High-Dynamic-Range Parallel Multi-Scale Retinex Enhancement with Spatially-Adaptive Prior

Soo-Chang Pei
Chih-Tsung Shen

National Taiwan University

Outlines

- Motivation
  - Synthesize the input to be a HDR(-like) image
- Our Proposed System
  - Single-Scale Illumination Estimator
  - Multi-Scale Illumination Estimator with Spatially-Adaptive Prior
  - Illumination/Reflectance Tuning
- Experimental Results
- Conclusions

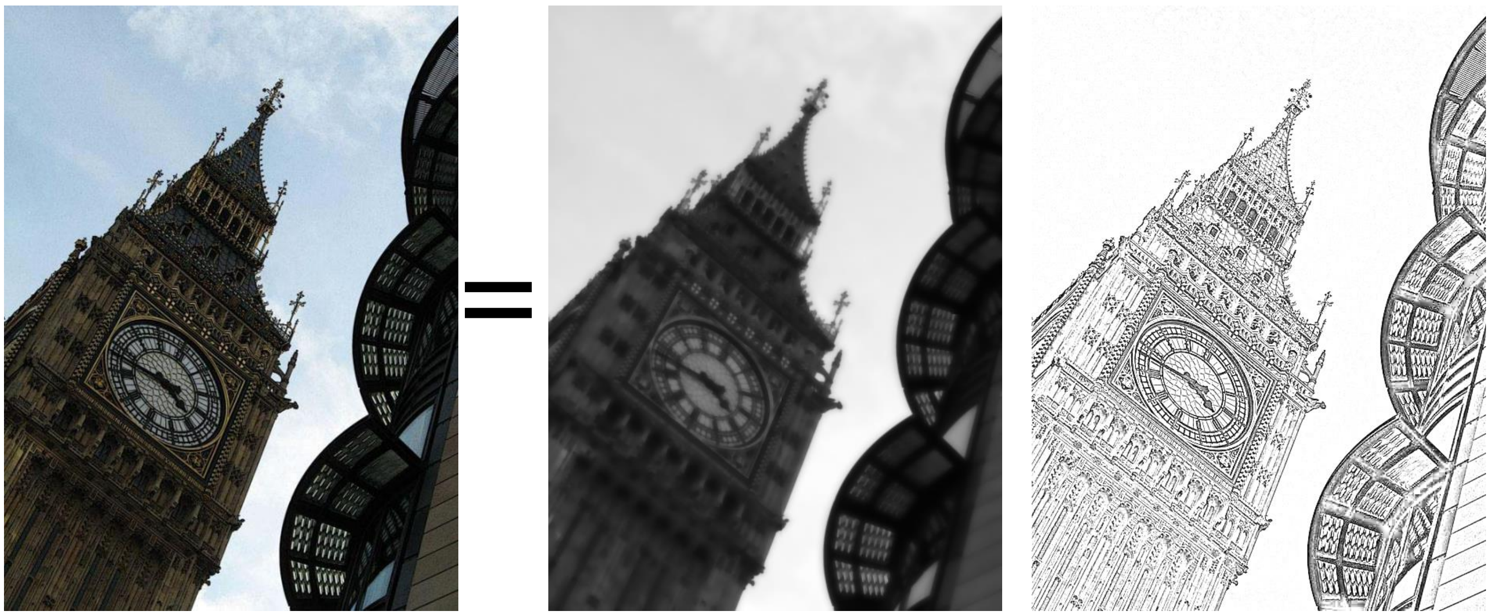
Motivation:

