A New Shadow Removal Method Using Color-Lines

Xiaoming Yu[1,2], Ge Li[1(✉)], Zhenqiang Ying[1], and Xiaoqiang Guo[3]

[1] School of Electronic and Computer Engineering, Shenzhen Graduate School, Peking University, Shenzhen, China
geli@ece.pku.edu.cn
[2] Computer Science and Technology, Dalian University of Technology, Dalian, China
[3] Academy of Broadcasting Science, SAPPRFT, Peking, China

Abstract. In this paper, we present a novel method for single-image shadow removal. From the observation of images with shadow, we find that the pixels from the object with same material will form a line in the RGB color space as illumination changes. Besides, we find these lines do not cross with the origin due to the effect of ambient light. Thus, we establish an offset correction relationship to remove the effect of ambient light. Then we derive a linear shadow image model to perform color-line identification. With the linear model, our shadow removal method is proposed as following. First, perform color-line clustering and illumination estimation. Second, use an on-the-fly learning method to detect umbra and penumbra. Third, estimate the shadow scale by the statistics of shadow-free regions. Finally, refine the shadow scale by illumination optimization. Our method is simple and effective for producing high-quality shadow-free images and has the ability for processing scenes with rich texture types and non-uniform shadows.

Keywords: Shadow removal · Enhancement · Color line · Offset correction

1 Introduction

Shadows are prevalent in natural images and provide important cues for understanding shape of the occlusion, contact between objects and illumination position etc. But shadows will affect the performance of many computer vision tasks such as image segmentation, object recognition etc [21–24]. Thus, shadow removal is an important preprocess for these tasks. The main difficulty for shadow removal is the lack of illumination information in shadow regions. Many approaches are proposed for estimating the shadow illumination by matching regions [8,16,19,25] or shadow boundary detection [4,5,7,12,20]. Those methods do not perform well when mismatching regions or failing to recognize shadow boundary.

From observation of images with shadow, we find that pixels from an object with same material will form a color-line [13] in the RGB color space as illumination changes. Inspired by [3], which use the haze-lines to perform image dehazing,

M. Felsberg et al. (Eds.): CAIP 2017, Part II, LNCS 10425, pp. 307–319, 2017.
DOI: 10.1007/978-3-319-64698-5_26

we may use the color-lines to cluster different materials in shadow images. But unlike the haze-lines which have a uniform intersection point, the color-lines are not easy for identification since the uncertain offsets. Thus, an offset correction method is proposed to derive a linear shadow image model. Through the linear model, color-lines clustering and illumination estimation can be effectively implemented. For performing shadow removal, an on-the-fly learning approach is used to do shadow detection. Then, an initial shadow-free result is presented based on the statistics of shadow-free regions. Finally, we use the illumination optimization to refine the results.

2 Related Work

Shadow removal generally involves two subtasks: shadow detection and shadow relighting. Many approaches have been proposed including automatic methods [5,8,10,25] and user-assisted methods [1,7,16,18,19,25] by a single image, multiple images and video sequences [9,14]. A complete review of existing works is beyond the scope of this paper, we refer readers to [1,25] for excellent overviews on these methods. In this paper, we mainly focus on single-image shadow removal.

Shadow removal based on the gradient domain was suggested by Finlayson et al. [4,5]. They construct a shadow-free image based on zeroing the gradients on the shadow boundaries. The similar idea is used in [12,20] for shadow removal and modification.

Inspired by color transfer technique [15], several shadow removal methods have been proposed [16,25]. Shor and Lischinski [16] perform shadow removal by estimating an affine shadow formation model. However, this method cannot handle multi-texture shadows well since it assumes that the textures are similar in the shadow area. Thus, Xiao et al. propose a multi-scale illumination transfer technique [25] to remove multi-texture shadows. But these methods are not suitable for soft shadow cases since the penumbra regions are not considered in their model.

In addition, many approaches use the image matting to guide their shadow removal. Wu et al. [17,18] consider an image as a linear combination of a shadow-free image and a shadow matte image. Guo et al. present a region-based approach [8] for shadow detection. By using the image matting technique to get the illumination attenuation factor, they recover the shadow illumination from paired regions. Since this method does not take into account the reflectance variation, it can't recover the texture details well. Zhang et al. propose a method [25] by matching the corresponding texture patches between the shadow and lit regions. With the coherent optimization processing among the neighboring patches, they produce shadow-free results with consistent illumination. However, this method is sensitive to user initial setting when they performing image matting for illumination factor estimation.

