewpoint using panoramic place representation. In *Proc. of Computer Vision and Pattern Recognition*, 2012.

[72] W. Xiong, J. Yu, Z. Lin, J. Yang, X. Lu, C. Barnes, and J. Luo. Foreground-aware image inpainting. In *Proc. of Computer Vision and Pattern Recognition*, 2019.

[73] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang. Free-form image inpainting with gated convolution. In *Proc. of International Conference on Computer Vision*, 2019.

[74] P. Yu, J. Guo, F. Huang, C. Zhou, H. Che, X. Ling, and Y. Guo. Hierarchical disentangled representation learning for outdoor illumination estimation and editing. In *Proc. of International Conference on Computer Vision*, 2021.

[75] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena. Self-attention generative adversarial networks. In *Proc. of International Conference on Machine Learning*, 2019.

[76] J. Zhang and J.-F. Lalonde. Learning high dynamic range from outdoor panoramas. In *Proc. of International Conference on Computer Vision*, 2017.

[77] J. Zhang, K. Sunkavalli, Y. Hold-Geoffroy, S. Hadap, J. Eisenmann, and J.-F. Lalonde. All-weather deep outdoor lighting estimation. In *Proc. of Computer Vision and Pattern Recognition*, 2019.

[78] R. Zhang, P. Isola, and A. A. Efros. Colorful image colorization. In *Proc. of European Conference on Computer Vision*, 2016.

[79] Y. Zhang, N. Huang, F. Tang, H. Huang, C. Ma, W. Dong, and C. Xu. Inversion-based creativity transfer with diffusion models. In *Proc. of Computer Vision and Pattern Recognition*, 2023.

[80] P. Zhu, R. Abdal, Y. Qin, and P. Wonka. SEAN: Image synthesis with semantic region-adaptive normalization. In *Proc. of Computer Vision and Pattern Recognition*, 2020.

[81] Y. Zhu, Y. Zhang, S. Li, and B. Shi. Spatially-varying outdoor lighting estimation from intrinsics. In *Proc. of Computer Vision and Pattern Recognition*, 2021.

LuminAIRe: Illumination-Aware Conditional Image Repainting for Lighting-Realistic Generation (Supplemental Material)

Jiajun Tang[1,2] **Haofeng Zhong**[1,2] **Shuchen Weng**[1,2] **Boxin Shi**[1,2*]
[1]National Key Laboratory for Multimedia Information Processing
[2]National Engineering Research Center of Visual Technology
School of Computer Science, Peking University
{jiajun.tang, hfzhong, shuchenweng, shiboxin}@pku.edu.cn

In this supplementary material, we provide more information about our data collection, implementation details, and network architectures. We also show additional results on our CAR-LUMINAIRE dataset and in-the-wild data.

8 Appendix

8.1 Details on the CAR-LUMINAIRE Dataset

When calculating the object's local coordinate system O_o, we assume that $X_c \parallel X_o$, that is to say, we assume that the ground is approximately horizontally level, which is satisfied at most times.

We divide the car into 35 classes of parts, of which the hierarchy and color coding are shown in Fig. 11.

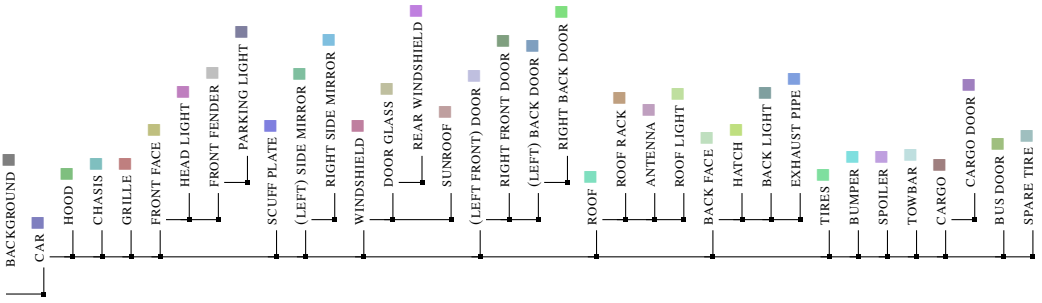


Figure 11: The hierarchy and color coding of the part labels used in the CAR-LUMINAIRE dataset.

The attribute of car models is also manually annotated when labeling the parts of models. We mark $N_c = 6$ major part-related attributes (car type, wind glass darkness, door glass darkness, roof glass darkness, car paint color, and car paint type) with $N_v = 81$ available choices. The N_g is set as 35 following our hierarchical labeling. The relationship between parts and corresponding attributes is represented as an association matrix [14] $A \in \mathbb{L}^{N_c \times N_g}$.

We use a randomly chosen camera pitch in $[-15°, 15°]$ and FoV in $[25°, 66°]$ for the background image cropping. The same FoV is used in image rendering for view consistency. For each combination of the background image and geometry condition, we render one image of the original car model and two images of variants with randomly chosen enhanced car paint materials. When splitting the

