LightPainter: Interactive Portrait Relighting with Freehand Scribble

Yiqun Mei[1] He Zhang[2] Xuaner Zhang[2] Jianming Zhang[2] Zhixin Shu[2] Yilin Wang[2]
Zijun Wei[2] Shi Yan[2] HyunJoon Jung[2] Vishal M. Patel[1]

[1]Johns Hopkins University [2]Adobe Inc.

Figure 1. LightPainter is an interactive lighting editing system that takes in an input image with freehand scribbles drawn on top and renders the correspondingly relit portrait. It enables creative portrait lighting editing (left) and allows users to reproduce a target lighting effect with ease (right).

Abstract

Recent portrait relighting methods have achieved realistic results of portrait lighting effects given a desired lighting representation such as an environment map. However, these methods are not intuitive for user interaction and lack precise lighting control. We introduce LightPainter, a scribble-based relighting system that allows users to interactively manipulate portrait lighting effect with ease. This is achieved by two conditional neural networks, a delighting module that recovers geometry and albedo optionally conditioned on skin tone, and a scribble-based module for relighting. To train the relighting module, we propose a novel scribble simulation procedure to mimic real user scribbles, which allows our pipeline to be trained without any human annotations. We demonstrate high-quality and flexible portrait lighting editing capability with both quantitative and qualitative experiments. User study comparisons with commercial lighting editing tools also demonstrate consistent user preference for our method.

1. Introduction

Lighting is a fundamental aspect of portrait photograph, as lights shape the reality, and give the work depth, colorfulness and excitement. Professional photographers [17, 38] spend hours designing lighting such that shadow and highlight are distributed accurately on the subject to achieve the desired photographic look. Getting the exact lighting setups requires years of training, expensive equipment, environment setup, timing, and costly teamwork. Recently, portrait relighting techniques [20, 22, 33, 43, 45, 48, 53, 61, 63] allow users to apply a different lighting condition to a portrait photo. These methods require a given lighting condition: some use an exemplar image [42, 43], which lacks precise lighting control and requires exhaustive image search to find the specific style; some use a high dynamic range (HDR) environment map [33, 45, 48, 53] that is difficult and unintuitive to interpret or edit.

Hand-drawn sketches and scribbles have been shown to be good for user interaction and thus are widely used in various image editing applications [6, 9, 10, 30, 32, 57]. Inspired

2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) | 979-8-3503-0129-8/23/$31.00 ©2023 IEEE | DOI: 10.1109/CVPR52729.2023.00027

979-8-3503-0129-8/23/$31.00 ©2023 IEEE
DOI 10.1109/CVPR52729.2023.00027

by this, we propose ***LightPainter***, a scribble-based interactive portrait relighting system. As shown in Figure 1, LightPainter is an intuitive and flexible lighting editing system that only requires casual scribbles drawn on the input. Unlike widely-used lighting representations such as environment maps and spherical harmonics, it is non-trivial to interpret free-hand scribbles as lighting effects for a number of challenges.

The first challenge is simulating scribbles to mimic real free-hand input as it is impractical to collect a large number of human inputs. In addition, unlike other sketch-based editing tasks [6,9,10,30,32,57] where sketches can be computed from edges or orientation maps, there is no conventional way to connect scribbles with lighting effects. To address such challenge, we propose a scribble simulation algorithm that can generate a diverse set of synthetic scribbles that mimic real human inputs. For an interactive relighting task, scribbles should be flexible and expressive: easy to draw and accurately reflecting the lighting effect, such as changes in local shading and color. Compared to a shading map, scribbles are often "incomplete": users tend to sparsely place the scribbles on a few key areas on the face. Therefore, we propose to use a set of locally connected "shading stripes" to describe local shading patterns, including shape, intensity, and color, and use them to simulate scribbles. To this end, we simulate scribbles by starting from a full shading map and applying a series of operations to generate coarse and sparse shading stripes. We show that training with our synthetic scribbles enables the system to generalize well to real user scribbles from human inputs, with which our model can generate high-quality results with desirable lighting effects.

The second challenge is how to effectively use local and noisy scribbles to robustly represent portrait lighting that is often a global effect. LightPainter uses a carefully designed network architecture and training strategy to handle these discrepancies. Specifically, we introduce a two-step relighting pipeline to process sparse scribbles. The first stage produces a plausible completion of the shading map from the input scribbles and the geometry; the second stage refines the shading and renders the appearance with a learned albedo map. We propose a carefully designed neural network with an augmented receptive field. Compared with commonly-used UNet for portrait relighting [21, 31, 33, 48], our design can better handle the sparse scribbles and achieve geometry-consistent relighting.

Last, there is one major challenge in portrait relighting that originates from the ill-posed nature of the intrinsic decomposition problem. That is to decouple albedo and shading from an image. It is also difficult to address with a learning framework due to the extreme scarcity of realistic labeled data and infinite possible lighting conditions for a scene. In the context of portrait relighting, it means recovering the true skin tone of a portrait subject is very challenging [12, 49]. Instead of trying to collect a balanced large-scale light-stage [8] dataset to capture the continuous and subtle variations in different skin tones, we propose an alternative solution dubbed *SkinFill*. We draw inspiration from the standard makeup routine and design SkinFill to allow users to specify skin tone in our relighting pipeline. We use a *tone map*, a per-pixel skin tone representation, to condition the albedo prediction to follow the exact skin tone as desired. This also naturally enables additional user control at inference time.

Similar to prior work [33, 45, 62], we train our system with a light stage [8] dataset. With our novel designs, LightPainter is a user-friendly system that enables creative and interactive portrait lighting editing. We demonstrate the simple and intuitive workflow of LightPainter through a thorough user study. We show it generates relit portraits with superior photo-realism and higher fidelity compared to state-of-the-art methods. We summarize our contributions as follows:

- We propose LightPainter, a novel scribble-based portrait relighting system that offers flexible user control, allowing users to easily design portrait lighting effects.
- We introduce a novel scribble simulation algorithm that can automatically generate realistic scribbles for training. Combining it with a carefully designed neural relighting module, our system can robustly generalize to real user input.
- We introduce SkinFill to allow users to specify skin tone in the relighting pipeline, which allows data-efficient training and offers additional control to address potential skin tone data bias.