Arbel and Hel-Or [1] use intensity surfaces and texture anchor points to remove shadows lying on curved surfaces. Gong and Cosker [7] estimate the

shadow scale of some single-material strips and then interpolate the scales in other regions using image inpainting technique. But this method may get poor results when shadow lies on the boundary of different materials since it cannot get enough strips to perform reliable interpolation.

3 Proposed Method

3.1 Shadow Image Model

Following the image formation equation in [2], an image $I(x)$ can be considered to be composed of an illumination $L(x)$ and a reflectance $R(x)$:

$$I_c(x) = L_c(x)R_c(x) \quad c \in \{\mathrm{R,G,B}\}. \tag{1}$$

The illumination $L_c(x)$ in lit region consists of the direct illumination L_c^d and the ambient illumination L_c^a. Assuming shadows in the scene are cast since the direct illumination L_c^d is blocked, the illumination can be expressed as:

$$L_c(x) = \hat{S}(x)L_c^d + L_c^a \quad \hat{S}(x) \in [0, 1], \tag{2}$$

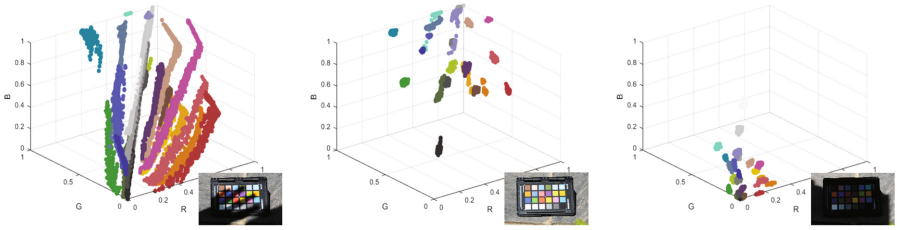
where $\hat{S}(x)$ is the attenuation factor of the direct illumination. $\hat{S}(x) = 0$ for umbra pixels, $\hat{S}(x) = 1$ for lit pixels, and others for penumbra pixels. Equation (2) is an affine function with the offset L_c^a.

Mathematically, $L_c(x)$ can be rewritten as the following affine function:

$$L_c(x) = (\hat{S}(x) + \alpha)L_c^d + (L_c^a - \alpha L_c^d) \quad \alpha \in \mathbb{R}, \tag{3}$$

where α is an offset factor. The first term $(\hat{S}(x)+\alpha)L_c^d$ describes the illumination changes in the direction of L^d, and the second term $(L_c^a - \alpha L_c^d)$ denotes the offset with a given α. Consider the reflectance of the offset in Eq. (3),

$$D_c^\alpha(x) = (L_c^a - \alpha L_c^d)R_c(x), \tag{4}$$



(a) (b) (c)

Fig. 1. (a) Shadow image and its RGB space distribution. (b) Shadow-free image and its RGB space distribution. (c) Umbra image and its RGB space distribution. (Color figure online)

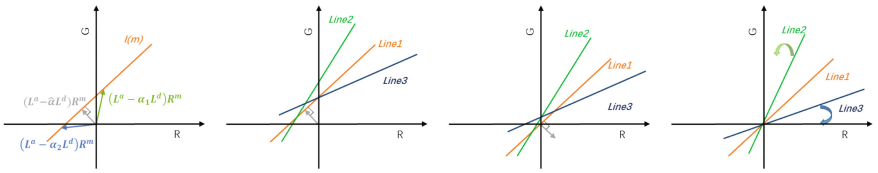


Fig. 2. (a) Color-line m with different offsets. (b) Different color-lines in RGB space. (c) Eliminate uniform offset of Line1. (d) Correct offset in different color-lines. (Color figure online)

the shadow image can be expressed based on Eqs. (1), (2) and (4):

$$I_c(x) = (1+\alpha)S(x)L_c^d R_c(x) + D_c^\alpha(x) \quad S(x) \in [\frac{\alpha}{1+\alpha}, 1], \tag{5}$$

where $S(x)$ is the new illumination attenuation factor.

