

Generating Digital Painting Lighting Effects via RGB-space Geometry

Anonymous Author(s)

Supplementary Material

1 We provide raw experimental data, present additional results, and elaborate the
2 details of our experiments.

3 1 Additional Ablative Results

4 Please refer to the ablative video for the additional ablation study. We replace our estimated stroke
5 density with several possible alternatives as follows.

6 **Luminance map.** Given the RGB channels $\{\mathbf{R}, \mathbf{G}, \mathbf{B}\} \subset \mathbb{R}^{H \times W \times 3}$, we compute the luminance
7 map $L_{\text{luminance}}$ with

$$L_{\text{luminance}} = 0.299\mathbf{R} + 0.587\mathbf{G} + 0.114\mathbf{B} \quad (1)$$

8 **Inverted luminance map.** To study whether simple inverting can help yielding better results, we
9 compute the inverted luminance map L_{inverted} with

$$L_{\text{inverted}} = 1.0 - L_{\text{luminance}} \quad (2)$$

10 **Scaled luminance map.** To study whether simple scaling can help yielding better results, we
11 compute the scaled luminance map L_{scaled} with

$$L_{\text{scaled}} = 0.5 + 0.5(L_{\text{luminance}} - 0.5) \quad (3)$$

12 **Scaled Inverted luminance map.** We also include the scaled inverted luminance map
13 $L_{\text{inverted\&scaled}}$ with

$$L_{\text{inverted\&scaled}} = 1.0 - L_{\text{scaled}} \quad (4)$$

14 **Saturation map.** To study whether saturation can replace stroke density, we compute the saturation
15 map $S_{\text{saturation}}$ with

$$S_{\text{saturation}} = \frac{\max\{\mathbf{R}, \mathbf{G}, \mathbf{B}\} - \min\{\mathbf{R}, \mathbf{G}, \mathbf{B}\}}{\max\{\mathbf{R}, \mathbf{G}, \mathbf{B}\}} \quad (5)$$

16 **Simple Euclidean distance map.** Finally, we also include a simple Euclidean distance as a color
17 geometry baseline. We first compute the global mean color of $\{\mathbf{R}, \mathbf{G}, \mathbf{B}\}$ channels, denoted by
18 $\{\mathbf{r}_{\text{mean}}, \mathbf{g}_{\text{mean}}, \mathbf{b}_{\text{mean}}\}$. Then, we compute the baseline Euclidean distance map D_{distance} with

$$D_{\text{distance}} = \sqrt{(\mathbf{R} - \mathbf{r}_{\text{mean}})^2 + (\mathbf{G} - \mathbf{g}_{\text{mean}})^2 + (\mathbf{B} - \mathbf{b}_{\text{mean}})^2} \quad (6)$$

19 and then these computed alternative maps are directly used to replace the stroke density maps within
20 our algorithm pipeline as described in the main paper.

21 2 Methodology

22 In the main paper, we have discussed the lighting effect generating of pointed light sources. Here we
23 present the use cases of light from infinity and spot light. We show examples in Fig. 1.

24 2.1 Light Source at Infinity

25 Here we introduce the method to compute a light source at infinity, which is similar to the point light
 26 case described in the main paper. We calculate the same light direction for all pixels. Some other
 27 types of light source, *e.g.*, spot light, can also be calculated similarly. Given a light direction unit
 28 vector $\hat{\mathbf{l}} = [\hat{l}_x, \hat{l}_y, \hat{l}_z]^\top$, an angle defined by $\tan \hat{\theta} = \hat{l}_y/\hat{l}_x$, and a pixel location $\mathbf{p} = [p_x, p_y]^\top$, we can
 29 replace the computation of the coarse light field map \mathbf{C}_i from Eq.(15) in in the main text with

$$e_i(p_x, p_y) = \frac{[n_i^*(\mathbf{p}, \hat{\theta}, \Delta p_d) - n_i^*(\mathbf{p}, \hat{\theta}, 0), \Delta p_d]}{|[n_i^*(\mathbf{p}, \hat{\theta}, \Delta p_d) - n_i^*(\mathbf{p}, \hat{\theta}, 0), \Delta p_d]|} \cdot \left[\frac{\hat{l}_z}{(\hat{l}_x^2 + \hat{l}_y^2)^{1/2}} \right] \quad (7)$$

30 This only changes how the coarse light field map is computed. The refinement and computation of
 31 the light field map remain unchanged.

