Low-Light Image Enhancement

Project report for IITB EE610 Image Processing 2021

Mitali Sunil Meratwal
Dept. of Electrical Engineering
IIT Bombay
190070033
190070033@iitb.ac.in

Suraj Sarvesha Samaga
Dept. of Electrical Engineering
IIT Bombay
190020114
suraj.samaga@iitb.ac.in

***Abstract*—Images taken in low light conditions lead to pixel values with a narrow dynamic range and contrast and hence, low visibility. We implement 3 different methods: (1) based on retines theory and DT-CWT, (2) Low-Light Image Enhancement via Illumination Map Estimation (LIME) algorithm and (3) a patch-wise pixel predicting machine learning model. We then compare and contrast the results for all these 3 methods. We observe that LIME produces the clearest images with reasonable computational load and runtime.**

I. Introduction

Various factors such as low-light environment, limited performance of photography equipment lead to insufficient lighting in image capturing which significantly degrades image clarity. Due to the low dynamic range and contrast in the pixel intensities, the entire image appears very dark and it is difficult to identify objects or minute details clearly. These lost details not only cause subjective unpleasantness, but also hurt the performance of many computer vision algorithms that are primarily designed for normal-light images. A case in point would be on-board object detection applications. Images taken on-field may suffer from imperfect lighting, and detecting objects from within such an image would be a very difficult task for color-thresholding based algorithms and ML-based methods due to the changed dynamic range and pixel intensities.

The most intuitive way to go about enhancement would be direct amplification of pixel intensities, but this could lead to saturation in medium-bright regions of the image leading to loss of details. Histogram Equalization could be applied to the amplified output, however such images usually suffer from low contrast and loss in detail again.

This calls for more complex approaches. We explore two approaches based on Retinex Theory, wherein the underlying assumption is that a color image can be decomposed into two factors, reflectance and illumination, and is known to give promising results for such problems. Since Deep Learning, especially CNNs are the holy grail for image processing tasks, and we also have a size-able dataset of low-light images, we also explore a machine learning based approach of predicting the central pixel of the clean image based on a $w \times w$ neighbourhood of the pixel in low-light image.

II. Background and Prior Work

LIME (Low-Light Image Enhancement via Illumination Map Estimation) is based on Retinex theory, which aims to enhance the low light image by estimating its illumination map. Retinex as proposed in [1] asserts the color constancy experienced by humans and explains how we are able to sense colors consistently inspite of differences in light levels. For example, the color of a lemon is perceived as yellow irrespective of the illumination suggesting that the eye does not respond only to the flux at a given unit area, but comparision of flux at a given point with the flux over the entire field of view on three systems. Therefore, according to retinex theory the color image can be decomposed into reflectance and illumination.

The approach based on retinex and DT-CWT [2] applies a Dual-Tree Complex Wavelet Transform (DT-CWT) on the input image. In a 1-D DT-CWT, the input signal is fed to two separate DWT 'trees' operating in parallel with different real filter banks representing the real and imaginary parts of the complex wavelet. For a 2-D image input, 1-D DTCWT is performed on image rows, and is then performed on the image columns. The output is 2 low-pass sub-images and 6 high-pass sub-bands, both of which are further filtered separately before reconstruction. As mentioned in [2], wavelet theory has low entropy and multiresolution features hence noise reduction can be achieved by processing wavelet coefficients.

Some of the prior work based on Retinex also inlcude single scale resolution (SSR) which computes the reflectivity image for each color channel and treats this as the final enhanced result. A modification to SSR to improve color reproducibility is multi scale resolution takes weighted average of SSR over different gaussian surrounding function but this reduces dynamic range compression with the reproduced image appearing unnatural and over enhanced. Another method [3] is based on dehazing inverted low light images and inverting the dehazed image again. Many deep learning techniques and architectures have been proposed recently [4] based on decomposition into reflectance and illumination images, adjustment using an encoder-decoder to brighten up the illumination and finally reconstruction of adjusted illumination and reflectance to get the enhanced image. By estimating only illumination, the authors reduce the solution space and complexity significantly without compromising on quality of results. The authors also aim at devising a method that not only produces enhanced images at par with state of the art but also has physical interpretation (1). Augemented Lagrangian Multiplier (ALM) based method is used to solve the constrained optimization problem by replacing the original problem with sequence of unconstrained problem till convergence. Block-matching and 3D filtering (BM3D) is a 3D block-matching technique used for denoising images. BM3D is a cascade of hard thresholding and wiener filtering. Each stage involves grouping, collaborative filtering and aggregation.