Original

Retinex

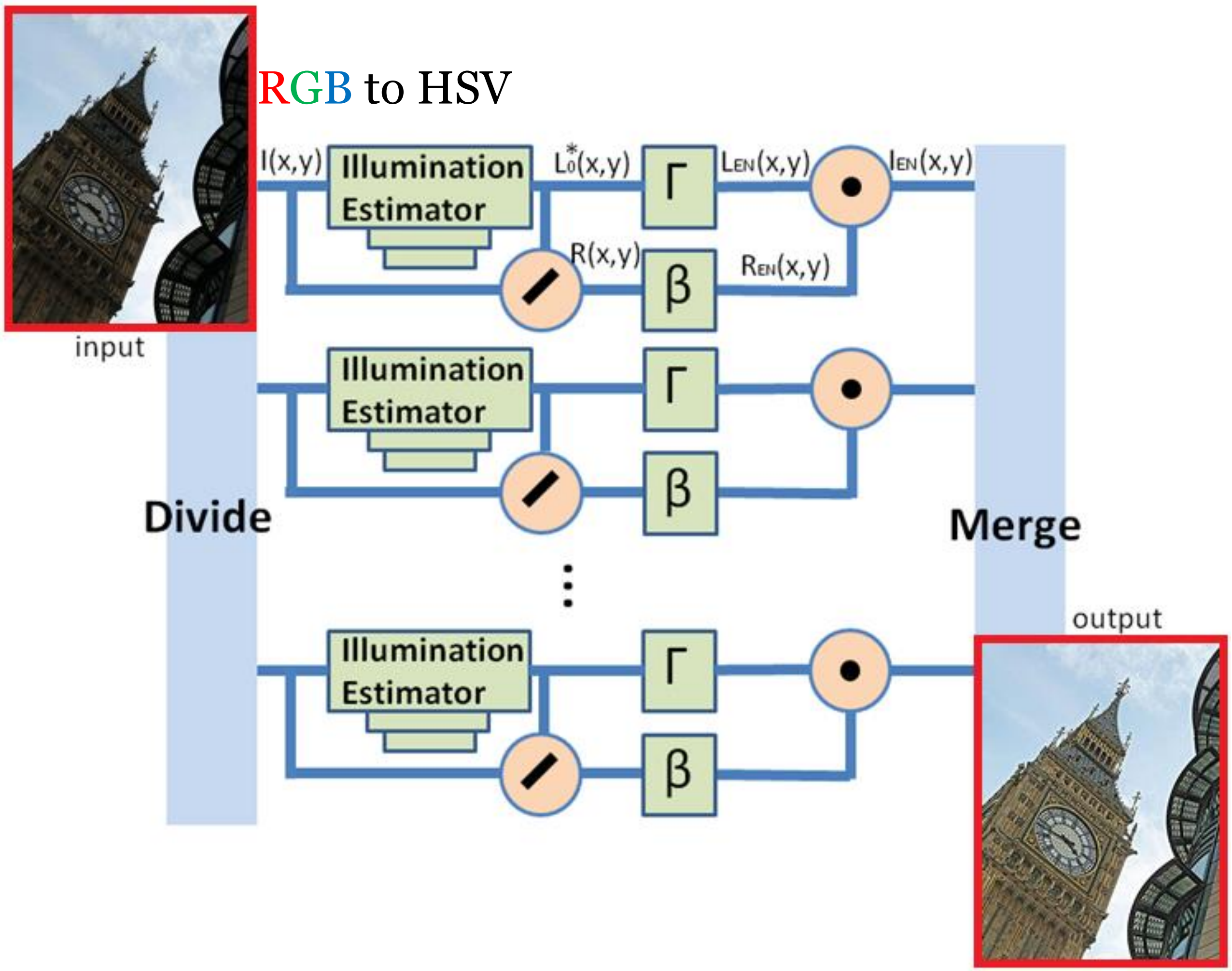
Our Synthesized HDR Result

Illumination/Reflectance Model



- intensity = illumination · reflectance
- $I(x,y) = L(x,y) \cdot R(x,y)$
- $0 \leq R(x,y) \leq 1$
- $L(x,y) \geq I(x,y)$

Our proposed system



Illumination Estimation

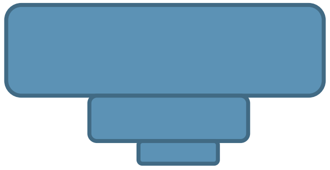
- The illumination should be piece-wise smooth.
- We can obtain the illumination by **minimization**:

$$F(L(x,y)) = \int_{\Omega} \|\nabla L(x,y)\|^2 + \alpha \cdot \|L(x,y) - I(x,y)\|^2$$

- We can extend it to multi-scale cases and make it spatially-adaptive to the content:

$$F(L_{-n}(x,y)) = \int_{\Omega} \|\nabla L_{-n}(x,y)\|^2 + \underline{w_{-n}(x,y)} \cdot \|L_{-n}(x,y) - I_{-n}(x,y)\|^2$$

such that $L_{-n}(x,y) \geq I_{-n}(x,y)$



The Pyramid

Illumination Estimation

- We solve the illumination by using gradient-descend algorithm.