2. Related Work

Portrait Relighting: The pioneering work of Debevec *et al*. [8] presents an advanced illumination rig (i.e. the light stage) to capture per-person reflectance field, which is used to render the subject under novel illuminations. Such technique has been used to create training data for a number of single-image relighting methods [31,33,45,59,62,63]. Portrait relighting has also been formulated as style transfer. Shih *et al*. [42] employ a multi-scale technique to transfer the local statistics of an exemplar to the target image. Shu *et al*. [43] formulate the lighting transfer as a geometry-aware mass transport problem with 3D morphable face model. Using quotient image to achieve relighting is introduced in [34, 41], where they multiply the source image with a ratio map to render novel illuminations. Intrinsic decomposition based approaches [3, 21, 24, 27, 29, 31, 33, 40, 48] factorize a source image into geometry, reflectance, and illumination, and apply novel lighting by conditioning on a

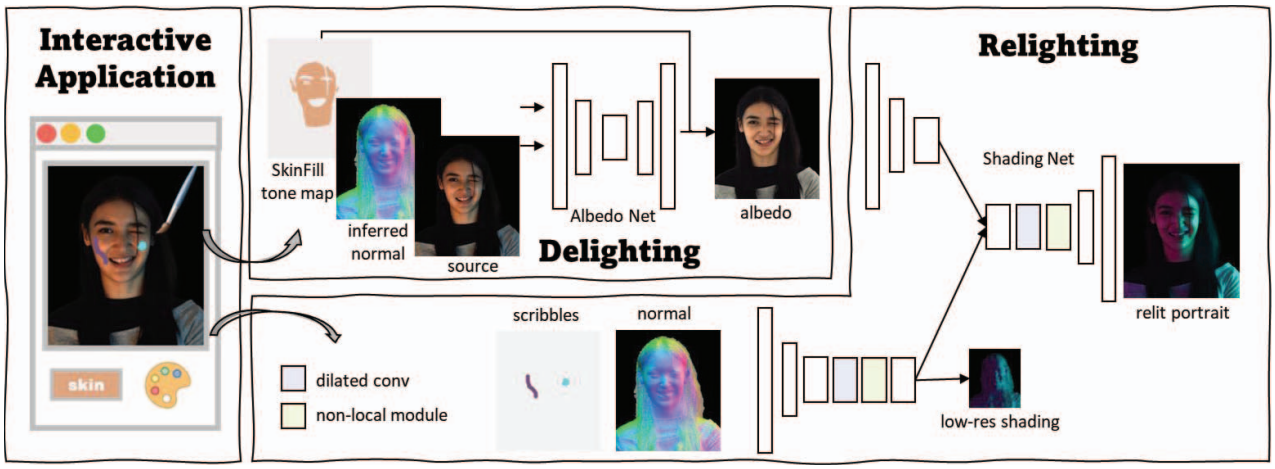


Figure 2. **An overview of LightPainter.** As the user starts scribbling with our interactive application, the neural modules interactively render a realistic relit image that is faithful to the user input.

new illumination. Such technique has received increasing popularity over recent years thanks to recently advanced light stage capturing systems [18] that make direct supervision of intrinsic components possible. Our method also falls into this category.

Lighting Representation: Lighting representation determines how users interact with the system. Several works [39, 63] use spherical harmonics, which is limited to only low-frequency illuminations. Reference-based methods [42, 43] use an image as a proxy to represent lighting, yet the requirement of a matching exemplar image reduces their practicability. A similar argument applies to environment map [33, 45, 48, 53, 63], which is inherently challenging to edit, thus hard to interactive with. Other works [21, 22, 31] model only directional lights, which constrains the type of lighting these methods support.

Image Manipulation with User Scribbles. Scribbling (or sketching) is one of the most intuitive interactions for human to express creative ideas. Drawing-based interface has been widely exploited in various image manipulation tasks [6, 9, 10, 13, 15, 30, 32, 52]. For example, the pioneering work of Eitz *et al*. [5] introduces Sketch2Photo to interactively perform image retrieval and synthesize from user sketch. Yu *et al*. [55] develops deepfill-v2, which allows users to conduct free-form inpainting with scribbled "holes". SketchHairSalon [51] makes hair design easy by drawing desired hair structures. Scribbles and sketches have also been used for face manipulation [36, 57] and image colorization [23, 28]. However, such intuitive interface has not been studied in the context of portrait relighting.

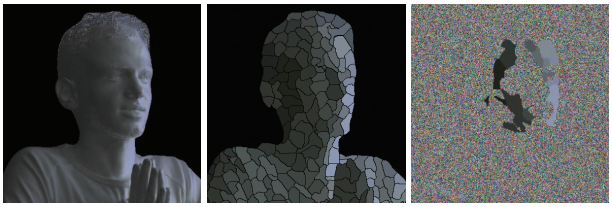
Commercial Lighting Editing Tools. Only a few commercial applications support a complete set of lighting editing capability. Applications such as Facetune [11],"Studio Lighting Mode" in iPhone [1] and "Portrait Relighting" feature in Google Pixel [16] only support a limited set of editing constrained to changing brightness or adding a fixed-color directional light in 2D. The recently released ClipDrop [7] provides more flexible editing by allowing users to place virtual lights with a chosen color, intensity, distance and radius. However, it does not support removing existing illuminations from the scene. Further, to pursue creative lighting effects, it is possible that users have to manually tune multiple lights at the same time. In the user study, we will show this process is difficult for many novices.

3. Method

In this section, we describe the framework of LightPainter. As shown in Figure 2, LightPainter consists of a frontend interactive application for the user to scribble and a backend performing relighting with respect to the user input. A detailed walk-through of the frontend interface can be found in the supplementary. Here we focus our discussion on the backend. Specifically, the backend comprises two conditional neural networks: a skin-tone-conditioned delighting module and a scribble-conditioned relighting module. The delighting stage recovers the geometry representing per-pixel surface normal, and an albedo image that optionally follows a user-selected skin tone. After delighting, estimated normal and albedo are transmitted to the relighting module, which renders the portrait under the lighting condition that respects the user's scribbles. In Section 3.1 and 3.2, we describe each stage in detail. We define our training objectives in Section 3.3.