From Eq. (5), there are three key observations in the RGB color space. First, if fix the $R(x)$, Eq. (5) is an affine function and pixels will form a color-line according to changes of direct light (like Fig. 1a). Second, when infinite extension of the attenuation factor ($S \in [-\infty, +\infty]$), the formation of color-lines does not intersect the origin unless the direction of L^d and L^a are identical. Third, for a given α, there is no single offset that will cause all the color-lines to intersect the origin since the color-line is reflectance dependent. Some similar conclusions are presented in [11].

From the above observations, we learn that the color-lines implies some important cues about the changes in illumination for objects of same materials. Furthermore, if the offset of different color-lines can be eliminated, we can distinguish different objects by different color-lines, and recover the shadow scale to achieve shadow removal.

3.2 Offset Correction

In [23,24], Ying et al. assume that different color-lines approximately intersect at one point in most instances. They use a uniform offset for image correction and present some interesting applications, but the global uniform offset is not reliable in the scenes with rich textures. To overcome this difficulty, we propose a robust method to estimate the offset in different materials.

First, manually select an area m_u where has similar reflectance R^{m_u} and obvious changes in illumination (Fig. 3a). Then use the principal component analysis (PCA) to find the color-line direction v of m_u and calculate the mean value of p of $I(m_u)$, The color-line can be expressed:

$$I_c(m_u) \approx t(m_u)v_c + p_c, \tag{6}$$

where $t(m_u)$ is the independent variable for the line.

As discussed above, different α will cause different offsets in Eq. (5). Intuitively, if the smallest offset is eliminated, the pixel value will change smallest and we may cause fewer errors. So we choose the smallest offset $D_c^{\hat{\alpha}}(m_u)$ which is perpendicular to the color-line (Fig. 2a):

$$D_c^{\hat{\alpha}}(m_u) = (L_c^a - \hat{\alpha} L_c^d) R_c^{m_u} \approx p_c - \frac{< p, c >}{||v||} v_c, \tag{7}$$

where $< ., . >$ is the dot product operation.

Next, we can get a rough offsets correction image $I'_c(x)$ (Fig. 2c) by eliminating uniform offset $D_c^{\hat{\alpha}}(m_u)$:

$$I'_c(x) = I_c(x) - D_c^{\hat{\alpha}}(m_u). \tag{8}$$

As we discussed earlier, a uniform offset is inaccurate. But with this rough offset correction, the offset in different color-lines is reduced to some extent, and the pixel vector direction approximates the product of the direct illumination L^d and the reflectance R. Since the L^d is globally fixed, the ration between different pixel directions is determined by the pixel reflectance. So the offset $D_c^{\hat{\alpha}}(x)$ can be estimated as:

$$D_c^{\hat{\alpha}}(x) = D_c^{\hat{\alpha}}(m_u) \frac{R_c(x)}{R_c^{m_u}} \approx D_c^{\hat{\alpha}}(m_u) \frac{||v||}{v_c} \frac{I'_c(x)}{||I'(x)||}. \tag{9}$$

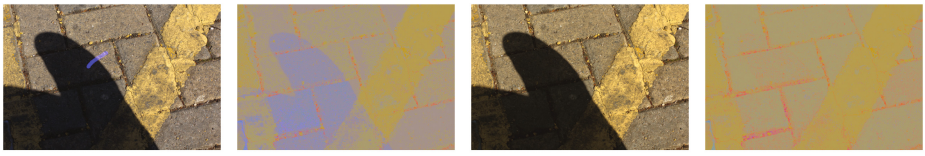
The final offset correction image $\hat{I}_c(x)$ (Figs. 2d and 3c) can be expressed as:

$$\hat{I}_c(x) = I_c(x) - D_c^{\hat{\alpha}}(x) = (1 + \hat{\alpha}) S(x) L_c^d R_c(x). \tag{10}$$

3.3 Color-Lines Clustering and Illumination Estimation

After offset correction, all the color-lines will intersect the origin when we infinite extension them. $\hat{I}_c(x)$ can be expressed in spherical coordinates as:

$$\hat{I}_c(x) = [d(x), \theta(x), \phi(x)], \tag{11}$$



(a) (b) (c) (d)

Fig. 3. (a) Shadow image and user selected region (blue line). (b) Original image clustering. (c) Offset correction image. (d) Correction image clustering. (Color figure online)