$$L_{-n}(x,y) \leftarrow L_{-n}(x,y) - \mu \cdot G_{-n}(x,y)$$

$$G_{-n}(x,y) = -\Delta L_{-n}(x,y) + w_{-n}(x,y) \cdot (L_{-n}(x,y) - I_{-n}(x,y))$$

- To avoid the Halo artifact, we force the weighting function to related to the intensity signal: (L(x,y) is hard to calculate in the cost-function, so we use I(x,y).)

$$w_{-n}(x,y) = w(\nabla I_{-n}(x,y))$$

$$\approx \alpha \cdot D_{-n}\{|I(x,y) \otimes H_x(x,y)| + |I(x,y) \otimes H_y(x,y)|\}$$

Illumination Estimation

- We can easily choose H;however, in order to detect strong edges:

$$H_x = \begin{pmatrix} 1 & 1 & 1 & 0 & -1 & -1 & -1 \\ 1 & 2 & 2 & 0 & -2 & -2 & -1 \\ 1 & 2 & 3 & 0 & -3 & -2 & -1 \\ 1 & 2 & 3 & 0 & -3 & -2 & -1 \\ 1 & 2 & 3 & 0 & -3 & -2 & -1 \\ 1 & 2 & 2 & 0 & -2 & -2 & -1 \\ 1 & 1 & 1 & 0 & -1 & -1 & -1 \end{pmatrix}$$

w(x,y)

- We upsample the resulting illumination after calculating the operation inside a given scale:

$$L_{-n+1}(x,y) = U\{L^*_{-n}(x,y)\}$$

Synthesize HDR by Illumination/Reflectance Tuning

- Since we obtain the final illumination L_0*(x,y), we can obtain the reflectance:

$$R(x,y) = I(x,y)/L_0^*(x,y)$$

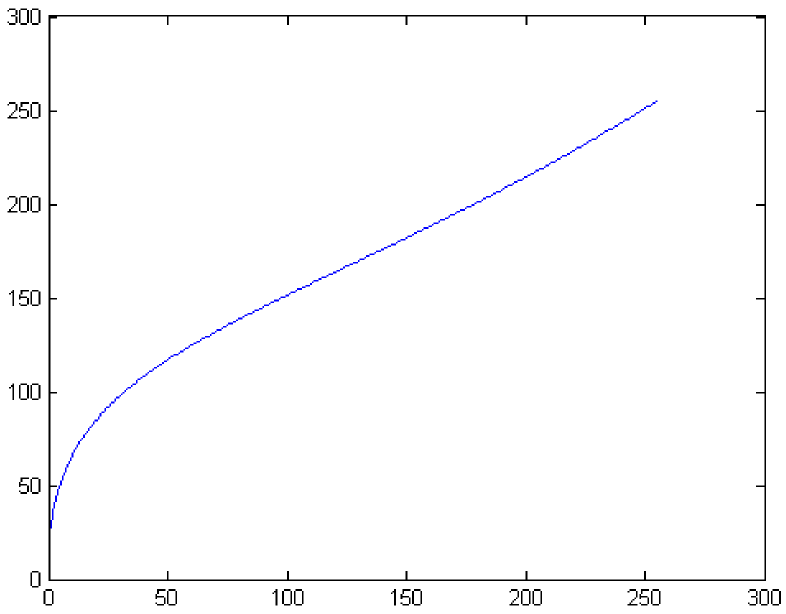
- We can tune up the illumination and reflectance to synthesize a HDR image:

$$L_{EN}(x,y) = \Gamma(L_0^*(x,y)) = W \cdot (\frac{L_0^*(x,y)}{W})^{k \cdot (1+\frac{L_0^*(x,y)}{W})}$$