*Corresponding author.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

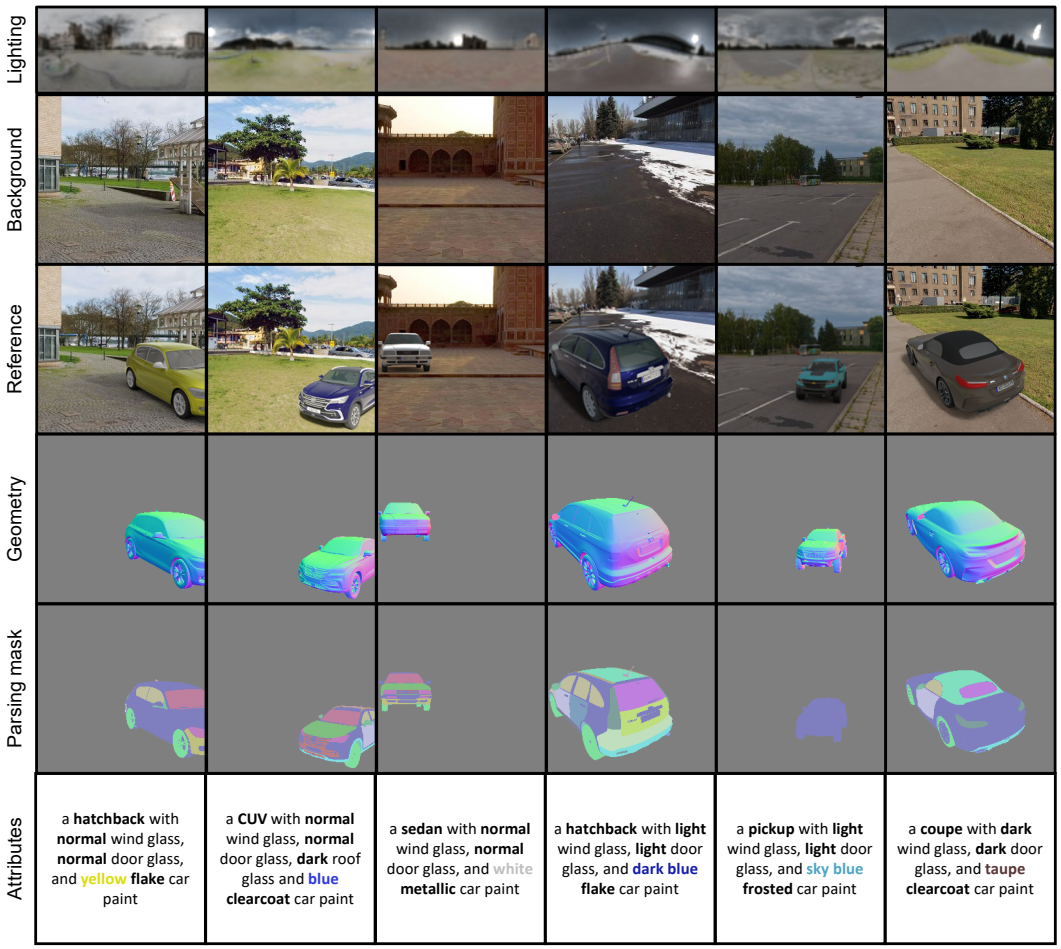


Figure 12: Data examples of our CAR-LUMINAIRE dataset.

dataset, we assure the images of each combination are only shown in one set. We only use images with the pixel ratio of the foreground between 10% and 50%, leaving 58,521 images (41,058 for the training set, 12,141 for the testing test, and 5,322 for the validation set). Each set of data contains a background image (256×256), a lighting annotation (128×64 envmap), a rendered reference image (256×256), a geometry annotation (256×256), a parsing mask (256×256), and an attribute annotation. Here we give more data examples of our CAR-LUMINAIRE dataset in Fig. 12.

8.2 Details on the Parametric Lighting Representation

As stated in Sec. 5.1 of the main paper, we use a combination of 2-nd order SH lighting and directional lighting as our parametric lighting representation, where the low-frequency SH lighting is designed to fit the ambient lighting in the environment and the high-frequency directional lighting is used to describe the sunlight. Since the original lighting annotations in our CAR-LUMINAIRE dataset are envmaps, therefore, conversions have to be made to get the training labels in our parametric lighting representation.

The part of directional light is represented as $L_{\text{dir}} = \{z_{\text{vis}}, z_{\text{int}}, z_{\text{ang}}, c_{\text{sun}}, l_{\text{sun}}\}$ in our representation. For each envmap, if the maximum grey-sacle intensity z_{int} is larger than a threshold $\delta_{\text{sun}} = 100$, the sun visibility z_{vis} is set as 1 otherwise 0 (and other parameters treated as invalid). Then we use a manually set ratio $r_{\text{sun}} = 0.1$ and only keep pixels with grey-scale intensity larger than $r_{\text{sun}} z_{\text{int}}$. We calculate the intensity-weighted mass center of the connected area $\mathcal{A}_{\text{sun}}$ containing the pixel of maximum intensity, as the direction of the sun $l_{\text{sun}} \in \mathbb{R}^2$ in the spherical coordinates. The diameter

of $\mathcal{A}_{\text{sun}}$ in pixels is used as z_{ang} and the mean RGB values divided by the mean intensity values over $\mathcal{A}_{\text{sun}}$ is used as the RGB weights $c_{\text{sun}} \in \mathbb{R}^3$.

The ideal directional light has no corresponding solid angle $\boldsymbol{\omega}$ and thus can not be directly used in Eq. (10) of the main paper, we approximately "assign" a small solid angle $\boldsymbol{\omega}_{\text{dir}}$ corresponding to a pixel (minimum visible unit) in the envmap. To calculate the equivalent intensity i_{dir} corresponding to the pixel in the envmap, we assume the intensity from the sun center to the surroundings approximately fits the Gaussian distribution $f_{\text{G}}(x) = \alpha_{\text{G}} \exp\left(-\frac{x^2}{2\sigma_{\text{G}}^2}\right)$, and therefore we have:

$$f_{\text{G}}(0.5) = z_{\text{int}}, \quad f_{\text{G}}(z_{\text{ang}}/2) = r_{\text{sun}} z_{\text{int}}, \tag{11}$$

and we can solve $\alpha_{\text{G}} = z_{\text{int}} r_{\text{sun}}^{-\frac{1}{z_{\text{ang}}^2 - 1}}$ and $\sigma_{\text{G}} = \sqrt{\frac{z_{\text{ang}}^2 - 1}{8 \ln(1/r_{\text{sun}})}}$ from Eq. (11) and therefore we have:

$$i_{\text{dir}} = \int_0^{z_{\text{ang}}/2} f_{\text{G}}(r) \mathrm{d}\boldsymbol{\omega}(r) / \boldsymbol{\omega}_{\text{dir}} \approx \int_0^{z_{\text{ang}}/2} f_{\text{G}}(r) 2\pi r \mathrm{d}r / \boldsymbol{\omega}_{\text{dir}}. \tag{12}$$

We use 2-nd order SH coefficients $\sigma_{\text{SH}} = \{\sigma_{\text{SH}}^{\text{R}}, \sigma_{\text{SH}}^{\text{G}}, \sigma_{\text{SH}}^{\text{B}}\}$ to represent low-frequency light in each RGB color channel, where each $\sigma_{\text{SH}}^* = \{\sigma_{0,0}^*, \sigma_{1,\text{-}1}^*, \sigma_{1,0}^*, ..., \sigma_{2,2}^*\}$ are the corresponding coefficients for the 2-nd order spherical harmonics basis $\{Y_{0,0}, Y_{1,\text{-}1}, Y_{1,0}, ..., Y_{2,2}\}$. Due to the orthogonality of the spherical harmonics basis, the coefficients for low-frequency lighting $i_{\text{SH}}(\theta, \varphi)$ are computed as:

$$\sigma_{l,m}^* = \int_0^{2\pi} \int_0^{\pi} i_{\text{SH}}^*(\theta, \varphi) Y_{l,m}(\theta, \varphi) \sin\theta \mathrm{d}\theta \mathrm{d}\varphi, \tag{13}$$

where we use the envmap annotation (clipped into $[0, r_{\text{sun}} z_{\text{int}}]$ if z_{vis} is 1) as $i_{\text{SH}}(\theta, \varphi)$. The reconstructed $\hat{i}_{\text{SH}}(\theta, \varphi)$ is simply the weighted sum of the spherical harmonics basis:

$$\hat{i}_{\text{SH}}^*(\theta, \varphi) = \sum_{i=0}^{l} \sum_{j=-l}^{l} \sigma_{i,j}^* Y_{i,j}(\theta, \varphi). \tag{14}$$

8.3 Details on the Illumination Image Rendering

The rendering of the illumination (candidate) images is conducted by applying Eq. (10) of the main paper pixel-wisely. Since the actual camera FoV is unknown, here we assume the camera viewing directions of all pixels are the same $\boldsymbol{v} = (0, 0, -1)$ (orthogonal camera model), which is shown in Fig. 5 of the main paper to be a reasonable approximation for lighting-realistic generation tasks.