32 2.2 Spot Light

33 Here we introduce the method to compute a spot light. We note, however, this calculation is not the
 34 only possible method to compute a spot light, and other types of light can be achieved in a similar
 35 fashion. We denote the position of the light as $\mathbf{l} = [l_x, l_y, l_z]^\top$. For any pixel location $\mathbf{p} = [p_x, p_y]^\top$,
 36 let us define a distance p_d and angle θ as:

$$p_d = \sqrt{(p_y - l_y)^2 + (p_x - l_x)^2} \quad (8)$$

37 and

$$\tan \theta = \frac{p_y - l_y}{p_x - l_x}, \quad \sin \theta = \frac{p_y - l_y}{p_d} \quad (9)$$

38 We then define the direction of the light as a unit vector $\mathbf{d} = [d_x, d_y, d_z]^\top$, and the HotSpot angle as
 39 α . The HotSpot factor $h(p_x, p_y)$ can then be written as:

$$h(p_x, p_y) = \left(\frac{l_z}{\sqrt{p_d^2 + l_z^2}} - \cos \alpha \right) \cdot \frac{1}{1 - \cos \alpha} \quad (10)$$

40 Where $h(p_x, p_y)$ increases to one when the target pixel \mathbf{p} is on the center line casted from the light
 41 source, and it decreases to zero when \mathbf{p} is out of the HotSpot angle α . After that, if the scale of the
 42 light source (the longest range of the illumination) is indicated by the user as l_r , we may define a
 43 distance factor $d(p_x, p_y)$ as:

$$d(p_x, p_y) = \text{clip}\left(\frac{l_r}{\sqrt{p_d^2 + l_z^2}} - 1\right) \quad (11)$$

44 where $\text{clip}(\cdot)$ is a mapping to clip the output between 0 and 1. The distance factor $d(p_x, p_y)$ ensures
 45 that the illumination is positive only when the target pixel \mathbf{p} is within the longest illumination distance
 46 l_r . Finally, the spot light effect of the \mathbf{C}_i in the main paper can then be formulated as:

$$\begin{aligned} e_i(p_x, p_y) &= e_i(l_x + p_d \cos \theta, l_y + p_d \sin \theta) = e_i^*(\mathbf{l}, \theta, p_d) \\ &= \underbrace{\frac{[n_i^*(\mathbf{l}, \theta, \Delta p_d + p_d) - n_i^*(\mathbf{l}, \theta, p_d), \Delta p_d]}{|[n_i^*(\mathbf{l}, \theta, \Delta p_d + p_d) - n_i^*(\mathbf{l}, \theta, p_d), \Delta p_d]|}}_{\text{wave}} \cdot \underbrace{\left(\frac{[l_z, p_d]^\top}{|[l_z, p_d]|} \right)}_{\text{light}} \cdot \underbrace{h(p_x, p_y)}_{\text{angle}} \cdot \underbrace{d(p_x, p_y)}_{\text{distance}} \end{aligned} \quad (12)$$

47 This only changes how the coarse light field map is computed. The refinement and computation of
 48 the light field map remains unchanged.

49 3 User Study

50 The main purpose of our user study is to fairly measure the performance of various algorithms in
 51 real-life digital painting workflow. In particular, we are mainly interested in two paradigms:

- 52 • The independent performance of various algorithms.
- 53 • The possibility of these algorithms to be embedded into professional softwares.

