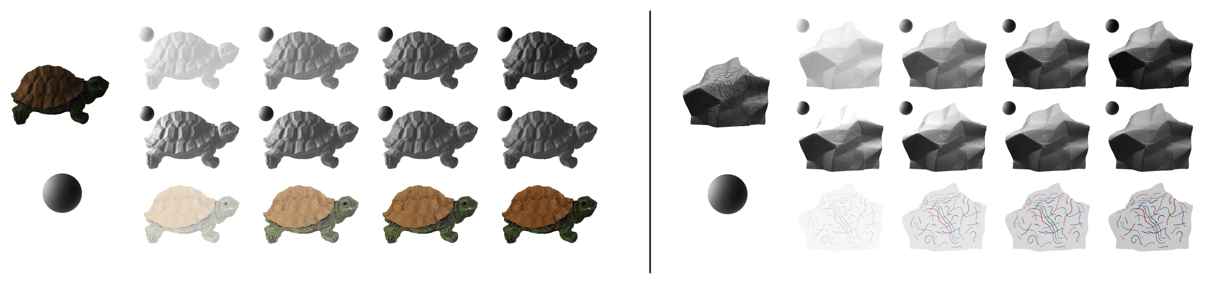
Intrinsic Image Transfer for Illumination Manipulation

Junqing Huang[1], Michael Ruzhansky[2,4], Qianying Zhang[3], Haihui Wang[1]
[1] Beihang University, [2] Ghent University, [3] Jinan University, [4] Queen Mary University of London

Introduction

Background: It is always inevitable for natural images to suffer from degradations under varying illumination conditions, in which a fine-ballanced illumination can be beneficial for many image processing and computer vision tasks.



Retinex model: An image $\mathcal{I}$ may be simply assumed to be factorized into two components: illumination $\mathcal{L}$ and reflectance $\mathcal{R}$,

$$\mathcal{I} = \mathcal{L} \odot \mathcal{R}, \tag{1}$$

where $\odot$ is a point-wise multiplication, $\mathcal{L}$ represents the illuminaiton-dependent properties such as shading, shadows or specular highlights, and $\mathcal{R}$ represents the material-dependent properties, known as the intrinsic image of a scene.

Method

Main Idea: A possible way to illumination manipulation is to control the corresponding sub-layers independently under the well-known spatial-smoothing illumination and illumination-invariant reflectance prior knowledge.

Model: Let $\boldsymbol{s} = \{s\}_i^N, \boldsymbol{c} = \{c\}_i^N \in \mathbb{R}^N$ be source and exemplar, the output $\boldsymbol{o} = \{o\}_i^N$ can be expressed as the solution,

$$\min_{o} E(\boldsymbol{o}) = \alpha E^l(\boldsymbol{o}) + \beta E^r(\boldsymbol{o}) + \gamma E^c(\boldsymbol{o}), \tag{2}$$

where E^l, E^r and E^c are photorealistic loss defined on the image illumination, reflectance and content, respectively.

Contributions:

- A generalized minimization framework— intrinsic image transfer (IIT) is designed under the well-known spatial-smoothing illumination and illumination-invariant reflectance prior knowledge.
- The photorealistic losses (illumination, reflectance, and content) are firstly defined on each sub-layer and then simplified without the necessity of taking an explicit intrinsic image decomposition.
- A closed-form solution to per-pixel image illumination manipulation is attained with favorable results on natural images having comparable or superior to the existing state-of-the-art methods.

Formulation

Illumination loss: The illumination layer is smoothing and changeable,

$$E^l(\boldsymbol{o}) = \sum_i (o_i^l - c_i^l)^2 = \sum_i \sum_{j\in\mathcal{N}_i} (\mathcal{K}_{i,j}^1 o_j - \mathcal{K}_{i,j}^2 c_j)^2, \tag{3}$$

where c is a so-called "exemplar" image, and $\mathcal{K}_{i,j}^{1(2)}$ are (Gaussian or bilateral) smoothing kernels,

$$\mathcal{K}_{i,j} \propto \exp\left(-\frac{1}{\delta_{\boldsymbol{f}}^2}\|\boldsymbol{f}_i - \boldsymbol{f}_j\|_2^2\right), \quad \sum_{j\in\mathcal{N}_i} \mathcal{K}_{i,j} = 1, \tag{4}$$

where $o_i^l = \sum_{j\in\mathcal{N}_i} \mathcal{K}_{i,j} o_j$ and $\boldsymbol{f}$ is feature vector.