3.1. Delighting Module

Inspired by Total Relighting [33], LightPainter uses two networks to separately estimate geometry and reflectance. For geometry, we use the existing algorithm [2] and fine-tune on our dataset. Our main difference from prior works is the reflectance prediction, where we propose a skin-tone-



(a) Shading (b) Superpixels (c) Simulated scribbles

Figure 3. **An example of the simulated scribbles.** (a) A complete shading obtained from the Phong shading [35]. (b) Segmented superpixels after quantization and color/intensity average. Each segment is coarse in both shape and intensity. (c) Simulated scribbles generated by sampling from (b). We fill in Gaussian noise into the empty region and background.

conditioned albedo model.

3.1.1 Skin-tone-conditioned Albedo Net

Data-driven albedo prediction is challenging as it requires a fully comprehensive and balanced dataset to avoid any skin tone bias. To address this challenge and recover an accurate albedo, we explore an alternative solution by leveraging user control. This is built upon the observation that the skin tone of a subject can be easily specified in practice, for example, using the skin swatches on a cosmetics website[1]. We thus propose to leverage user interaction to aid our albedo generation. Specifically, we ask the user to optionally provide a skin color to the system. As shown in Figure 2, the network then generates the albedo conditioned on the received color vector $\hat{v}$, along with the estimated normal and the source portrait. At training time, $\hat{v}$ can be extracted as the mean skin color from the ground truth albedo. For inference, if $\hat{v}$ is not provided by the user, the albedo generation scheme falls back to a standard unconditional method.

It is crucial to determine how to best leverage $\hat{v}$. Intuitively, the skin tone should be accessible by all pixels in the skin region for guidance. The network must also be designed to follow the guidance so as the generated albedo matches the user's desire. In the following, we introduce a new technique, dubbed *SkinFill*, by drawing inspiration from the makeup routine.

SkinFill. In standard makeup routine, skin color can be modified by blending foundation smoothly over the facial skin, followed by local retouching. This inspires two design choices of SkinFill: (1) matching the exact skin tone by uniformly shifting the pixel value in the skin region, (2) recovering local facial details from the input to ensure fidelity and realism. Specifically, SkinFill first creates a per-pixel skin-tone representation T, the *tone map*, by filling in the skin-parsing mask M_{skin} with color $\hat{v}$, i.e. $T = M_{skin} \odot \hat{v}$.

[1]For example, skin-tone finder provided by Sephora

The tone map T is then used to condition the network for better facial detail recovery, and directly added to the network prediction to shift the pixel value in the skin region (see the Delighting part in Figure 2). SkinFill brings several benefits: (1) the tone map makes user guidance easily accessible at all skin pixels. (2) uniformly shifting the skin color towards $\hat{v}$ enforces the prediction to follow the user's intention. (3) The network can focus on recovering the local facial details without the need of regressing the skin tone.

3.2. Relighting Module

In this section, we describe the scribble-based relighting module in detail. To begin with, we introduce a scribble simulation algorithm that enables training a network with synthetic scribbles that resemble real users' inputs. We also describe the shading network that renders a realistic relit portrait conditioned on the input scribbles.

3.2.1 Scribble Simulation

In order to train a relighting network that is conditioned on scribbles, we propose a scribble simulation algorithm that automatically generates scribbles that mimic real user inputs and with large variations.

As aforementioned, we use "shading stripes" to represent user scribbles, which reflect local shading patterns. Different shading levels naturally correspond to different scribble intensities. Hence the first step in our simulation algorithm is to obtain a full shading map. This is accomplished by rendering with the Phong shading [35] model using the ground truth geometry and environment map. As shown in Figure 3 (a), the rendered shading map can be viewed as an "ideal" scribble containing detailed and complete lighting information. However, drawings from novices are usually noisy – irregular and incomplete. To make our system robust to these imperfect inputs, we apply a series of augmentations to model these defects in real scribbles. Specifically, we first convert the shading into Lab color space and "coarsen" the luminance channel L by randomly quantizing it into multiple bins. Then we perform a superpixel segmentation using SEEDS [4] and average the color within each segment. As shown in Figure 3 (b), each segment ends up being coarse in intensity and shape, similar to the noisy scribble that novices tend to draw. Finally, we "sparsify" the simulated scribbles by randomly sampling a small subset of segments at each training step. The sampling rate is drawn from a truncated exponential distribution with $\lambda = 3$, which results in mostly sparse inputs. In addition, we always keep the segments of top 5% brightest and darkest intensity to make sure our sampling captures the full dynamic range of shading. On the other hand, this also ensures that the most representative lighting information is preserved to help the network reasonably complete the full shading map. An example of simulated scribbles is illus-

trated in Figure 3 (c). More details and hyper-parameter choices can be found in the supplement.

3.2.2 Scribble-conditioned Shading Net

Two-Step Relighting Pipeline. In prior works, relighting is often performed in a single step by taking the lighting condition as a conditional input. In our interactive setting, the lighting condition is from the user scribbles, which can be local, sparse and coarse. We found that directly predicting the relit images from the scribbles does not perform well, where the network would struggle at generating a plausible completion of shading.

To address this challenge, we introduce a two-step relighting pipeline: The first step completes shading following the subject geometry, and the second stage refines the completed shading to render a relit image. As shown in Figure 2, we use the bottom branch to complete a low-resolution shading map conditioned on the normal and scribbles. The output is supervised with the ground truth shading, which enforces the network to learn to propagate sparse lighting information following the surface geometry, which encourages geometry-consistent relighting. The shading feature is then concatenated with the albedo feature (encoded by the upper brunch), and transmitted to the decoder. The decoder then refines the completed shading and renders the final image.

Improved Network Architecture. We adapt our network architecture to better tolerate "incomplete" user scribbles. Intuitively, the receptive field of the network should be sufficiently large so that the network can leverage the global context given sparse and local user input. To achieve this, we build our network with a U-Shaped structure [37] and further adapt it with additional dilated convolutions [54] and non-local modules [46]. These improvements allow a global receptive field and enhance the information flow among distant locations, thus are more suitable for our task. We will demonstrate that our architectural design is crucial for faithful relighting from sparse scribbles.

3.3. Training Objective

To ensure both realism and fidelity of the relit image, our neural network optimizes the following training objectives:

Reconstruction loss $\mathcal{L}_{R_{alb}}$ and $\mathcal{L}_{R_{relit}}$: the standard $L1$ distance between the generated albedo/portrait and the ground truth albedo/portrait to ensure content fidelity.