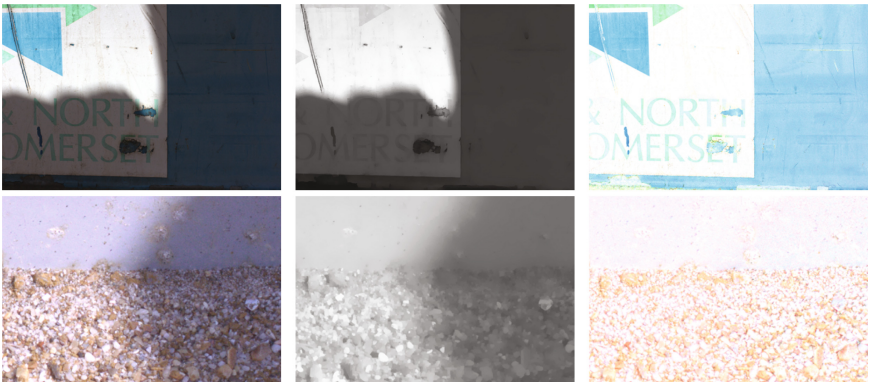


Fig. 4. First column: Shadow images. Second column: Illumination attenuation factors that estimated by color-lines. Third column: Shadow-free images by Eq. (15).

where $d(x)$ is the distance to the origin ($||\hat{I}_c(x)||$), $\theta(x)$ and $\phi(x)$ are the longitude and latitude, respectively.

For the same reflectance, changes in S affect only $d(x)$ without changing either $\theta(x)$ and $\phi(x)$. It means that we can find different color-lines by clustering the similar $[\theta(x), \phi(x)]$.

Similarly to [3], we use a uniform sampling of a sphere to build a KD-Tree for color-lines clustering. The different color clusters represent different materials (see Fig. 3d).

For a given color-line m, $d(x)$ depends on the pixel illumination:

$$d(x) = (1 + \hat{\alpha})S(x)||L^d \circ R^m||, \tag{12}$$

where $\circ$ denotes the pixel-wise multiplication.

Assume that the furthest pixel of different color-lines is shadow-free (this assumption will be relaxed in Sect. 3.4), the distance $d_{\max}$ of a color-line is:

$$d_{\max} = (1 + \hat{\alpha})||L^d \circ R||. \tag{13}$$

According to (12) and (13), the illumination attenuation factor $S(x)$ can be calculated as:

$$S(x) = \frac{d(x)}{d_{\max}(x)}. \tag{14}$$

A simple method for shadow removal is to perform global illumination uniformity:

$$I_c^{\hat{f}}(x) = \frac{\hat{I}_c(x)}{S(x)} + D_c^{\hat{\alpha}}(x), \tag{15}$$

where $I_c^{\hat{f}}(x)$ is the estimated shadow-free image. This method work well in the simple flat scenes (first row in Fig. 4). But the results may look overexposed and flat (second row in Fig. 4). In natural shadow-free scenes, the illumination will

change upon different distance between the scene and camera, or the existence of self-shadow etc. Thus, the shadow-free pixels are some compact clusters (Fig. 1b) at the top of the color-lines rather than the points. According to this fact, we refine our shadow removal method in the following sections.

3.4 Refined Shadow Removal Method

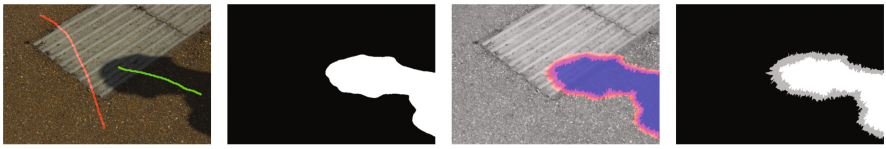
Shadow Detection. In order to perform shadow removal, we need to detect the shadows firstly based on [7].

First, two user-supplied rough inputs that indicates sample lit and shadow pixels are required to construct a KNN classifier (K = 3) in the Log-RGB color space.

Then, a label (zero and one) image will be received by classifying the pixels. Next, a spatial filtering with a Gaussian kernel is applied to the label image of posterior probability, and we can get the rough shadow mask by binarizing the filtered image using a threshold of 0.5.

After getting the mask, the sampling lines perpendicular to the shadow boundary are sampled by the gradient of Illumination factor $S(x)$. Instead of processing aligned and selected samples [7] which are time consuming, our sampling work is just to distinguish and get the rough penumbra regions as in [1,10].

Finally, the scene is classified as lit region (N), penumbra (P) and umbra (U) like Fig. 5d.