Since we use the normal map as the representation of geometry, which is not a complete 3D shape model (such as meshes, or signed distance functions), we only calculate single-bounce light effects, ignoring complex light transport effects such as self-cast shadow or inter-reflections. This is a trade-off between using the costly (and maybe more unreliable) single-view full 3D reconstruction or ignoring inconspicuous indirect light bounces.

The integration over the hemisphere $\Omega_{\boldsymbol{n}}$ in Eq. (10) can be done discretely on an envmap. Therefore, the most intuitive way for the calculation is converting our parametric lighting representations back to envmaps before applying Eq. (10). However, a more efficient computation can be done utilizing the properties of our parametric representation, where we use $\rho \in \{1, 2, 4, 8, 16, 32, 64, 128\}$ for $\{I_{\text{c}}\}$.

Specifically, we divide I as the sum of two parts I_{SH} and I_{dir} corresponding to our representation. Then each pixel p of $I_{\text{diff,dir}}$ and $I_{\text{spec,dir}}^{\rho}$ can be calculated without integration as $i_{\text{dir}} c_{\text{sun}} (\boldsymbol{n}_p \cdot \boldsymbol{l}_{\text{dir}}) \boldsymbol{\omega}_{\text{dir}}$ and $i_{\text{dir}} c_{\text{sun}} (\boldsymbol{n}_p \cdot \boldsymbol{h}_{\text{dir}})^{\rho} \boldsymbol{\omega}_{\text{dir}}$, where $\boldsymbol{l}_{\text{dir}}$ is the Cartesian coordinate representation of l_{sun} and $\boldsymbol{h}_{\text{dir}} = \boldsymbol{l}_{\text{dir}} - \boldsymbol{v} / ||\boldsymbol{l}_{\text{dir}} - \boldsymbol{v}||$ is the half vector introduced in Sec. 5.2 of the main paper. The negative dot product is clipped to 0 to avoid underflow. Besides, we also clip the minimums of I_{SH} to 0.

For $I_{\text{diff,SH}}$, each pixel p is fast calculated by using $Y_{l,m}(\theta, \varphi)$ to describe the distribution of $\boldsymbol{l}$ [13]:

$$\begin{aligned} I_{\text{diff,SH}}^p = [&c_1 \sigma_{2,2} ({n_x^p}^2 - {n_y^p}^2) + c_3 \sigma_{2,0} {n_z^p}^2 + c_4 \sigma_{0,0} - c_5 \sigma_{2,0} \\ &+ 2c_1 (\sigma_{2,\text{-}2} n_x^p n_y^p + \sigma_{2,1} n_x^p n_z^p + \sigma_{2,\text{-}1} n_y^p n_z^p) + 2c_2 (\sigma_{1,1} n_x^p + \sigma_{1,\text{-}1} n_y^p + \sigma_{1,0} n_z^p)] / \pi, \end{aligned} \tag{15}$$

with weights $c_1 = 0.429043$, $c_2 = 0.511664$, $c_3 = 0.743125$, $c_4 = 0.886227$, and $c_5 = 0.247708$.

For $I^{\rho}_{\text{spec,SH}}$, we have $\theta_{\boldsymbol{l}} = 2\theta_{\boldsymbol{h}}$ and $\varphi_{\boldsymbol{l}} = \varphi_{\boldsymbol{h}}$. Similarly, $\hat{Y}_{l,m}(\theta, \varphi) = Y_{l,m}(2\theta, \varphi)$ is used to describe the distribution of $\boldsymbol{h}$ [22], which gives the fast approximation of pixel p with Blinn-Phong model [2]:

$$
\begin{aligned}
I^{\rho,p}_{\text{spec,SH}} \approx \{&\sigma_{0,0}(c_4)^{\rho} + \sigma_{1,-1}(4c_2 n_y^p n_z^p)^{\rho} + \sigma_{1,0}[2c_2(2{n_z^p}^2 - 1)]^{\rho} + \sigma_{1,1}(4c_2 n_x^p n_z^p)^{\rho} \\
&+ \sigma_{2,-2}(8c_1 n_x^p n_y^p {n_z^p}^2)^{\rho} + \sigma_{2,-1}[2c_1(4n_y^p {n_z^p}^3 - 2n_y^p n_z^p)]^{\rho} + \sigma_{2,0}[c_5(12{n_z^p}^4 - 12{n_z^p}^2 + 2)]^{\rho} \\
&+ \sigma_{2,1}[2c_1(4n_x^p {n_z^p}^3 - 2n_x^p n_z^p)]^{\rho} + \sigma_{2,2}[c_1(4{n_x^p}^2 {n_z^p}^2 - 4{n_y^p}^2 {n_z^p}^2)]^{\rho}\}(\rho + 4)/8\pi.
\end{aligned} \tag{16}
$$

8.4 Details on the User Study

We randomly sample 200 sets of results of compared methods and ask volunteers to choose one in each set that best matches the following description: *(i)* "The repainted region which seems most *realistic*"; *(ii)* "the whole repainted image which seems most *harmonized* in lighting"; *(iii)* "the whole repainted image which seems most *realistic overall*".

The volunteers are shown with the masked repainted foreground images, *i.e.*, without the background context when asked about the *realistic* question. Then the full repainted images are shown and the *harmonized* question is asked on the same set of results, where we use our repainted background region for all results to prevent our method to be identified or guessed out by only noticing the difference in the background. The original results of compared methods are shown to the volunteers when asking about the *realistic overall* question. We first ask the *realistic* question, then the *harmonized* question, and at last the *realistic overall* question. We have reported the results of the first two questions in Tab. 1 of the main paper while the results for the *realistic overall* question are: **Ours**: **77.32%**, **Pavllo *et al.*** [12]: 14.50%, **UniCoRN** [6]: 7.03%, **Weng *et al.*** [18]: 0.92%, **MISC** [19]: 0.23%.

The order of sets and images in each set is randomized, and we deliberately duplicate 5 sets of the samples as the quality control questions to judge whether the volunteers have paid attention when finishing the questionnaires. Questionnaires that failed in the quality control questions are discarded.