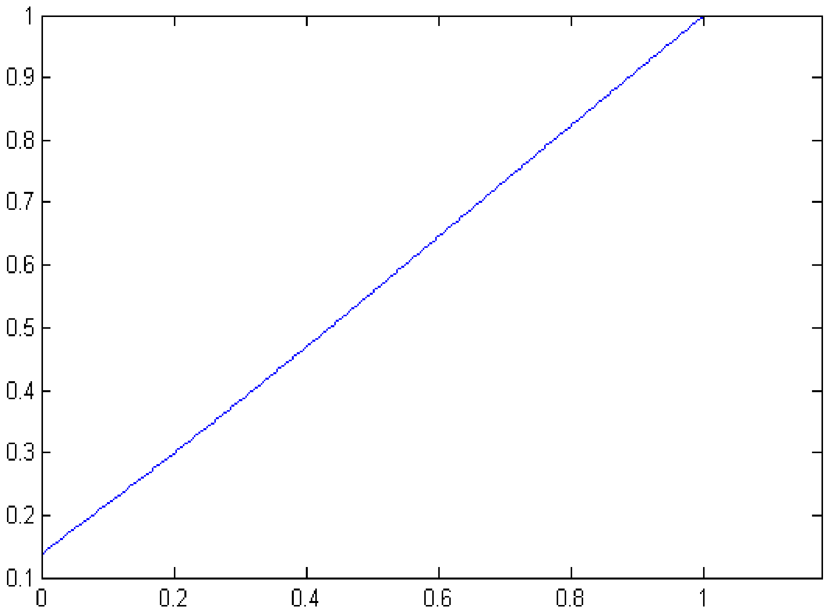
$$R_{EN}(x,y) = \beta(R(x,y)) = exp(g \cdot (\frac{1}{1+e^{-b \cdot log(R(x,y))}} - \frac{1}{2}))$$

W is the white value; k=0.4, b=2, and g=4.

Synthesize HDR by Illumination/Reflectance Tuning



$$\Gamma(L_0^*(x,y))$$



$$\beta(R(x,y))$$

Synthesize HDR by Illumination/Reflectance Tuning

- The final HDR image can be obtained:

$$I_{EN}(x,y) = L_{EN}(x,y) \cdot R_{EN}(x,y)$$

- Why Multi-Scale? Why Parallel?
  - The operation is proportional to the area of image. Multi-scale approach can save the computational power and time.
  - Our proposed system is designed to suitable for parallel computing, e.g. Multi-core, GPU.