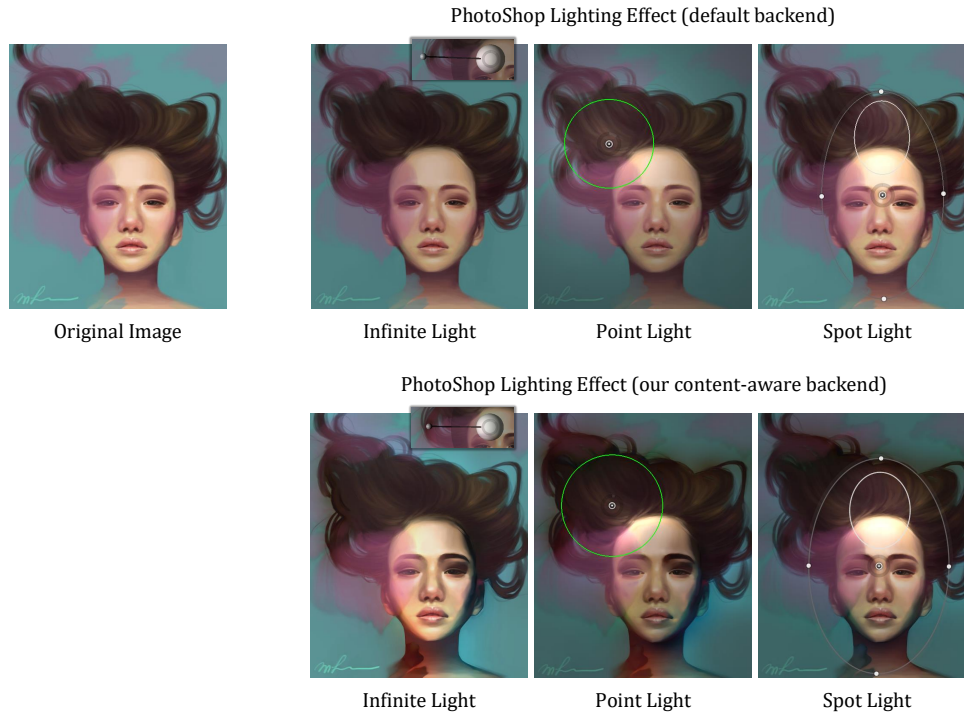


Figure 1: **Three basic types of the light sources.** We compare our lighting effect algorithm to the default algorithm in PhotoShop. Except for the position, direction, and orientation of the light sources, all other parameters in this figure, *e.g.*, the intensity and the ambient occlusion, remain constant.

54 **3.1 Candidates**

55 Here we list all algorithms and tools involved in the experiment.

- 56 • Kender and Smith [1984]. The typical filtering-based shape-from-shadow algorithm.
- 57 • Wu et al. [2008]. The typical optimization-based surface construction algorithm.
- 58 • Yu and Smith [2019]. A recent learning-based normal estimation algorithm.
- 59 • Ours. The proposed algorithm in the main paper.
- 60 • Professional Tool (PT). The lighting effect editor of PhotoShop.

61 **3.2 Rules**

62 The user study consists of a total of ten rounds, and we here introduce the rules in the first round. The
 63 remaining rounds shares the same rules.

64 **3.2.1 Setup**

65 First of all, 6 users and 1 image were randomly selected. The image is used as the lighting target in
 66 this round. The 6 users were randomly divided into two groups. In the first group, a person plays the
 67 role of an offeror. The rest five people in the second group play the role of the bidders, namely bidder
 68 ABCDE.

69 **3.2.2 Offering Stage**

70 The offeror receives the image, and then makes some indications. The indications include necessary
 71 instructions to relight the image. An example of the indications can be seen in Fig. 2. After that, the
 72 image and the indications are sent to all the bidders.

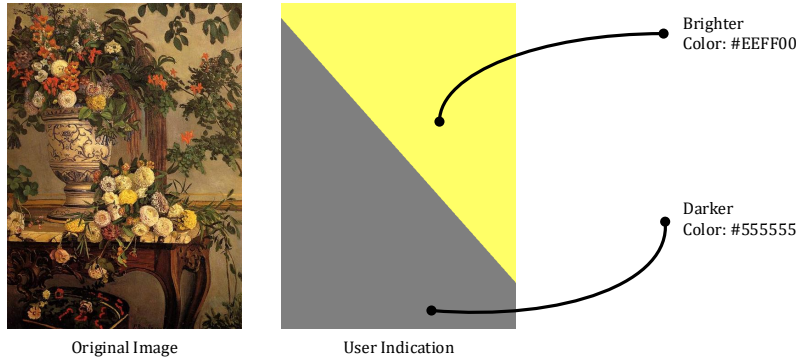


Figure 2: **Sample of user indication.** A lighting environment can be roughly separated into two parts: the bright part and the dark part. Our user indicates the position and the color of these two parts by drawing regions on a canvas. Other participants then relight the input image according to this indication.