8.5 Training Details

Experimental settings. Our pipeline is implemented in PyTorch [11] and trained step-wise. We first train our *NetL* on the held-out background images with a batch size of 64 and an initial learning rate of 1×10^{-4} (which halves every 20 epochs) for 60 epochs, where we estimate the sun position l_{sun} in the form of an 8×32 classification task and we apply log-compressed tone mapping [8] $T = {\log(1+16H)}/{\log(1+16)}$ for the HDR sun intensity z_{int}. Our *NetS* and *NetB* are separately trained on our CAR-LUMINAIRE dataset with a batch size of 32 and a fixed learning rate of 2×10^{-4} for 60 epochs. Then we run our full pipeline optimization (one discriminator step after each generator step) with fixed *NetL*, *NetS*, and *NetB* to learn the network parameters of *NetF*, with a batch size of 24 and a fixed learning rate of 2×10^{-4} for 30 epochs. During the training of *NetS* and *NetF*, we use the hierarchical labeling enhancement at the probability of 0.5, where each part label has a probability of 0.5 to be coarsened to its upper-level label. Before illumination injection, the illumination image I is clipped by an empirically set threshold $\delta_I = 2.0$ to simulate the over-exposure of highlights in LDR images and avoid extremely high inputs to network layers. For cross-modality conditional consistency constraints, we pretrain the image encoder Enc^{i} (omitted in the main paper) and the attribute encoder Enc^{c} (Fig. 3) on our CAR-LUMINAIRE dataset following previous work [21].

The baseline methods are trained on our CAR-LUMINAIRE dataset with the same batch size of 24 as our *NetF* for 30 epochs using their default settings in their released code. We use Adam optimizer [9] in all of our experiments, and all experiments are conducted on 4 NVIDIA Tesla V100 graphic cards.

Training losses. Our full pipeline is trained with the following losses:

$$
\mathcal{L} = \mathcal{L}_{\text{L}} + \mathcal{L}_{\text{S}} + \mathcal{L}_{\text{B}} + \mathcal{L}_{\text{F}}, \tag{17}
$$

where $\mathcal{L}_{\text{L}}$, $\mathcal{L}_{\text{S}}$, $\mathcal{L}_{\text{B}}$, and $\mathcal{L}_{\text{F}}$ are the loss terms for our *NetL*, *NetS*, *NetB*, and *NetF*, respectively.

For our *NetL*, $\mathcal{L}_{\text{L}}$ consists of two parts $\mathcal{L}_{\text{L}} = \mathcal{L}_{\text{SH}} + \mathcal{L}_{\text{dir}}$ corresponding to our lighting modeling:

$$
\mathcal{L}_{\text{SH}} = \mathcal{L}_{\text{coeff}} + \mathcal{L}_{\text{pano}}, \quad \mathcal{L}_{\text{dir}} = \mathcal{L}_{\text{vis}} + \mathcal{L}_{\text{pos}} + \mathcal{L}_{\text{param}}, \tag{18}
$$

where $\mathcal{L}_{\text{coeff}}$ is an L_2 loss for σ_{SH} with $\sigma^{*}_{l,m}$ from Eq. (13), $\mathcal{L}_{\text{pano}}$ is an L_1 loss for envmaps reconstructed by SH coefficients $\hat{i}^{*}_{\text{SH}}(\theta, \varphi)$ from Eq. (14) with $i_{\text{SH}}(\theta, \varphi)$, $\mathcal{L}_{\text{vis}}$ is a binary cross-entropy loss

for z_{vis}, $\mathcal{L}_{\text{pos}}$ is a cross-entropy loss for the 8×32 classification results of l_{sun}, and $\mathcal{L}_{\text{param}}$ are L_2 losses for the remaining parameters (log-compressed z_{int}, z_{ang}, and c_{sun}). For images with the sun not visible ($z_{\text{vis}} = 0$) in the lighting annotations, we set $\mathcal{L}_{\text{pos}} = \mathcal{L}_{\text{param}} = 0$.

For our *NetS*, $\mathcal{L}_{\text{S}}$ is defined as:

$$\mathcal{L}_{\text{S}} = \mathcal{L}_{\text{sp}} + \mathcal{L}_{\text{s-smooth}}, \tag{19}$$

where $\mathcal{L}_{\text{sp}}$ is an L_2 loss for G, and $\mathcal{L}_{\text{s-smooth}} = \sum [(\nabla_i G)^2 + (\nabla_j G)^2]$ is the smoothness loss for G.

For our *NetB*, $\mathcal{L}_{\text{B}}$ is defined as:

$$\mathcal{L}_{\text{B}} = \mathcal{L}_{\text{bg}} + \mathcal{L}_{\text{b-smooth}} + \mathcal{L}_{\text{b-dis}}, \tag{20}$$

where $\mathcal{L}_{\text{bg}}$ is an L_1 loss for y^{b}, $\mathcal{L}_{\text{b-smooth}} = \sum [(\nabla_i (y^{\text{b}}/x^{\text{b}}))^2 + (\nabla_j (y^{\text{b}}/x^{\text{b}}))^2]$ is the smoothness loss for y^{b}, and $\mathcal{L}_{\text{b-dis}}$ is the discriminator loss for y^{b} with the background regions of reference images.

For our *NetF*, $\mathcal{L}_{\text{F}}$ is defined following UniCoRN [14] as:

$$\mathcal{L}_{\text{F}} = \mathcal{L}_{\text{fg}} + \mathcal{L}_{\text{r}} + \mathcal{L}_{\text{bc}} + \mathcal{L}_{\text{fm}} + \mathcal{L}_{\text{per}} + \mathcal{L}_{\text{cm}}, \tag{21}$$

where $\mathcal{L}_{\text{fg}}$ and $\mathcal{L}_{\text{r}}$ are discriminator losses judging whether y^{f} is real and whether y^{r} is composited, $\mathcal{L}_{\text{bc}}$ is an L_1 loss enforcing $(1 - m) \odot y^{\text{f}}$ close to x^{b}, $\mathcal{L}_{\text{fm}}$ and $\mathcal{L}_{\text{per}}$ are the feature matching loss [17] and the perceptual loss [3] for y^{r}, and $\mathcal{L}_{\text{cm}}$ is the cross-modality conditional matching loss [21].

8.6 Implementations of Baseline Methods

We use the released code of UniCoRN [14], MISC [19], Weng *et al*. [18], and Pavllo *et al*. [12] as the implementations of our baseline method. As mentioned in the main paper, modifications have been made to the released code of Weng *et al*. [18] and Pavllo *et al*. [12] for taking attributes as the input color condition x^{c}. Besides, for Pavllo *et al*. [12], since their generated background is not conditioned on either the original background or other conditions and thus is not controllable, we discard their generated background and replace it with the input background image to fit the formulation of CIR.

8.7 Additional Results

Lighting and shape estimations. Our LuminAIRe pipeline consists of lighting and shape estimations which will inevitably introduce errors. As stated in Sec. 5.1 of the main paper, the low-frequency part (SH lighting) and high-frequency part (directional light) of the lighting are separately estimated. Here we report the lighting estimation errors from masked background images (foreground regions masked by zeros) in each part: *(i)* directional (sun) light: mean angular error (MAE): 28.37°, mean azimuth error: 3.84°, and mean elevation error: 27.74°; *(ii)* SH lighting: mean absolute error of SH coefficients: 0.0488, mean square error of SH coefficients: 0.0054, mean absolute error of envmaps reconstructed by SH coefficients: 0.0435, and mean square error of envmaps reconstructed by SH coefficients: 0.0043. Similarly, we report the estimation errors on normalized normal maps in the shape estimation: mean angular error (MAE): 9.83°, mean absolute error: 0.0167, and mean square error: 0.0039. These errors would prevent us from recovering the exact lighting effects, however, are tolerable for the demand of lighting-realistic repaintings.