(a) (b) (c) (d)

Fig. 5. (a) Shadow image with user inputs. (b) Rough shadow mask. (c) Penumbra sampling lines (red lines) in illumination factor $S(x)$. (d) Shadow mask: white for umbra U, gray for penumbra P, and black for lit region N. (Color figure online)

Umbra Removal. Similar to the distribution of lit pixels in RGB space, the umbra pixels are a more compact cluster (Fig. 1c). We can remove the umbra by making the distribution of it to be same as that of N as following:

$$\hat{I}_c^f(U^m) = (\hat{I}_c(U^m) - E(\hat{I}_c(U^m)))\frac{SD(\hat{I}_c(N^m))}{SD(\hat{I}_c(U^m))} + E(\hat{I}_c(N^m)), \tag{16}$$

where m is the color-line index, $E(.)$ is the mean operator and $SD(.)$ is the standard deviation operator.

The operation in Eq. (16) keeps similar statistical distribution between U^m and N^m. Similar ideas are also found in [16,19] to estimate parameters of a shadow affine model. However, they ignore the penumbra. Besides, these methods require the single material in shadow regions [16] or similar material distribution between shadow and non-shadow regions [19].

Penumbra Removal. As intensity of illumination changes gradually, penumbra cannot be removed like the method used in umbra. Through the umbra-free regions $\hat{I}_c^f(U^m)$ and the illumination factor S, we relight the penumbra by pixel shadow scale estimation. First, we calculate the mean umbra scale k_c^U:

$$k_c^U = E\left(\frac{\hat{I}_c(U^m)}{\hat{I}_c^f(U^m)}\right). \tag{17}$$

Then, amplify scale k^U through the illumination factor S to estimate penumbra scale $K(P^m)$:

$$K_c(P^m) = k_c^U + (1 - k_c^U)\frac{S(P^m) - E(S(U^m))}{E(S(N^m)) - E(S(U^m))}. \tag{18}$$

In Eq. (18), we assume the scale in penumbra $K(P^m)$ is gradually changed from umbra scale k^U to non-shadow scale 1. The penumbra removal image can be express as:

$$\hat{I}_c^f(P^m) = \frac{\hat{I}_c(P^m)}{K_c(P^m)}. \tag{19}$$

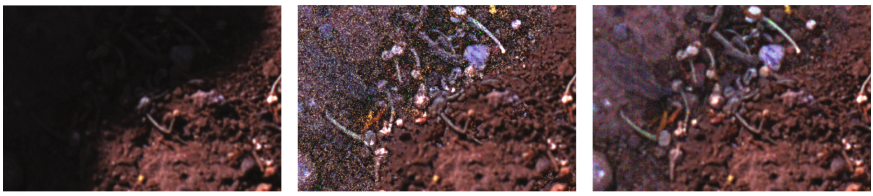
Illumination Optimization. In the above assumption, we assume that the furthest pixel in different color-lines is shadow-free, and constrain the distribution of umbra and penumbra pixels. This assumption does not hold for all the scenes (like Fig. 6b). In order to improve the robustness, we modify the previous assumption to the following:

1. Most pixels under the shadow regions corresponded to similar material pixels in the lit regions.
2. The illumination changes are locally smooth;

Assumption 1 means that we can use the proposed method to remove the shadow of most pixels. Assumption 2 implies that we can estimate the illumination in unknown shadow regions through the pixels that have been relighted.

Using the above assumptions, we refine the shadow scale $K_c(x) = \frac{\hat{I}_c(x)}{\hat{I}_c^f(x)}$ by minimizing the following energy function:

$$\underset{K_c'}{\operatorname{argmin}} \sum_x \left(\frac{K_c(x) - K_c'(x)}{\sigma^S(x)}\right)^2 + \lambda \sum_x \sum_{y \in N_x} \left(\frac{K_c'(x) - K_c'(y)}{||I(x) - I(y)||}\right)^2, \tag{20}$$



(a) (b) (c)

Fig. 6. (a) A shadow image that has many unknown materials. (b) Initial shadow-free image. (c) Optimized shadow-free image.

where $\sigma^S(x)$ is the standard deviation of $S(x)$ which is calculated per color-line, and allows us to apply our estimate only to pixels where the assumptions hold, λ is a smoothing factor for controlling the balance in the data term (left) and the smoothness term (right), and N_x denotes the set of four nearest neighbors of each pixel x in the image. In our experiments, we set $\lambda = 0.001$.