Table 1: **The standard of the usability scores.** These scores reflect to what extent the tools are user-friendly.

Level	Score
Hard to use	-2
Baseline	0
Easy to use	+2

73 3.2.3 Single Tool Stage

74 The four tools, Wu et al. [2008]; Kender and Smith [1984]; Yu and Smith [2019] and ours, are
 75 randomly assigned to the bidder ABCD. After that, the bidder ABCD use these tools to relight the
 76 image according to the indications. They submit their results to the offeror when they are finished.
 77 The interaction time is recorded. After receiving all the submissions, the offeror ranks these results.

78 3.2.4 Tool Combination Stage

79 The bidder ABCD are then allowed to refine their results using PhotoShop lighting effect editor. The
 80 bidder E directly uses PhotoShop lighting effect editor to relight the original image. They submit
 81 their results to the offeror when they are finished. The interaction time is recorded. After receive all
 82 the submissions, the offeror is asked to rank these new results.

83 3.2.5 Feedback Stage

84 The bidder ABCD are asked whether their tool is easy to use. Their answer are translated to scores
 85 according to Table 1.

86 3.3 Results

87 3.3.1 Interaction Time

88 We report the interaction time of each round in Table 2. The statistics are also visualized in Fig. 3,
 89 where each col indicates the consumed time in the *single tool stage* and the *tool combination stage*.
 90 For PhotoShop lighting effect editor without other algorithms, we directly report the consumed time
 91 in the single stage.

92 3.3.2 User Preference

93 We report the preference ranking in Fig. 4 and Tab 3. The ranks are reported separately when different
 94 algorithms are directly used to relight images, or tool combinations are applied to relight images. We
 95 also report the ranking of the results directly achieved using PhotoShop lighting effect editor.

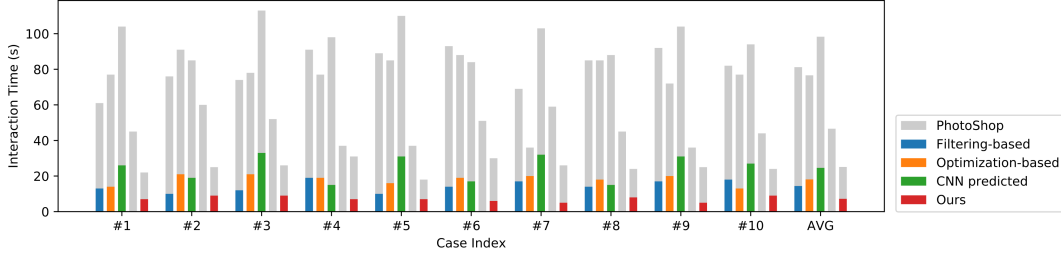


Figure 3: **Visualization of the interaction time.** The user first uses different algorithms to relight the image, and then refine their results using PhotoShop. Or they may directly use PhotoShop. The consumed time is visualized separately in the bars.

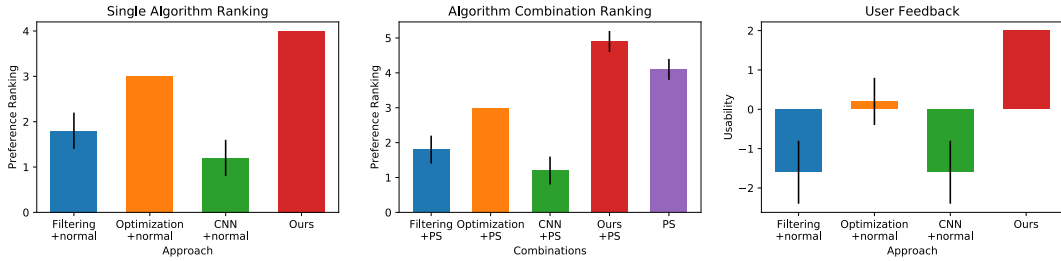


Figure 4: **Visualization of the user study results.** We report the preference ranking of single algorithm comparison, algorithm combination comparison, and the user feedback after using these tools.