More comparisons and ablation variants. We conduct a breakdown evaluation on how our method and compared methods work on foreground regions (noted as "fg.") and how repainted background regions (noted as "bg.") by our method contribute to the realistic perception. We also compare our method with more ablation variants (**Ours-A** and **Ours-AI**) for the completeness of the ablation study. Despite that we can not compare our method with image harmonization methods in an exact fair setting, as an intuitive reference, we choose two of the latest methods (**DHT+** [5] and **PCT-Net** [4]) and use the repainting results from **Ours-AI** as their inputs. The quantitative results are shown in Tab. 2 and the qualitative comparisons are shown in Fig. 13, where the harmonized images show better integrity than input as M-score indicates, however, do not show better lighting effects and may have severe color-shifting issues as R-prcn and SSIM scores indicate.

More results on our CAR-LUMINAIRE **dataset.** More qualitative results on our CAR-LUMINAIRE dataset are shown in Fig. 14 and Fig. 15. Our LuminAIRe generally performs better qualitatively than baseline methods in generating realistic, harmonized, and consistent lighting effects.

In-the-wild performance. To test the generalization ability of the compared methods, we show qualitative results of in-the-wild data which are collected from the Waymo dataset [15] and the

Table 2: Additional quantitative evaluation results. Separated evaluations of foreground and background regions are shown. Qualitative results of additional ablation variants and image harmonization baselines are also shown. ↑ (↓) means higher (lower) is better. "fg." stands for "foreground" and "bg." stands for "background". Please note that "original bg." corresponds to the background regions of all compared baseline methods since they leave the background untouched.

Method	FID ↓	R-prcn ↑	M-score ↓	SSIM ↑
MISC fg.	76.26	—	—	0.8228
Weng *et al*. fg.	44.14	—	—	0.8306
Pavllo *et al*. fg.	6.14	—	—	0.8671
UniCoRN fg.	9.53	—	—	0.8541
Ours fg.	**4.30**	—	—	**0.8689**
Original bg.	21.43	—	—	0.8309
Ours bg.	**4.94**	—	—	**0.8494**
Ours-A	**5.04**	**74.29%**	13.76	**0.7167**
Ours-AI	5.72	**74.73%**	15.36	0.7106
DHT+ [5]	5.94	67.34%	9.02	0.7057
PCT-Net [4]	5.31	69.59%	**7.85**	0.7035

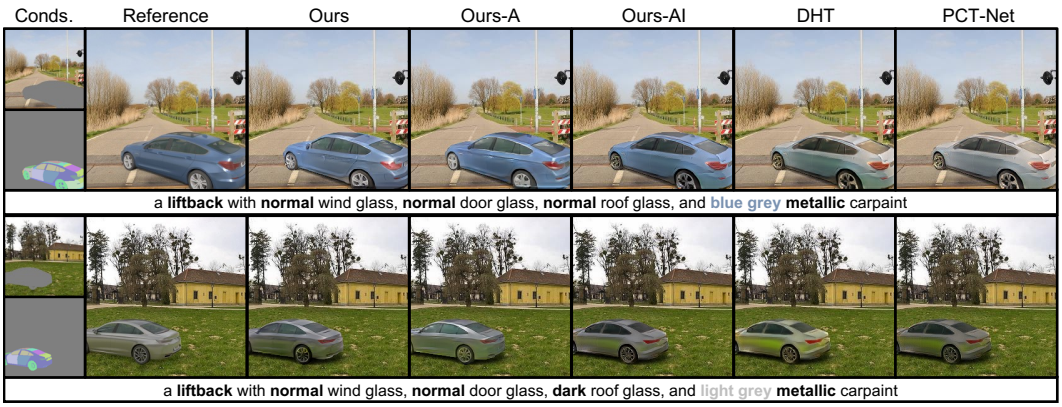


Figure 13: Qualitative results of additional ablation variants and image harmonization baselines.

UASOL dataset [1] in Fig. 16. Although their data distribution is far different from our synthetic data, our LuminAIRe still gives reasonable lighting-realistic results compared with baseline methods.

Failure cases. Here we analyze examples of failure cases in Fig. 17. When the repainted region is across the boundary of the shadows (the first row), the global lighting assumption may lead to unrealistic lighting effects. A too-coarse parsing mask (the second row) would raise serious geometry ambiguity and renders a failed repainting. The lighting effects would become less realistic if the accumulated errors in lighting and shape estimations were too large (the third row). The occasionally badly repainted background (the fourth row) would also do harm to the lighting-realistic perception.

8.8 Detailed Network Architectures

We show the detailed network architectures of the *NetL*, *NetS*, *NetB*, and *NetF* from Fig. 19 to Fig. 21, with the structures and default settings of common blocks shown in Fig. 18.

The network architectures of the image encoder Enc^{i} and the attribute encoder Enc^{c} for measuring cross-modality conditional consistency remain the same with the HCMSM proposed in UniCoRN [14]. We adopt the network backbone of their F^{G} for our *NetF*, where we inject the illumination images I as the illumination condition x^{i} in 2D image space from the resolutions of 32×32 to 256×256. Specifically, we replace the batch normalization layers with instance normalization layers in FABN module and ignore the texture condition x^{p} when injecting the illumination image I.

Figure 14: More qualitative comparisons on our CAR-LUMINAIRE dataset.

Figure 15: More qualitative comparisons on our CAR-LUMINAIRE dataset.