Experimental Results

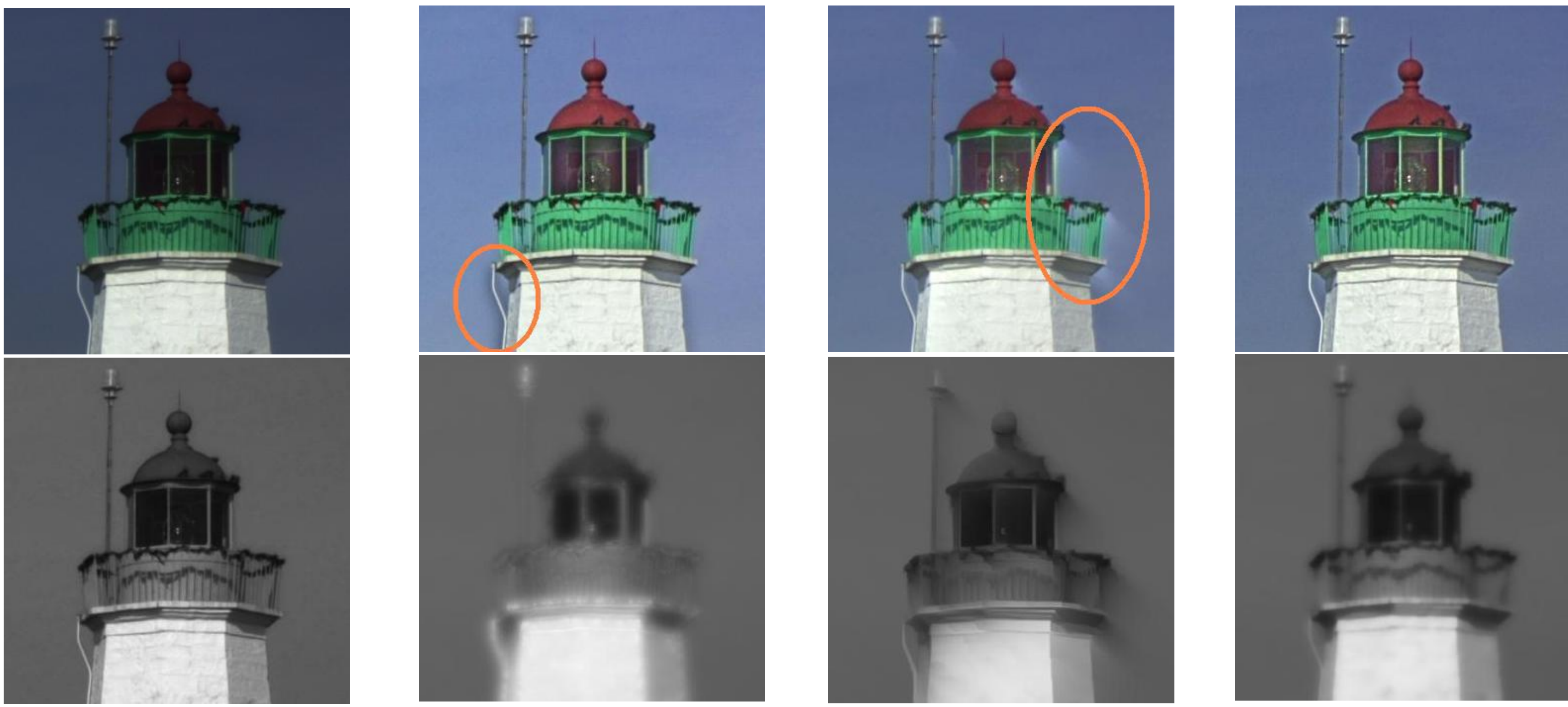
Original

Kimmel et al.

Saponara et al.

Ours

Illumination images against Halo



Original | Kimmel et al. | Saponara et al. | Our Results

Experimental Results

Original

Melyan et al.

Saponara et al.

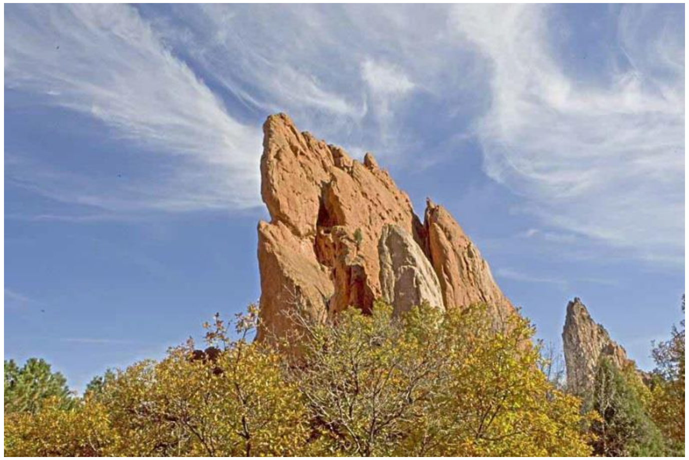
Ours

Experimental Results

Original

Melyan et al.

Shan et al.



Ours

Illumination Estimation:
Size=1000x1600.Matlab Simulation(4 Cores,3 Scales/Core,50 iters/Scale)=1.2766sec

Illumination Estimation:
Size=1200x1600.Matlab Simulation(4 Cores,3 Scales/Core,50 iters/Scale)=1.6675sec

Illumination Estimation:
Size=331x550.Matlab Simulation(4 Cores,3 Scales/Core,50 iters/Scale)=0.2005sec

Illumination Estimation:
Size=374x550.Matlab Simulation(4 Cores,3 Scales/Core,50 iters/Scale)=0.2071 sec

Illumination Estimation:
Size=302x510.Matlab Simulation(4 Cores,3 Scales/Core,50 iters/Scale)=0.1941 sec

Illumination Estimation:
Size=1200x1920.Matlab Simulation(4 Cores,3 Scales/Core,50 iters/Scale)=1.7492 sec

Illumination Estimation:
Size=768x1024. Matlab Simulation(4 Cores,3 Scales/Core,50 iters/Scale)=0.6593 sec

Illumination Estimation:
Size=897x1600. Matlab Simulation(4 Cores,3 Scales/Core,50 iters/Scale)=1.199 sec

Conclusions

- We propose a system to synthesize LR image into a HDR(-like) output.
- We estimate the illumination using a spatially-adaptive cost function.
- Our proposed system is suitable for multi-core processor, and GPU. Moreover, we speed up the process by using multi-scale approach.