Perceptual loss [60] $\mathcal{L}_{P_{alb}}$ and $\mathcal{L}_{P_{relit}}$: the features-wise distance of the predicted albedo/portrait and ground truth albedo/portrait extracted by a pre-trained VGG [44]. This loss is used to improve visual quality.

Shading reconstruction loss $\mathcal{L}_{R_{shad}}$: the standard $L1$ distance between the completed low-resolution shading and the downscaled ground truth to enforce shading completion.

The overall loss can be expressed as follows:

$$\mathcal{L} = \mathcal{L}_{R_{alb}} + \mathcal{L}_{P_{alb}} + \mathcal{L}_{R_{relit}} + \mathcal{L}_{P_{relit}} + \mathcal{L}_{R_{shad}} \quad (1)$$

4. Data and Implementation Details

Data Preparation. Following prior practice [33, 45, 62], we use a light stage [8] to collect training and testing OLAT data. Our capture system is structurally similar to that used in [45]. Our light stage contains 160 programmable LED-based lights and 4 frontal-viewing high-speed cameras. The detailed configuration can be found in the supplementary material. Our dataset contains 59 subjects. Each subject is photographed with 5-15 different poses and accessories, resulting in 2123 OLAT sequences in total. A subset of 13 subjects with diverse races and different genders are used for testing. The ground truth normal is computed following the algorithm described in [33]. The ground truth diffuse albedo is acquired by capturing the subject in a flat unidirectional lighting condition with all LED lights turned on.

To obtain a paired dataset for supervised training, we render our OLAT images under diverse lighting environments, which are collected from the Laval Indoor [14] and Outdoor HDR datasets and PolyHaven [19]. We collect in total 2571 real environment maps. We in addition create 2000 synthetic environment maps by placing colored eclipse shapes on a black canvas. We randomly select a subset of 450 environment maps for testing. We also augment lighting by randomly rotating the environment maps when rendering data, resulting in 870K training samples. For the test set, we randomly pair each test OLAT sequence with two lighting environments to form input and output pairs, resulting in 757 testing pairs.

Implementation and Training Details. Both Shading Net and Albedo Net have an encoder-decoder structure with three downsampling and upsampling layers. Multiple standard convolutions, dilated convolutions [54] and non-local attentions [46] are inserted at the bottleneck of each network. Further details can be found in the supplementary material. For training, we resize the rendered data to 800×600 resolution and randomly crop 512×512 region from 32 images to form a mini-batch. The network is trained using Adam optimizer [26]. The learning rate is set to 1e-4 for the first 2 epochs, then reduced to a half after each epoch. We stop the training after 5 epochs. The proposed model is implemented using PyTorch and the training takes about 1 day on 8 Nvidia A100 GPUs.

5. Experiments

We now demonstrate the high-quality portrait relighting capability of LightPainter via extensive evaluations, comparisons, and user studies. We also provide ablation studies to demonstrate the benefits of our system design.

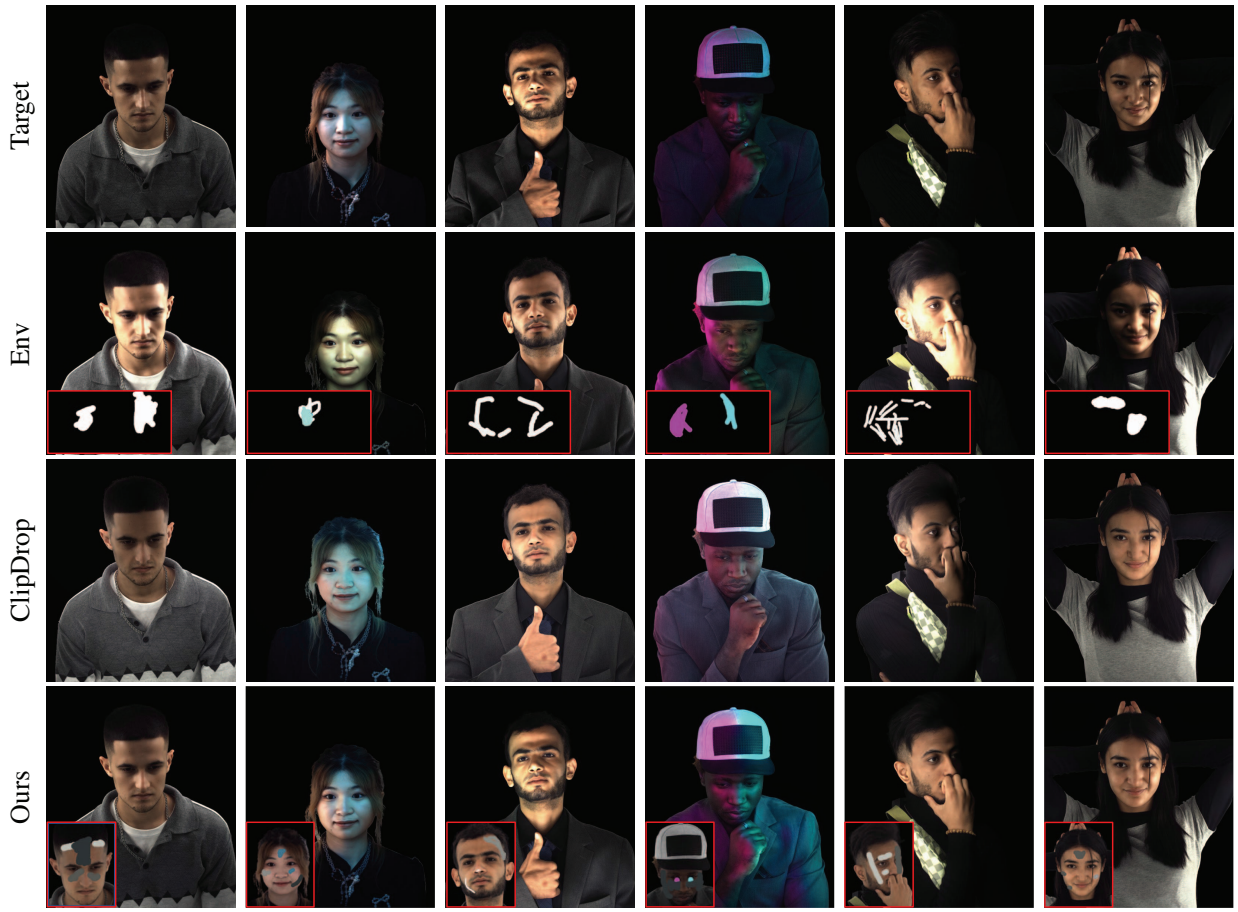


Figure 4. **Visual comparisons on user-generated images using three relighting systems.** User-drawn scribbles and environment maps are shown as insets. Lighting effects that are produced by our method are the most faithful and consistent with the target. **Best viewed by zooming to 4X.**

Evaluation Metrics. We report perceptual metrics LPIPS [60], NIQE [58], and pixel similarity metrics PSNR and SSIM [47]. In addition, we use the Deg (cosine similarity between LightCNN [50] features) to evaluate identity preservation capability. All results are computed within the subject mask pre-computed using [56].