Reflectance loss: A local linear embedding (LLE) is used to encode the reflectance layer s,

$$s_i^r = \sum_{j\in\Omega_i} \omega_{i,j}^{s^r} s_j^r, \quad \sum_{j\in\Omega_i} \omega_{i,j}^{s^r} = 1. \tag{5}$$

Due to the translation-invariant property of LLE algorithm, it is possible extent Eq.(5) to images, giving the loss,

$$E^r(\boldsymbol{o}) = \sum_i \Big(o_i - \sum_{j\in\Omega_i} \omega_{i,j}^o o_j\Big)^2$$
$$\text{s.t.} \quad \omega_{i,j}^o = \omega_{i,j}^s, \quad s_i = \sum_{j\in\Omega_i} \omega_{i,j}^s s_j, \tag{6}$$

Content loss: An additionally content loss $E^c(\boldsymbol{o})$ is also introduced to avoid the illumination over-fitting,

$$E^c(\boldsymbol{o}) = \sum_i (o_i - s_i)^2. \tag{7}$$

Optimization: Rewrite the losses in a matrix form:

$$E(\boldsymbol{o}) = \alpha\|\boldsymbol{K}^1\boldsymbol{o} - \boldsymbol{K}^2\boldsymbol{c}\|_2^2 + \beta\|\boldsymbol{M}\boldsymbol{o}\|_2^2 + \gamma\|\boldsymbol{o} - \boldsymbol{s}\|_2^2,$$
$$\text{s.t.} \quad \omega_{i,j}^o = \omega_{i,j}^s, \quad s_i = \sum_{j\in\Omega_i} \omega_{i,j}^s s_j, \tag{8}$$

where $\boldsymbol{K}^{1(2)}$ are kernel matries and $\boldsymbol{M} = [\boldsymbol{I} - \boldsymbol{W}]$ with identity $\boldsymbol{I}$ and $\boldsymbol{W}$ containing entries $\omega_{i,j}^o$. We solve the LLE weights,

$$\min \sum_i \Big(s_i - \sum_{j\in\Omega_i} \omega_{i,j}^s s_j\Big)^2 + \epsilon\|\boldsymbol{\omega}^s\|_2^2, \text{s.t.} \sum_{j\in\Omega_i} \omega_{i,j}^s = 1. \tag{9}$$

Once $\boldsymbol{K}$ and $\boldsymbol{W}$ are available, Eq. (9) is solved by setting $dE/d\boldsymbol{o} = 0$, giving the linear system:

$$(\alpha\boldsymbol{K}^T\boldsymbol{K} + \beta\boldsymbol{M}^T\boldsymbol{M} + \gamma\boldsymbol{I})\boldsymbol{o} = \alpha\boldsymbol{K}^T\boldsymbol{K}\boldsymbol{c} + \gamma\boldsymbol{s}, \tag{10}$$

where $\boldsymbol{L} = \alpha\boldsymbol{K}^T\boldsymbol{K} + \beta\boldsymbol{M}^T\boldsymbol{M} + \gamma\boldsymbol{I}$ is a large sparse positive matrix and Eq. (10) has a closed-form solution.

Algorithm: We summarize the proposed image illumination manipulation as follows:

Algorithm1: Intrinsic Image Transfer (IIT),

Input: Images $\{s_i\}_{i=1,\cdots,N}$, $\{c_i\}_{i=1,\cdots,N}$ and parameters α, β, γ;

1. Identifying filters: $\boldsymbol{K}^1$ and $\boldsymbol{K}^2$

1. Set parameters: $\mathcal{N}_i, \delta_s, \delta_r$ or $(\mathcal{N}_i, \delta_s)$;
2. Compute $\mathcal{K}_{i,j}$ and $\boldsymbol{K}^1$, $\boldsymbol{K}^2$ in Eq. (4);

2. Computing LLE weights: $\boldsymbol{W}$

1. Set parameters: Ω_i, ϵ;
2. Find neighbors Ω_i for each pixel i;
3. Compute $\omega_{i,j}^s$ in Eq. (9);
4. Set $\omega_{i,j} = \omega_{i,j}^s$, and $\boldsymbol{M} = \boldsymbol{I} - \boldsymbol{W}$;