The final shadow-free image $I^f(x)$ is:

$$I_c^f(x) = \frac{\hat{I}_c(x)}{K_c'(x)} + D_c^{\hat{\alpha}}(x). \tag{21}$$

The main framework is summarized in Algorithm 1. The user scribbles include a region with obvious illumination changes (a color line) and part of the regions which indicate the scene illumination feature. Since our method focuses on illumination recovery, we simplify the operation with the user assistance which can be replaced by some automatic methods.

Algorithm 1. Shadow Removal

Input : $I(x)$, user scribbles
Output: Shadow-free image $I^f(x)$

Estimate the color-line offsets $D^{\hat{\alpha}}(x)$ and perform offset correction;
Perform Color-line clustering and illumination estimation $S(x)$;
Shadow detection;
foreach *color-line* **do**
    Umbra removal: let umbra pixels subject to similar statistical distribution with lit pixels;
    Penumbra removal: estimate the penumbra scale by umbra scale and illumination;
end
Optimize shadow scale to get shadow-free image $I^f(x)$.

Table 1. Shadow removal errors by online benchmark site [6], standard derivations are shown in brackets.

Degree	All pixels errors				Shadow pixels errors			
	Zhang et al. [25]	Guo et al. [8]	Gong and Cosker [7]	Ours	Zhang et al. [25]	Guo et al. [8]	Gong and Cosker [7]	Ours
Texturness								
Weak	0.35(0.17)	0.53(0.50)	0.26(0.16)	**0.25(0.16)**	0.16(0.20)	0.42(0.57)	**0.10(0.09)**	**0.09(0.11)**
Medium	0.39(0.25)	0.59(1.09)	0.26(0.11)	**0.24(0.10)**	0.28(0.25)	0.47(1.15)	0.12(0.09)	**0.10(0.07)**
Strong	0.58(0.38)	0.71(0.60)	0.49(0.40)	**0.36(0.25)**	0.39(0.50)	0.64(1.03)	0.36(0.44)	**0.20(0.18)**
Mean	0.44(0.27)	0.61(0.73)	0.34(0.22)	**0.29(0.17)**	0.27(0.38)	0.51(0.92)	0.19(0.21)	**0.13(0.12)**
Softness								
Weak	0.37(0.24)	0.52(1.08)	**0.23(0.10)**	**0.23(0.10)**	0.24(0.42)	0.39(1.13)	0.10(0.09)	**0.09(0.07)**
Medium	0.40(0.20)	0.70(0.36)	0.34(0.15)	**0.30(0.15)**	0.25(0.26)	0.64(0.43)	0.15(0.10)	**0.12(0.11)**
Strong	0.69(0.49)	1.09(0.75)	0.60(0.27)	**0.44(0.24)**	0.49(0.62)	1.01(0.97)	0.40(0.25)	**0.24(0.19)**
Mean	0.48(0.31)	0.77(0.73)	0.39(0.18)	**0.32(0.16)**	0.33(0.43)	0.68(0.84)	0.22(0.15)	**0.15(0.13)**
Brokenness								
Weak	0.37(0.23)	0.59(0.98)	0.25(0.13)	**0.24(0.11)**	0.24(0.40)	0.48(1.04)	0.11(0.09)	**0.10(0.08)**
Medium	0.43(0.22)	0.42(0.29)	0.29(0.14)	**0.28(0.16)**	0.27(0.27)	0.27(0.35)	0.14(0.11)	**0.12(0.11)**
Strong	1.07(0.47)	1.42(1.06)	0.69(0.30)	**0.59(0.23)**	0.88(0.72)	1.55(1.84)	0.52(0.32)	**0.39(0.17)**
Mean	0.63(0.31)	0.81(0.78)	0.41(0.19)	**0.37(0.17)**	0.46(0.46)	0.76(1.08)	0.26(0.17)	**0.20(0.12)**
Colorfulness								
Weak	0.36(0.18)	0.48(0.64)	0.24(0.11)	**0.23(0.10)**	0.21(0.24)	0.36(0.78)	0.10(0.08)	**0.09(0.07)**
Medium	0.60(0.50)	1.67(2.29)	0.48(0.18)	**0.43(0.18)**	0.57(1.06)	1.56(2.07)	0.24(0.14)	**0.22(0.15)**
Strong	0.78(0.57)	1.20(0.99)	0.56(0.31)	**0.55(0.27)**	0.72(1.00)	1.34(2.33)	0.46(0.48)	**0.40(0.30)**
Mean	0.58(0.41)	1.12(1.31)	0.43(0.20)	**0.40(0.18)**	0.50(0.77)	1.09(1.73)	0.27(0.23)	**0.24(0.17)**
Others								
	0.35(0.16)	0.38(0.52)	**0.19(0.06)**	0.20(0.07)	0.16(0.22)	0.25(0.58)	**0.06(0.02)**	0.07(0.05)