96 3.3.3 Tool Usability

97 We report the user feedback in Table 4. The index of each user in each round is reported and the user's
 98 answers of the question "Whether the tool is easy to use?" are reported. The scores are translated
 99 according to Table 1.

100 References

101 John R. Kender and Earl Smith. 1984. Shape from Darkness: Deriving Surface Information from
 102 Dynamic Shadows. *National Conference on Artificial Intelligence* (1984).

Table 2: **The interaction time.** The consumed time of different user study configurations are recorded. The reported refinement time not includes the original editing time.

Case	Single Algorithm				Refined by PhotoShop				
	[1984]	[2008]	[2019]	Ours	[1984]	[2008]	[2019]	Ours	(PS only)
1	13	14	26	7	48	63	78	15	45
2	10	21	19	9	66	70	66	16	60
3	12	21	33	9	62	57	80	17	52
4	19	19	15	7	72	58	83	24	37
5	10	16	31	7	79	69	79	11	37
6	14	19	17	6	79	69	67	24	51
7	17	20	32	5	52	16	71	21	59
8	14	18	15	8	71	67	73	16	45
9	17	20	31	5	75	52	73	20	36
10	18	13	27	9	64	64	67	15	44
<i>Mean</i>	14.4	18.1	24.6	7.2	66.8	58.5	73.7	17.9	46.6
<i>Std</i>	3.07	2.7	6.99	1.47	10.07	15.25	5.75	4.01	8.31

Table 3: **The user preference ranking.** The users are asked to rank the original results of the algorithms, the results refined by the PhotoShop, and the results from PhotoShop.

Case	Single Algorithm				Refined by PhotoShop				
	[1984]	[2008]	[2019]	Ours	[1984]	[2008]	[2019]	Ours	(PS only)
1	2	3	1	4	2	3	1	5	4
2	2	3	1	4	2	3	1	4	5
3	2	3	1	4	2	3	1	5	4
4	2	3	1	4	2	3	1	5	4
5	1	3	2	4	2	3	1	5	4
6	1	3	2	4	1	3	2	5	4
7	2	3	1	4	1	3	2	5	4
8	2	3	1	4	2	3	1	5	4
9	2	3	1	4	2	3	1	5	4
10	2	3	1	4	2	3	1	5	4
<i>Mean</i>	1.8	3.0	1.2	4.0	1.8	3.0	1.2	4.9	4.1
<i>Std</i>	0.4	0.0	0.4	0.0	0.4	0.0	0.4	0.3	0.3

Table 4: **The user feedbacks.** Users are asked to give scores of their tools after using them. The scores are queried after the entire user study is finished.

Case	User Index				User Feedback			
	[1984]	[2008]	[2019]	Ours	[1984]	[2008]	[2019]	Ours
1	6	5	3	1	-2	0	-2	2
2	9	4	8	3	0	0	0	2
3	2	8	7	4	-2	0	-2	2
4	3	2	8	9	-2	0	-2	2
5	2	7	6	5	-2	0	-2	2
6	5	1	7	4	-2	2	0	2
7	8	7	1	2	0	0	-2	2
8	2	8	7	1	-2	0	-2	2
9	7	8	9	1	-2	0	-2	2
10	5	7	8	3	-2	0	-2	2
				<i>Mean</i>	-1.6	0.2	-1.6	2.0
				<i>Std</i>	0.8	0.6	0.8	0.0

103 Tai-Pang Wu, Jian Sun, Chi-Keung Tang, and Heung-Yeung Shum. 2008. Interactive normal
 104 reconstruction from a single image. *ACM Transactions on Graphics* 27, 5 (dec 2008), 1.

105 Ye Yu and William A. P. Smith. 2019. InverseRenderNet: Learning single image inverse rendering.
 106 *CVPR* (2019).