5.1. User Study

To demonstrate LightPainter can benefit general users on portrait lighting editing, we perform a user study to evaluate the quality and user experience of LightPainter. We recruit general users to conduct portrait relighting with three systems that use different interactive approaches:

- ClipDrop [7][2], a commercial lighting editing web service, where users can place virtual lights into the scene with a chosen light color, intensity, distance and radius.
- Env: a re-implementation of Total Relighting [33], the state-of-the-art portrait relighting method. With this tool, the user provides a hand-drawn "environment map" to perform lighting editing.
- LightPainter, our scribble-based system.

[2]www.clipdrop.co/relight

Table 1. Quantitative evaluation on user-generated images.

Methods	LPIPS↓	NIQE↓	Deg↑	PSNR↑	SSIM↑
ENV	0.1359	6.067	0.8939	20.06	0.7832
ClipDrop [7]	0.1435	6.967	0.8917	20.49	0.6116
LightPainter (ours)	**0.0868**	**5.643**	**0.9379**	**24.94**	**0.8373**

For this experiment, we provide a *target image* with a random lighting effect and assign a task to the users to reproduce this lighting effect on an input *source image* with each tool individually. The source image and target image are from our testing set and only differ in lighting. We impose a time limit of 2 minutes per task for each tool. Note that ClipDrop does not remove existing lighting effects on source images, we use the albedo image (from our test set) as the source image for fair comparisons.

In Figure 4, we show a random subset of visual results from the user study. Using Env or ClipDrop [7], users are often unable to faithfully reproduce the target lighting ef-

Figure 5. **Examples of how user leverage SkinFill on skin color retouching.** "Auto" denotes albedo generated by the automatic prediction mode of the proposed LightPainter. The selected skin color swatch is appended at the end of each row. Both TR [33] and "Auto" suffer from skin-tone bias. In contrast, with SkinFill, LightPainter can predict the correct skin color that follows the user's desire.

fects. We notice that results from using Env exhibit a significant mismatch in both brightness and shading patterns. This shows the difficulty for users to interpret the lighting position and intensity, and associate lighting effects on an image with environment map representations. While ClipDrop [7] is easier for users to interact with, the desired detailed lighting patterns in the target image, such as the highlight, are still largely absent in the results. In contrast, LightPainter allows users to faithfully reproduced the target lighting effect, with convincing details, within two minutes.

We also provide quantitative evaluations (computed between user-generated results and target images) on each method, as shown in Table 1. The quantitative results are collected on 40 trials performed by 20 users. LightPainter achieves the best performance on all metrics, which not only produces the target lighting most closely, but also shows the best image quality and identity preservation capability.

We gathered feedback from the 20 participated users on their experience with each relighting system. We summarize the findings as follows: 1. All users are satisfied with our scribble-based relighting scheme, and feel that drawing over the portrait is very easy to operate. 2. Most users (17/20) feel that editing/drawing environment map is confusing. 3. More than half of users (12/20) feel tuning virtual lights (i.e. guessing the color, intensity and distance of each light associated with the scene) is very difficult and it is hard to obtain the desirable lighting effect.

5.2. Skin-tone Control

With SkinFill, LightPainter can predict the albedo that respects the user-specified skin color. This grants the user with the flexibility of tuning skin tone. We demonstrate its use case in Figure 5. This also helps resolve the potential

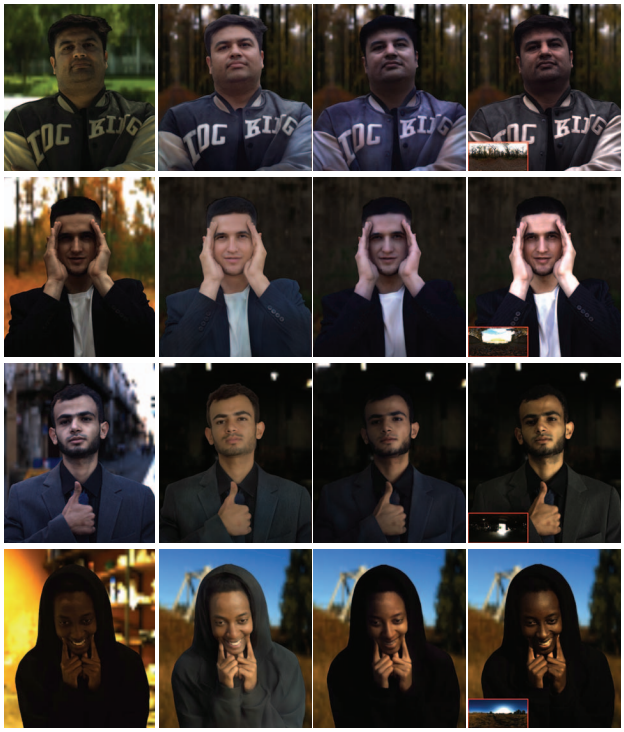


Figure 6. **Qualitative comparisons on environment-map-based portrait relighting.** Our method can produce on-par or better results comparing with TR [33]. Inputs and targets (ground truth) are generated using the test set of light stage subjects and environment maps not seen during training.

data bias and ambiguity in predicting an accurate skin tone from a single image. As already shown in machine learning based approaches, the skin color of the predicted albedo from both total relighting [33] and the automatic mode of LightPainter is inaccurate and appears washed-out. In contrast, with additional user interference with SkinFill, LightPainter can produce a more faithful albedo that respects the user's intent. More experiments can be found in the supplement.