4 Experiment

4.1 Quantitative Results

We evaluate our method on the online benchmark site and dataset [6] provided by Gong and Cosker, which is for open comparison of single image shadow removal. As shown in Table 1, our method outperforms the previous methods [7,8,25] in most cases.

4.2 Visual Results

In Fig. 7, we compare our results to that of state-of-the-art methods [7,8,25]. Our result is more natural than that of the other methods in most cases. The methods of [8,25] are region-based, and reliant on image matting for estimating illumination attenuation factor, but their estimation is unreliable when the shadow regions are rich in texture. The results of [7] exist high-light artifacts since it estimates the image relighting scale by sparse sampling in the penumbra.

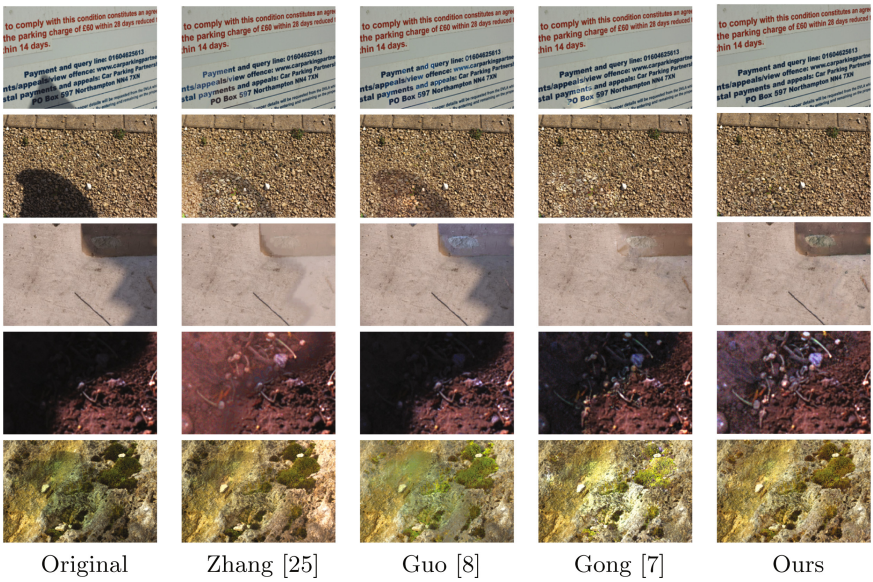


Fig. 7. Comparisons using images from dataset [6].

5 Conclusion

We present a novel single image shadow removal method based on the color-line regularity in natural images. We demonstrate that the offsets of different color-lines are different and derive a linear shadow image model by our offset

correction method. Then we use this model for color-line clustering and shadow illumination recovery. Since existing shadow removal methods only use the local regions for illumination estimation, they are sensitive to the choice of regions. By exploiting umbra and penumbra regions in detail, we propose a refine shadow removal method using color-lines for global illumination estimation and local optimization, which is more robust and effective.

Acknowledgments. This work was supported by the grant of National Science Foundation of China (No. U1611461), Shenzhen Peacock Plan (20130408-183003656), Guangdong Province Projects (2014B010117007), and Science and Technology Planning Project of Guangdong Province, China (No. 2014B090910001).