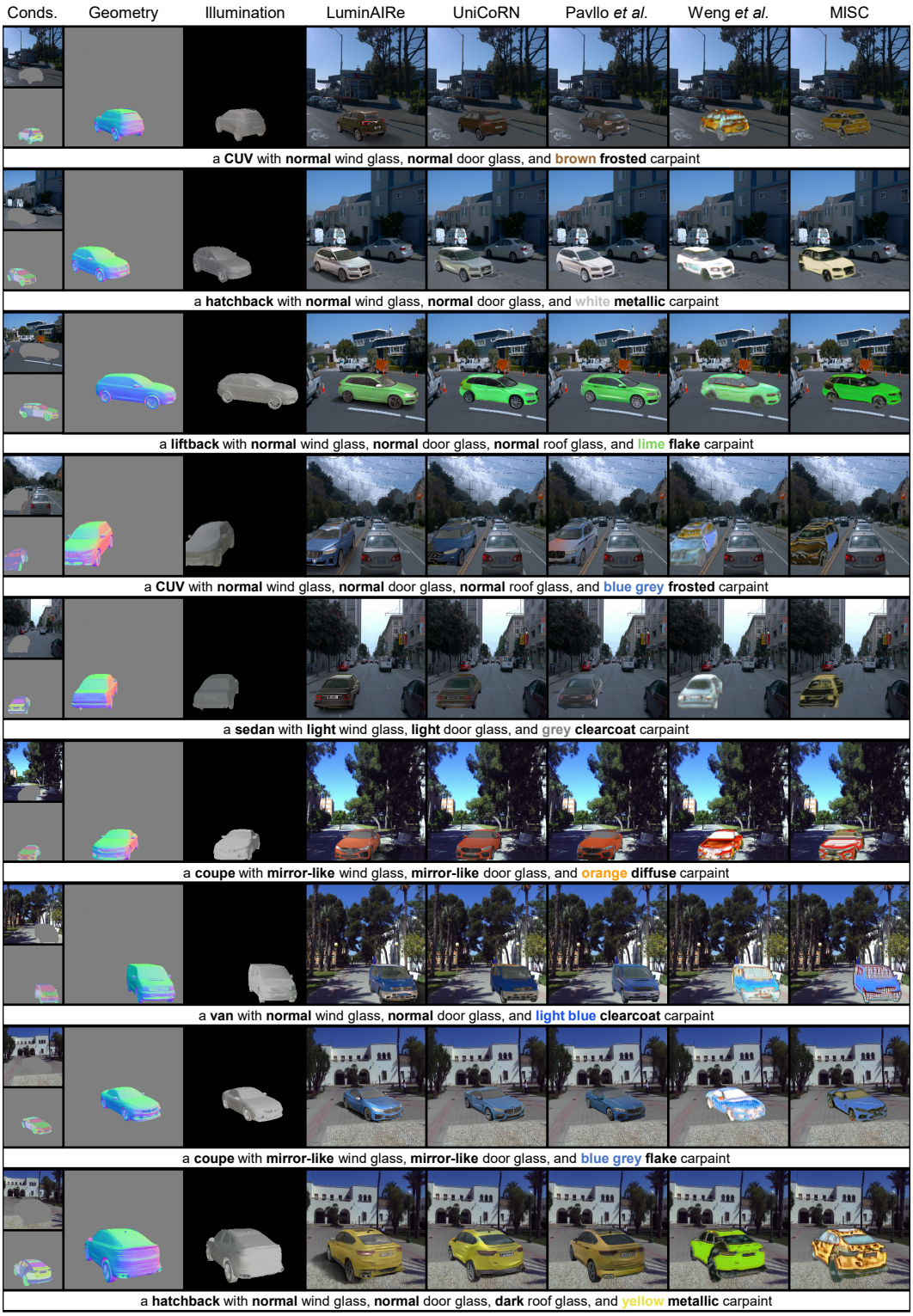


Figure 16: More qualitative comparisons on in-the-wild data.

Figure 17: Failure cases on our CAR-LUMINAIRE dataset.

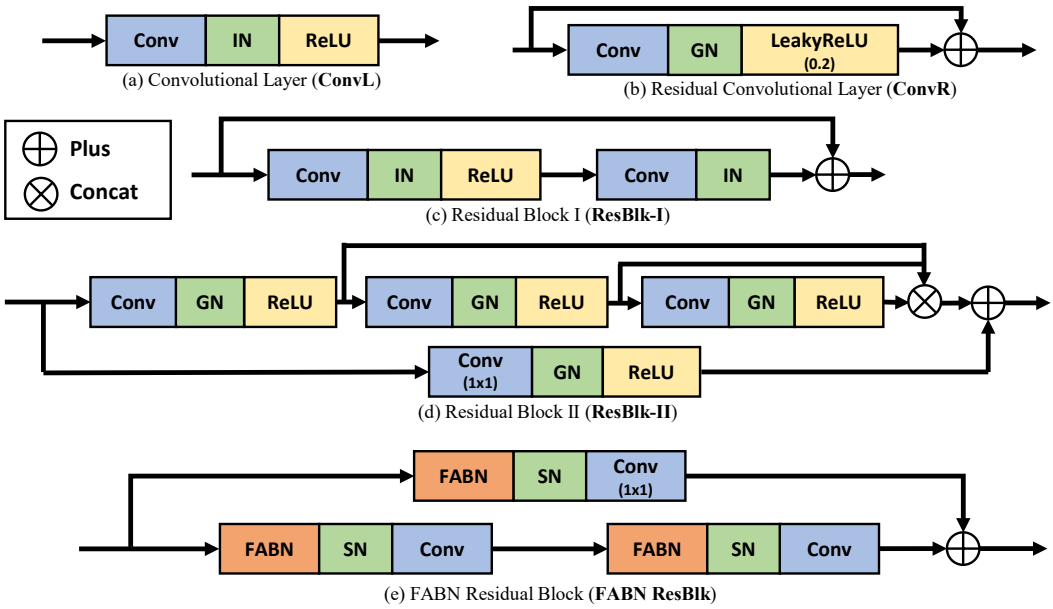
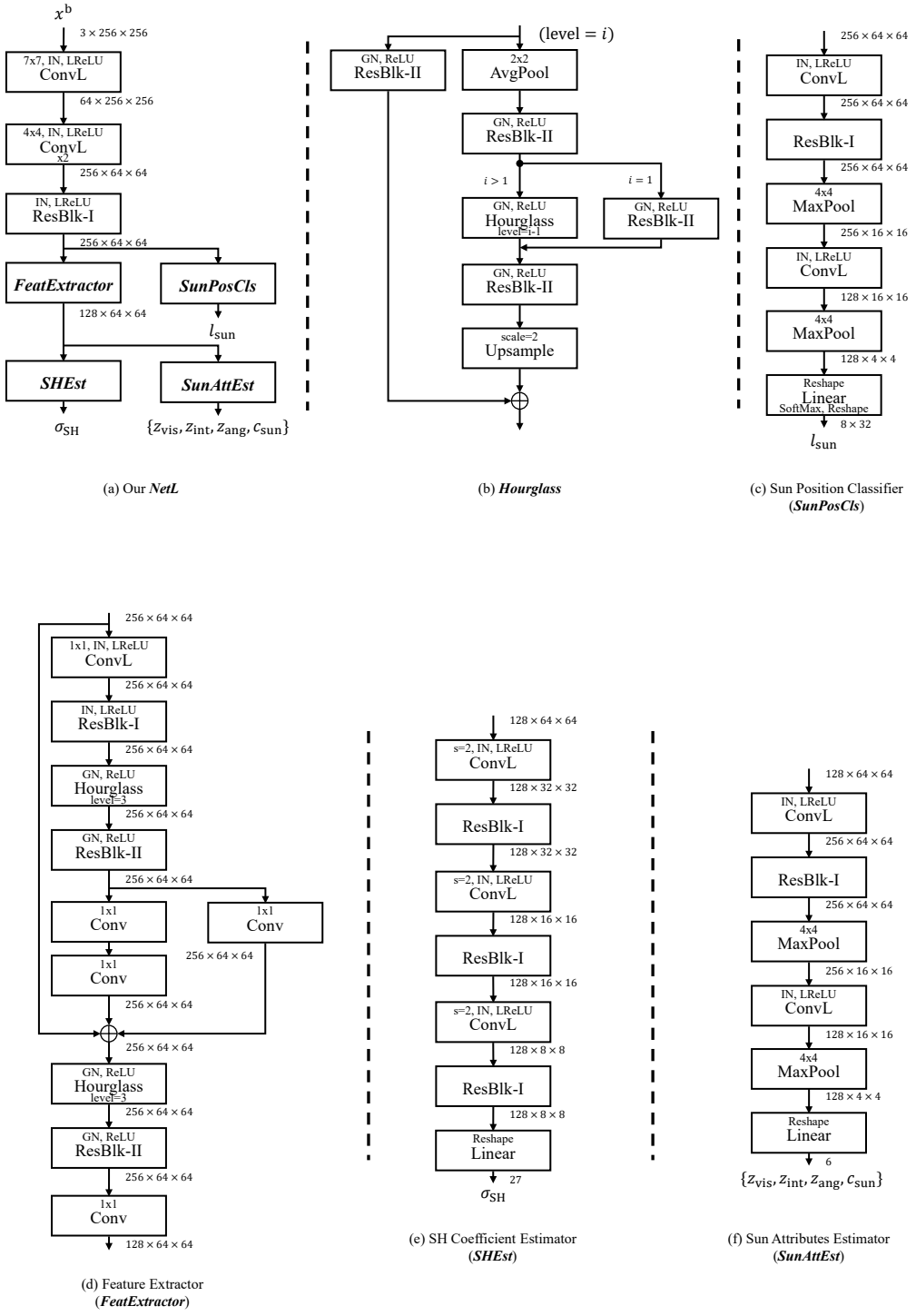