5.3. Comparisons with State-of-the-art Methods

To further demonstrate the relighting quality of LightPainter, we compare it with the state-of-the-art methods SIPR-W [48] and Total Relighting [33] (TR) on environment-map-based relighting. Since LightPainter is not designed for using environment map as its lighting representation, we perform relighting with "estimated" scribbles instead of user scribbles. Specifically, we render the shading map using the Phong model [35] with the estimated normal map to replace the user scribbles. These estimated scribbles are in fact "ideal" inputs for LightPainter, with more complete shading information than either the simulated scribbles we used for training or the real user scribbles. We adopt the off-the-shelf image matting method [56]

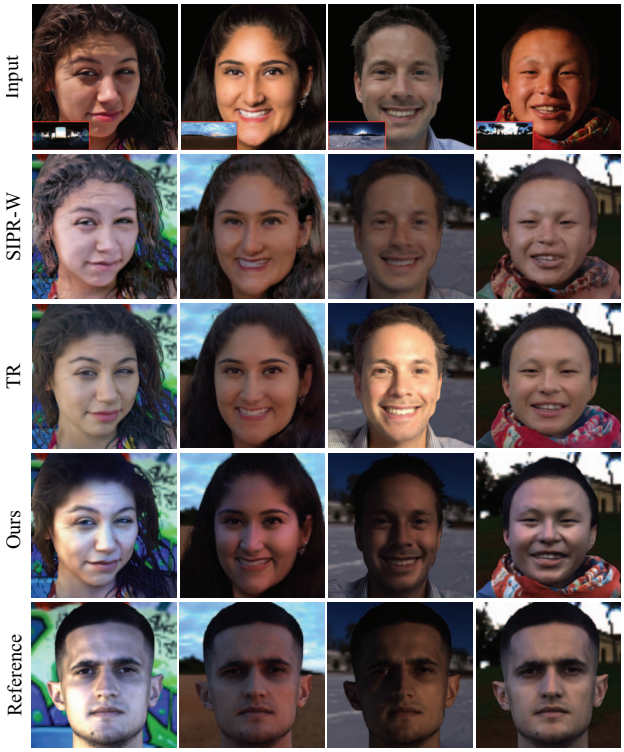


Figure 7. **Qualitative comparisons on in-the-wild face relighting.** We compare relighting results with SIPR-W [48] (row-2) and TR [33] (row-3). The environment maps are shown as insets (row-1). We provide a reference image (row-5) rendered with OLAT data as guidance of the lighting effect under the input environment.

to extract the foreground portrait and composite it with the new background. We report (1) quantitative and qualitative results on portrait/upper-body relighting on our testing data, and (2) qualitative comparisons on in-the-wild images from FFHQ [25]. Results of SIPR-W [48] and TR [33] are directly obtained from their authors.

Evaluation with Light Stage Data. For this experiment, 757 ground truth images are rendered with environment maps and OLAT data from the test set using Image-base Relighting [8]. We report the quantitative results in Table 2 and provide comparisons. Our approach performs better in perceptual quality, identity preservation, and image similarity. In Figure 6, we show qualitative comparison with [33]. Our method produces photo-realistic results and respects the target lighting more faithfully.

Evaluation with In-the-wild Images. We demonstrate in-the-wild portrait relighting capability with images from FFHQ [42] dataset. We show qualitative results in Figure 7. Since there is no relighting ground truth, we provide a reference image by rendering a face under the target lighting environment using OLAT and Image-base Relighting [8]. LightPainter generates high-quality relighting results with convincing lighting effects. Our results are more consistent with lighting effects in the reference images compared to TR [33]. Compare to [48], our results are more robust and exhibit no noticeable artifacts.

Table 2. Quantitative comparsion with Total Relighting (TR).

Methods	LPIPS↓	NIQE↓	Deg.↑	PSNR↑	SSIM↑
TR [33]	0.2160	8.137	0.8454	20.31	0.5705
LightPainter	**0.1383**	**6.195**	**0.9076**	**25.71**	**0.8449**

Table 3. Ablation study on enlarged receptive field.

Methods	LPIPS↓	NIQE↓	Deg.↑	PSNR↑	SSIM↑
One-Step	0.1423	7.691	0.8874	23.88	0.5278
w/o Non-Local	0.0875	7.266	0.9170	27.46	0.8152
w/o dilated conv	0.1020	8.318	0.9169	27.18	0.8777
LightPainter	**0.0848**	**7.012**	**0.9310**	**28.48**	**0.8899**

5.4. Ablation Study

We now provide ablation studies to demonstrate the benefit of our key designs in LightPainter. All results are evaluated on our test set using synthetic scribbles, ground-truth albedo and normal with a fixed sampling ratio of 0.3.

Effectiveness of Two-step Relighting Scheme. Our system adopts a two-step relighting scheme to enforce geometry-consistent shading. We investigate its effectiveness by comparing it with a single-step baseline, which directly predicts the relit images from the scribbles without shading completion. The results for one-step baseline are reported in the first row of Table 3. As shown, two-step approach (i.e. LightPainter) significantly improves performance.

Augmented Receptive Field. To handle sparse user input, we adopt non-local blocks [46] and dilated convolutions [54] to enlarge the receptive field of the Shading Net. We conduct experiments to validate this design choice. As shown in Table 3, removing either non-local attention or dilate convolution harms performance.

6. Conclusion

In this paper, we introduce LightPainter, a novel interactive and intuitive portrait relighting system that uses freehand scribbles as the user interface. To address the challenges of relighting with scribbles, we propose novel network designs and a shading-based scribble simulation approach to generate training data. We also introduce a conditional delighting module that predicts high-quality albedo optionally conditioned on skin tone. Experiments and user study demonstrate the state-of-the-art portrait relighting performance of our method. Noticeably, LightPainter allows users to create desirable portrait lighting effects with ease. Discussion of limitations can be found in the supplementary material.

Acknowledgments This work was supported by NSF CARRER award 2045489. We thank Chaowei Company for the support of light stage data.