References

1. Arbel, E., Hel-Or, H.: Shadow removal using intensity surfaces and texture anchor points. IEEE Trans. Pattern Anal. Mach. Intell. **33**(6), 1202–1216 (2011)
2. Barrow, H., Tenenbaum, J.: Recovering intrinsic scene characteristics from images. In: Hanson, A., Risenman, E. (eds.) Computer Vision Systems (1978)
3. Berman, D., Avidan, S., et al.: Non-local image dehazing. In: Computer Vision and Pattern Recognition, pp. 1674–1682. IEEE (2016)
4. Finlayson, G.D., Hordley, S.D., Drew, M.S.: Removing shadows from images. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2353, pp. 823–836. Springer, Heidelberg (2002). doi:10.1007/3-540-47979-1_55
5. Finlayson, G.D., Hordley, S.D., Lu, C., Drew, M.S.: On the removal of shadows from images. IEEE Trans. Pattern Anal. Mach. Intell. **28**(1), 59–68 (2006)
6. Gong, H., Cosker, D.: Shadow removal dataset and online benchmark for variable scene categories. http://www.cs.bath.ac.uk/~hg299/shadow_eval/eval.php
7. Gong, H., Cosker, D.: Interactive removal and ground truth for difficult shadow scenes. JOSA A **33**(9), 1798–1811 (2016)
8. Guo, R., Dai, Q., Hoiem, D.: Paired regions for shadow detection and removal. IEEE Trans. Pattern Anal. Mach. Intell. **35**(12), 2956–2967 (2013)
9. Huang, J.B., Chen, C.S.: Moving cast shadow detection using physics-based features. In: Computer Vision and Pattern Recognition, pp. 2310–2317. IEEE (2009)
10. Khan, S.H., Bennamoun, M., Sohel, F., Togneri, R.: Automatic shadow detection and removal from a single image. IEEE Trans. Pattern Anal. Mach. Intell. **38**(3), 431–446 (2016)
11. Maxwell, B.A., Friedhoff, R.M., Smith, C.A.: A bi-illuminant dichromatic reflection model for understanding images. In: Computer Vision and Pattern Recognition, pp. 1–8. IEEE (2008)
12. Mohan, A., Tumblin, J., Choudhury, P.: Editing soft shadows in a digital photograph. IEEE Comput. Graph. Appl. **27**(2), 23–31 (2007)
13. Omer, I., Werman, M.: Color lines: Image specific color representation. In: Computer Vision and Pattern Recognition, vol. 2, p. II. IEEE (2004)
14. Prati, A., Mikic, I., Trivedi, M.M., Cucchiara, R.: Detecting moving shadows: algorithms and evaluation. IEEE Trans. Pattern Anal. Mach. Intell. **25**(7), 918–923 (2003)
15. Reinhard, E., Adhikhmin, M., Gooch, B., Shirley, P.: Color transfer between images. IEEE Comput. Graph. Appl. **21**(5), 34–41 (2001)

16. Shor, Y., Lischinski, D.: The shadow meets the mask: pyramid-based shadow removal. In: Computer Graphics Forum, vol. 27, pp. 577–586 (2008)
17. Wu, T.P., Tang, C.K.: A Bayesian approach for shadow extraction from a single image. In: Computer Vision-ICCV, vol. 1, pp. 480–487. IEEE (2005)
18. Wu, T.P., Tang, C.K., Brown, M.S., Shum, H.Y.: Natural shadow matting. ACM Trans. Graph. **26**(2), 8 (2007)
19. Xiao, C., She, R., Xiao, D., Ma, K.L.: Fast shadow removal using adaptive multi-scale illumination transfer. Computer Graphics Forum, vol. 32, pp. 207–218 (2013)
20. Xu, L., Qi, F., Jiang, R.: Shadow removal from a single image. In: Intelligent Systems Design and Applications, vol. 2, pp. 1049–1054. IEEE (2006)
21. Ying, Z., Li, G.: Robust lane marking detection using boundary-based inverse perspective mapping. In: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1921–1925. IEEE (2016)
22. Ying, Z., Li, G., Tan, G.: An illumination-robust approach for feature-based road detection. In: 2015 IEEE International Symposium on Multimedia (ISM), pp. 278–281. IEEE (2015)
23. Ying, Z., Li, G., Wen, S., Tan, G.: ORGB: Offset correction in RGB color space for illumination-robust image processing. In: International Conference on Acoustics, Speech and Signal Processing. IEEE (2017, in press)
24. Ying, Z., Li, G., Zang, X., Wang, R., Wang, W.: A novel shadow-free feature extractor for real-time road detection. In: Proceedings of the 2016 ACM on Multimedia Conference, pp. 611–615. ACM (2016)
25. Zhang, L., Zhang, Q., Xiao, C.: Shadow remover: image shadow removal based on illumination recovering optimization. IEEE Trans. Image Process. **24**(11), 4623–4636 (2015)