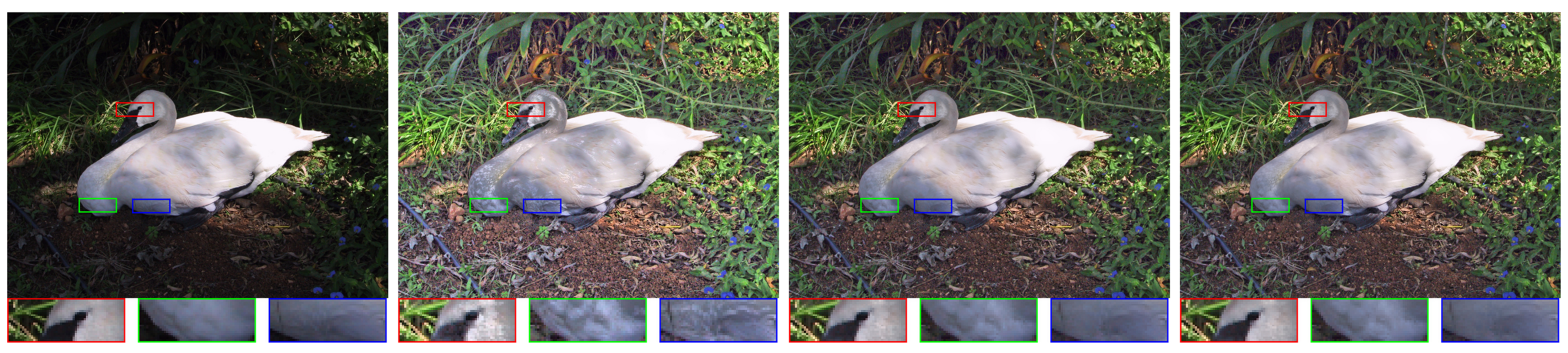
3. Reconstruction

1. Compute the *Laplacian matrix* $\boldsymbol{L}$ in Eq. (10);
2. Solve Eq. (10) with PCG algorithm;

Output: Image $\{o_i\}_{i=1,\cdots,N}$;

Experiments & Results

Gaussian filter (GF) v.s. Bilateral Filter (BF)



(a) Input (b) CLAHE exemplar (c) IIT+ GF (Ours) (d) IIT+ BF (Ours)

- Visual results of our IIT algorithm with the Gaussian and bilateral filters, respectively. The noise and distortions around the swan's "neck" and "wing" are suppressed significantly.

Natural Image Tone Mapping

(a) Input (b) Photoshop CC (c) Nasa Retinex (d) IIT+ BF (Ours)

- Visual comparison of image enhancement. The exemplars in (d) are produced by the state-of-art NASA Retinex.

Photo-realistic Style Transfer

(a) Input (b) Ours (c) Exemplar (d) Reference style

- Photorealistic style transfer. Given an image (a) and a reference style (d), a stylized exemplar (c) is provided by the deep-learning methods, which is then refined by our IIT method in (b) with more consistent textures and structures.

Verification for Exemplars

(a) Input (b) CLAHE exemplar (c) Ours (d) CLAHE exemplar (e) Ours

- Visual comparison of our IIT algorithm by using exemplars with different levels of brightness.

HDR Image Compression

(a) Log encoding (b) GD (c) Ours (d) Log encoding (e) WLS Filter (f) Ours

- Visual comparison of high dynamic range (HDR) image compression with the CLAHE exemplars.

Quantitative Evaluation:

	Cityscapes (WESPE)			NASA (Retinex)			DPED (CLAHE)		
Method	(SF / SN) TMQI	IL-NIQE	NIMA	(SF / SN) TMQI	IL-NIQE	NAMA	(SF / SN) TMQI	IL-NIQE	NIMA
NASA Retinex	(- / -) -	-	-	(0.916 / **0.731**) 0.937	20.71	**4.562**	(- / -) -	-	-
Photoshop CC	(**0.988** / 0.323) 0.887	17.37	3.859	(0.948 / 0.428) 0.892	21.65	4.003	(**0.982** / 0.507) 0.916	22.38	4.479
APE	(0.946 / 0.272) 0.840	24.25	4.002	(0.981 / 0.618) 0.937	**20.91**	3.922	(0.980 / 0.566) 0.927	21.62	4.613
Google Nik	(0.927 / 0.527) 0.906	21.32	4.131	(0.968 / 0.812) 0.965	23.15	3.822	(0.963 / 0.567) 0.925	21.53	4.523
WESPE	(0.915 / **0.839**) 0.956	20.25	**4.338**	(- / -) -	-	-	(0.931 / **0.626**) 0.928	22.25	4.534
IIT+GF (Ours)	(0.979 / 0.835) **0.971**	4.252	4.313	(0.957 / 0.650) 0.936	20.62	4.475	(0.969 / 0.587) 0.929	21.95	**4.555**
IIT+BF (Ours)	(0.981 / 0.826) 0.970	**16.48**	4.293	(**0.960** / 0.678) **0.942**	**20.57**	4.470	(0.973 / 0.589) **0.931**	**21.31**	4.540

- The exemplars in the datasets are given by the WESPE, Retinex and CLAHE methods, respectively.

Robustness to Exemplars

- Illumination compensation on Yale face dataset. From left to right: (a) Input, (b) Exemplar (noisy), (c) Ours, (d) Exemplar (distortion), and (e) Ours.

Project Webpage:
Code & Dataset & Model