References

[1] Apple. Use portrait mode on your iphone. https://support.apple.com/en-us/HT208118. 3

[2] Gwangbin Bae, Ignas Budvytis, and Roberto Cipolla. Estimating and exploiting the aleatoric uncertainty in surface normal estimation. In *IEEE International Conference on Computer Vision*, pages 13137–13146, 2021. 3

[3] Jonathan T Barron and Jitendra Malik. Shape, illumination, and reflectance from shading. *IEEE transactions on pattern analysis and machine intelligence*, 37(8):1670–1687, 2014. 2

[4] Michael Van den Bergh, Xavier Boix, Gemma Roig, Benjamin de Capitani, and Luc Van Gool. Seeds: Superpixels extracted via energy-driven sampling. In *European Conference on Computer Vision*, pages 13–26. Springer, 2012. 4

[5] Tao Chen, Ming-Ming Cheng, Ping Tan, Ariel Shamir, and Shi-Min Hu. Sketch2photo: Internet image montage. *ACM Transactions on Graphics*, 28(5):1–10, 2009. 3

[6] Wengling Chen and James Hays. Sketchygan: Towards diverse and realistic sketch to image synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 9416–9425, 2018. 1, 2, 3

[7] ClipDrop. https://clipdrop.co/relight. 3, 6, 7

[8] Paul Debevec, Tim Hawkins, Chris Tchou, Haarm-Pieter Duiker, Westley Sarokin, and Mark Sagar. Acquiring the reflectance field of a human face. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 145–156, 2000. 2, 5, 8

[9] Tali Dekel, Chuang Gan, Dilip Krishnan, Ce Liu, and William T Freeman. Sparse, smart contours to represent and edit images. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3511–3520, 2018. 1, 2, 3

[10] James H Elder and Richard M Goldberg. Image editing in the contour domain. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 374–381. IEEE, 1998. 1, 2, 3

[11] Facetune. https://www.facetuneapp.com/. 3

[12] Haiwen Feng, Timo Bolkart, Joachim Tesch, Michael J. Black, and Victoria Abrevaya. Towards racially unbiased skin tone estimation via scene disambiguation. In *European Conference on Computer Vision*, 2022. 2

[13] Jakub Fišer, Ondřej Jamriška, Michal Lukáč, Eli Shechtman, Paul Asente, Jingwan Lu, and Daniel Sỳkora. Stylit: illumination-guided example-based stylization of 3d renderings. *ACM Transactions on Graphics*, 35(4):1–11, 2016. 3

[14] Marc-André Gardner, Kalyan Sunkavalli, Ersin Yumer, Xiaohui Shen, Emiliano Gambaretto, Christian Gagné, and Jean-François Lalonde. Learning to predict indoor illumination from a single image. *ACM Transactions on Graphics*, 36(6):1–14, 2017. 5

[15] Arnab Ghosh, Richard Zhang, Puneet K Dokania, Oliver Wang, Alexei A Efros, Philip HS Torr, and Eli Shechtman. Interactive sketch & fill: Multiclass sketch-to-image translation. In *IEEE International Conference on Computer Vision*, pages 1171–1180, 2019. 3

[16] Google. Portrait light: Enhancing portrait lighting with machine learning. https://ai.googleblog.com/2020/12/portrait-light-enhancing-portrait.html. 3

[17] Christopher Grey. *Master lighting guide for portrait photographers*. Amherst Media, 2014. 1

[18] Kaiwen Guo, Peter Lincoln, Philip Davidson, Jay Busch, Xueming Yu, Matt Whalen, Geoff Harvey, Sergio Orts-Escolano, Rohit Pandey, Jason Dourgarian, et al. The relightables: Volumetric performance capture of humans with realistic relighting. *ACM Transactions on Graphics*, 38(6):1–19, 2019. 3

[19] Poly Haven. Poly haven. 5

[20] Andrew Hou, Michel Sarkis, Ning Bi, Yiying Tong, and Xiaoming Liu. Face relighting with geometrically consistent shadows. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4217–4226, 2022. 1

[21] Andrew Hou, Michel Sarkis, Ning Bi, Yiying Tong, and Xiaoming Liu. Face relighting with geometrically consistent shadows. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4217–4226, June 2022. 2, 3

[22] Andrew Hou, Ze Zhang, Michel Sarkis, Ning Bi, Yiying Tong, and Xiaoming Liu. Towards high fidelity face relighting with realistic shadows. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 14719–14728, June 2021. 1, 3

[23] Yi-Chin Huang, Yi-Shin Tung, Jun-Cheng Chen, Sung-Wen Wang, and Ja-Ling Wu. An adaptive edge detection based colorization algorithm and its applications. In HongJiang Zhang, Tat-Seng Chua, Ralf Steinmetz, Mohan S. Kankanhalli, and Lynn Wilcox, editors, *The 13th ACM International Conference on Multimedia*, pages 351–354. ACM, 2005. 3

[24] Chaonan Ji, Tao Yu, Kaiwen Guo, Jingxin Liu, and Yebin Liu. Geometry-aware single-image full-body human relighting. October 2022. 2

[25] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019. 8

[26] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 5

[27] Ha A Le and Ioannis A Kakadiaris. Illumination-invariant face recognition with deep relit face images. In *IEEE Winter Conference on Applications of Computer Vision*, pages 2146–2155. IEEE, 2019. 2

[28] Anat Levin, Dani Lischinski, and Yair Weiss. Colorization using optimization. *ACM Transactions on Graphics*, 23(3):689–694, 2004. 3

[29] Chen Li, Kun Zhou, and Stephen Lin. Intrinsic face image decomposition with human face priors. In *European Conference on Computer Vision*, pages 218–233. Springer, 2014. 2

[30] Kamyar Nazeri, Eric Ng, Tony Joseph, Faisal Z Qureshi, and Mehran Ebrahimi. Edgeconnect: Generative image inpainting with adversarial edge learning. *arXiv preprint arXiv:1901.00212*, 2019. 1, 2, 3

[31] Thomas Nestmeyer, Jean-François Lalonde, Iain Matthews, and Andreas Lehrmann. Learning physics-guided face relighting under directional light. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5124–5133, 2020. 2, 3

[32] Kyle Olszewski, Duygu Ceylan, Jun Xing, Jose Echevarria, Zhili Chen, Weikai Chen, and Hao Li. Intuitive, interactive beard and hair synthesis with generative models. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 7446–7456, 2020. 1, 2, 3

[33] Rohit Pandey, Sergio Orts Escolano, Chloe Legendre, Christian Haene, Sofien Bouaziz, Christoph Rhemann, Paul Debevec, and Sean Fanello. Total relighting: learning to relight portraits for background replacement. *ACM Transactions on Graphics*, 40(4):1–21, 2021. 1, 2, 3, 5, 6, 7, 8