Figure 18: Common blocks used in the network architectures. Notations: BN = Batch Normalization [7], IN = Instance Normalization [16], GN = Group Normalization [20], SN = Spectral Normalization [10], FABN = Feature Adaptive Batch Normalization [14].



(a) Our ***NetL***

(b) ***Hourglass***

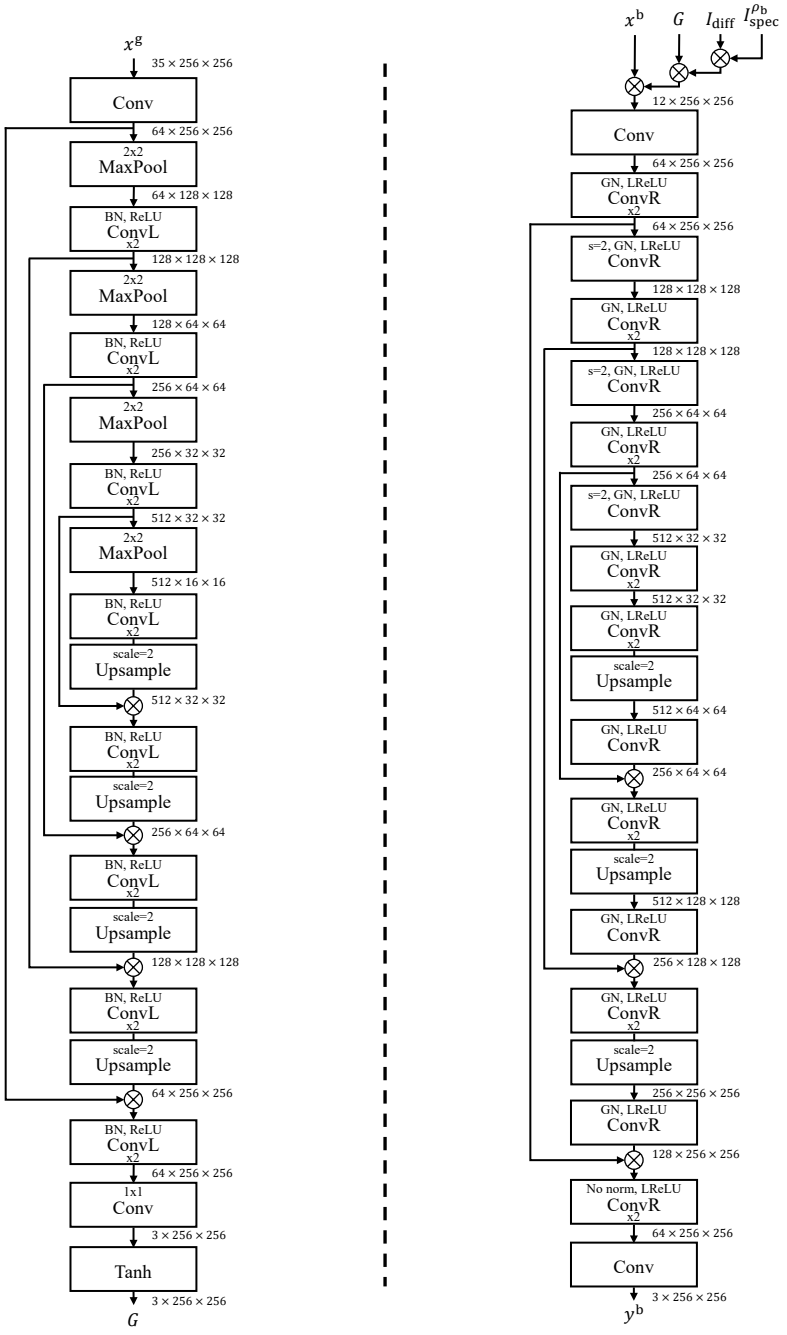
(c) Sun Position Classifier (***SunPosCls***)

(d) Feature Extractor (***FeatExtractor***)

(e) SH Coefficient Estimator (***SHEst***)