[34] Pieter Peers, Naoki Tamura, Wojciech Matusik, and Paul Debevec. Post-production facial performance relighting using reflectance transfer. *ACM Transactions on Graphics*, 26(3):52–es, 2007. 2

[35] Bui Tuong Phong. Illumination for computer generated pictures. *Communications of the ACM*, 18(6):311–317, 1975. 4, 7

[36] Tiziano Portenier, Qiyang Hu, Attila Szabó, Siavash Arjomand Bigdeli, Paolo Favaro, and Matthias Zwicker. Faceshop: deep sketch-based face image editing. *ACM Transactions on Graphics*, 37(4):99, 2018. 3

[37] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 5

[38] James Boniface Schriever and Thomas Harrison Cummings. *Complete Self-instructing Library of Practical Photography: Negative retouching; etching and modeling; encyclopedic index*, volume 8. American school of art and photography, 1909. 1

[39] Soumyadip Sengupta, Angjoo Kanazawa, Carlos D Castillo, and David W Jacobs. Sfsnet: Learning shape, reflectance and illuminance of facesin the wild'. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 6296–6305, 2018. 3

[40] Davoud Shahlaei and Volker Blanz. Realistic inverse lighting from a single 2d image of a face, taken under unknown and complex lighting. In *2015 11th IEEE international conference and workshops on automatic face and gesture recognition (FG)*, volume 1, pages 1–8. IEEE, 2015. 2

[41] Amnon Shashua and Tammy Riklin-Raviv. The quotient image: Class-based re-rendering and recognition with varying illuminations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(2):129–139, 2001. 2

[42] YiChang Shih, Sylvain Paris, Connelly Barnes, William T Freeman, and Frédo Durand. Style transfer for headshot portraits. *ACM Transactions on Graphics*, 33(4):1–14, 2014. 1, 2, 3, 8

[43] Zhixin Shu, Sunil Hadap, Eli Shechtman, Kalyan Sunkavalli, Sylvain Paris, and Dimitris Samaras. Portrait lighting transfer using a mass transport approach. *ACM Transactions on Graphics*, 36(4):1, 2017. 1, 2, 3

[44] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 5

[45] Tiancheng Sun, Jonathan T Barron, Yun-Ta Tsai, Zexiang Xu, Xueming Yu, Graham Fyffe, Christoph Rhemann, Jay Busch, Paul E Debevec, and Ravi Ramamoorthi. Single image portrait relighting. *ACM Transactions on Graphics*, 38(4):79–1, 2019. 1, 2, 3, 5

[46] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 7794–7803, 2018. 5, 8

[47] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 6

[48] Zhibo Wang, Xin Yu, Ming Lu, Quan Wang, Chen Qian, and Feng Xu. Single image portrait relighting via explicit multiple reflectance channel modeling. *ACM Transactions on Graphics*, 39(6):1–13, 2020. 1, 2, 3, 7, 8

[49] Joshua Weir, Junhong Zhao, Andrew Chalmers, and Taehyun Rhee. Deep portrait delighting. *European Conference on Computer Vision*, 2022. 2

[50] Xiang Wu, Ran He, Zhenan Sun, and Tieniu Tan. A light cnn for deep face representation with noisy labels. *IEEE Transactions on Information Forensics and Security*, 13(11):2884–2896, 2018. 6

[51] Chufeng Xiao, Deng Yu, Xiaoguang Han, Youyi Zheng, and Hongbo Fu. Sketchhairsalon: Deep sketch-based hair image synthesis. *ACM Transactions on Graphics*, 40(6):1–16, 2021. 3

[52] Shuai Yang, Zhangyang Wang, Jiaying Liu, and Zongming Guo. Deep plastic surgery: Robust and controllable image editing with human-drawn sketches. In *European Conference on Computer Vision*, 2020. 3

[53] Yu-Ying Yeh, Koki Nagano, Sameh Khamis, Jan Kautz, Ming-Yu Liu, and Ting-Chun Wang. Learning to relight portrait images via a virtual light stage and synthetic-to-real adaptation. *ACM Transactions on Graphics*, 2022. 1, 3

[54] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. In Yoshua Bengio and Yann LeCun, editors, *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016. 5, 8

[55] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Free-form image inpainting with gated convolution. In *IEEE International Conference on Computer Vision*, pages 4471–4480, 2019. 3

[56] Qihang Yu, Jianming Zhang, He Zhang, Yilin Wang, Zhe Lin, Ning Xu, Yutong Bai, and Alan Yuille. Mask guided matting via progressive refinement network. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1154–1163, 2021. 6, 7

[57] Yu Zeng, Zhe Lin, and Vishal M Patel. Sketchedit: Mask-free local image manipulation with partial sketches. In *IEEE*

Conference on Computer Vision and Pattern Recognition, pages 5951–5961, 2022. 1, 2, 3

[58] Lin Zhang, Lei Zhang, and Alan C Bovik. A feature-enriched completely blind image quality evaluator. *IEEE Transactions on Image Processing*, 24(8):2579–2591, 2015. 6

[59] Longwen Zhang, Qixuan Zhang, Minye Wu, Jingyi Yu, and Lan Xu. Neural video portrait relighting in real-time via consistency modeling. In *IEEE International Conference on Computer Vision*, pages 802–812, 2021. 2

[60] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 586–595, 2018. 5, 6

[61] Xuaner Zhang, Jonathan T Barron, Yun-Ta Tsai, Rohit Pandey, Xiuming Zhang, Ren Ng, and David E Jacobs. Portrait shadow manipulation. *ACM Transactions on Graphics*, 39(4):78–1, 2020. 1

[62] Xiuming Zhang, Sean Fanello, Yun-Ta Tsai, Tiancheng Sun, Tianfan Xue, Rohit Pandey, Sergio Orts-Escolano, Philip Davidson, Christoph Rhemann, Paul Debevec, et al. Neural light transport for relighting and view synthesis. *ACM Transactions on Graphics*, 40(1):1–17, 2021. 2, 5

[63] Hao Zhou, Sunil Hadap, Kalyan Sunkavalli, and David W Jacobs. Deep single-image portrait relighting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7194–7202, 2019. 1, 2, 3