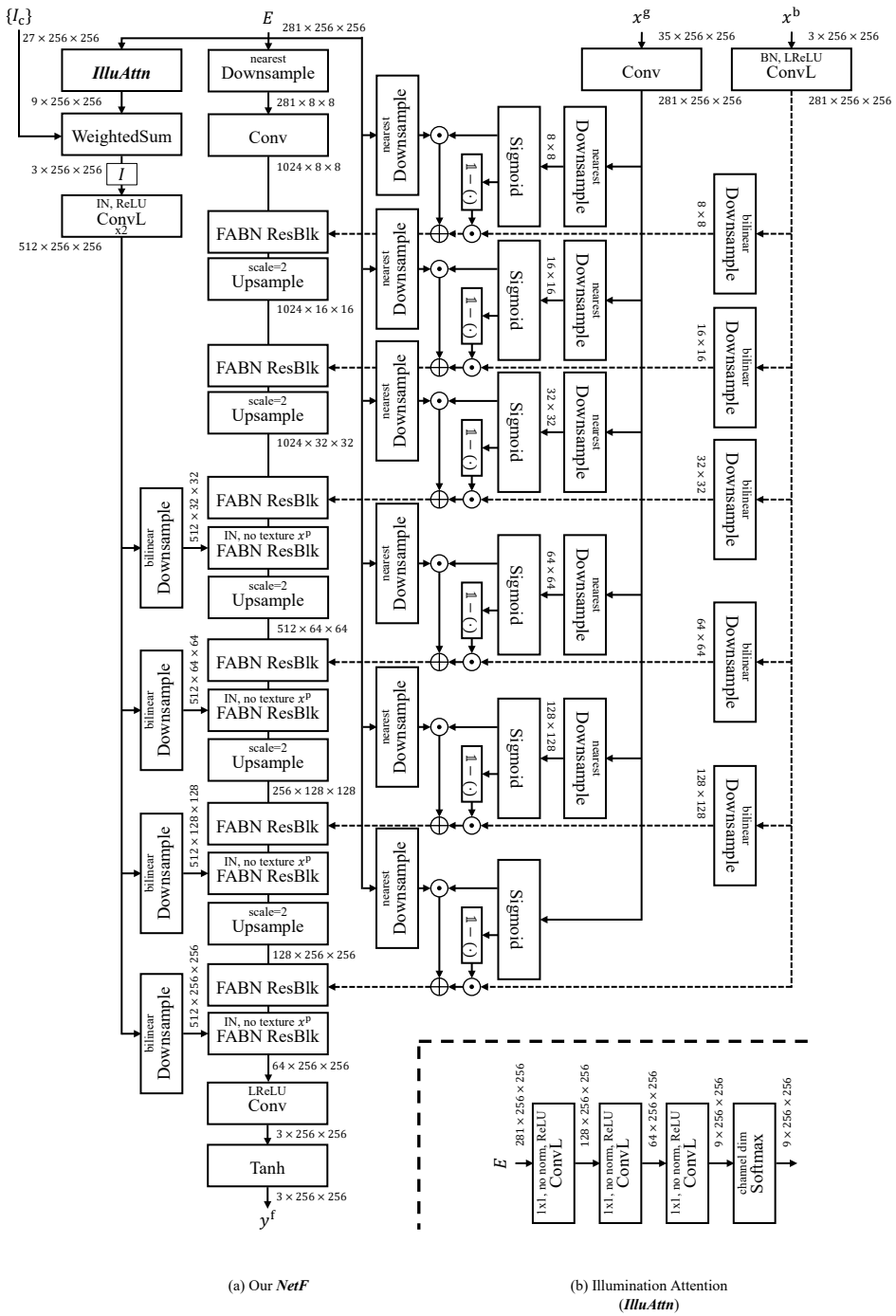
(f) Sun Attributes Estimator (***SunAttEst***)

Figure 19: Network architectures of our pipeline.



(a) Our ***NetS*** (b) Our ***NetB***

Figure 20: Network architectures of our pipeline (cont'd).



(a) Our ***NetF***

(b) Illumination Attention (***IlluAttn***)

Figure 21: Network architectures of our pipeline (cont'd).

References

[1] Z. Bauer, F. Gomez-Donoso, E. Cruz, S. Orts, and M. Cazorla. UASOL, a large-scale high-resolution outdoor stereo dataset. *Scientific Data*, 6:1–14, 2019.

[2] J. F. Blinn. Models of light reflection for computer synthesized pictures. *SIGGRAPH Comput. Graph.*, 11(2):192–198, 1977.

[3] L. A. Gatys, A. S. Ecker, and M. Bethge. Image style transfer using convolutional neural networks. In *Proc. of Computer Vision and Pattern Recognition*, 2016.

[4] J. J. A. Guerreiro, M. Nakazawa, and B. Stenger. PCT-Net: Full resolution image harmonization using pixel-wise color transformations. In *Proc. of Computer Vision and Pattern Recognition*, 2023.

[5] Z. Guo, Z. Gu, B. Zheng, J. Dong, and H. Zheng. Transformer for image harmonization and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(11):12960–12977, 2023.

[6] Z. Huang, N. Zhao, and J. Liao. UniColor: A unified framework for multi-modal colorization with transformer. In *Proc. of ACM SIGGRAPH Asia*, 2022.

[7] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proc. of International Conference on Machine Learning*, 2015.

[8] N. K. Kalantari and R. Ramamoorthi. Deep high dynamic range imaging of dynamic scenes. *ACM Transactions on Graphics (Proc. of ACM SIGGRAPH)*, 36(4), 2017.

[9] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *Proc. of International Conference on Learning Representations*, 2015.

[10] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. In *Proc. of International Conference on Learning Representations*, 2018.

[11] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al. PyTorch: An imperative style, high-performance deep learning library. In *Proc. of Neural Information Processing Systems*, 2019.

[12] D. Pavllo, A. Lucchi, and T. Hofmann. Controlling style and semantics in weakly-supervised image generation. In *Proc. of European Conference on Computer Vision*, 2020.

[13] R. Ramamoorthi and P. Hanrahan. An efficient representation for irradiance environment maps. In *Proc. of ACM SIGGRAPH*, 2001.

[14] J. Sun, S. Weng, Z. Chang, S. Li, and B. Shi. UniCoRN: A unified conditional image repainting network. In *Proc. of Computer Vision and Pattern Recognition*, 2022.

[15] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, V. Vasudevan, W. Han, J. Ngiam, H. Zhao, A. Timofeev, S. Ettinger, M. Krivokon, A. Gao, A. Joshi, Y. Zhang, J. Shlens, Z. Chen, and D. Anguelov. Scalability in perception for autonomous driving: Waymo open dataset. In *Proc. of Computer Vision and Pattern Recognition*, 2020.

[16] D. Ulyanov, A. Vedaldi, and V. S. Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.

[17] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro. High-resolution image synthesis and semantic manipulation with conditional GANs. In *Proc. of Computer Vision and Pattern Recognition*, 2018.

[18] S. Weng, W. Li, D. Li, H. Jin, and B. Shi. Conditional image repainting via semantic bridge and piecewise value function. In *Proc. of European Conference on Computer Vision*, 2020.

[19] S. Weng, W. Li, D. Li, H. Jin, and B. Shi. MISC: Multi-condition injection and spatially-adaptive compositing for conditional person image synthesis. In *Proc. of Computer Vision and Pattern Recognition*, 2020.

[20] Y. Wu and K. He. Group normalization. In *Proc. of European Conference on Computer Vision*, 2018.

[21] T. Xu, P. Zhang, Q. Huang, H. Zhang, Z. Gan, X. Huang, and X. He. AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks. In *Proc. of Computer Vision and Pattern Recognition*, 2018.

[22] R. Yi, C. Zhu, and K. Xu. Weakly-supervised single-view image relighting. In *Proc. of Computer Vision and Pattern Recognition*, 2023.