III. Data and Methodology

The ML model has been trained on the training set of the LOL dataset [5] comprising of 485 low-light images. All the methods are tested using the test set of the LOL dataset consisting of 15 images.

A. LIME

The idea is to enhance a low-light image by estimating its illumination map. Denoting the captured and recovered image by $\mathbf{L}$ and $\mathbf{R}$ respectively, retinex theory tells us that

$$\mathbf{L} = \mathbf{R} \circ \mathbf{T} \tag{1}$$

where $\mathbf{T}$ is the illumination map and the operator $\circ$ refers to element-wise multiplication. Hence our low-light enhancement problem reduces to estimating $\mathbf{T}$ accurately. Heuristically, illumination is at least the maximal value of three channels at a certain location. Hence as an initial guess we estimate the illumination map as

$$\mathbf{T}(\mathbf{x}) \leftarrow \max_{c \in R,G,B} \mathbf{L^c}(\mathbf{x}) \tag{2}$$

for each pixel x of the image.

A good solution to (1) must preserve the overall structure and smooth the textural details. For this purpose, we must solve an objective function that has a term that takes care of the difference between the refined map and the initial estimate, and another term to take care of the smoothness. Once such function would be

$$\min_{\mathbf{T}} \|\hat{\mathbf{T}} - \mathbf{T}\|_F^2 + \alpha \|\mathbf{W} \circ \nabla \mathbf{T}\|_1 \tag{3}$$

Where $\mathbf{T}$ and $\hat{\mathbf{T}}$ are the refined and initial illumination maps, $\|\cdot\|_F$ and $\|\cdot\|_1$ are the Frobenius and l_1 norms respectively. Moreover, $\mathbf{W}$ is the weight matrix, $\nabla\mathbf{T}$ is the first order derivative filter and α is a coefficient to balance both terms.

For ease of solving and separability in (3), we denote $\nabla\mathbf{T}$ by $\mathbf{G}$. The Augmented Lagrangian Multiplier method is employed to solve (3). The Augmented Lagrangian of (3) can be written as

$$\mathcal{L} = \|\hat{\mathbf{T}} - \mathbf{T}\|_F^2 + \alpha \|\mathbf{W} \circ \nabla \mathbf{T}\|_1 + \Phi(\mathbf{Z}, \nabla\mathbf{T} - \mathbf{G}) \tag{4}$$

where $\Phi(\mathbf{Z}, \nabla\mathbf{T} - \mathbf{G}) = \frac{\mu}{2}\|\nabla\mathbf{T} - G\|_F^2 + \langle \mathbf{Z}, \nabla\mathbf{T} - \mathbf{G}\rangle$, $\mathbf{Z}$ is the Lagrangian multiplier and μ is a positive penalty scalar. To solve (4), we iteratively update one variable at a time by fixing the others. Conveniently, each step has a simple closed-form solution, given by

$$\mathbf{T}^{(t+1)} \leftarrow \mathcal{F}^{-1}\left(\frac{\mathcal{F}(2\hat{\mathbf{T}} + \mu^{(t)}\mathbf{D^T}(\mathbf{G}^{(t)} - \frac{\mathbf{Z}^{(t)}}{\mu^{(t)}}))}{2 + \mu^{(t)}\sum_{d \in h,v} \overline{\mathcal{F}(\mathbf{D}_d)} \circ \mathcal{F}(\mathbf{D}_d)}\right) \tag{5}$$

$$\mathbf{G}^{(t+1)} = \mathcal{S}_{\frac{\alpha\mathbf{W}}{\mu^{(t)}}}\left[\nabla\mathbf{T}^{(t+1)} + \frac{\mathbf{Z}^{(t)}}{\mu^{(t)}}\right] \tag{6}$$

where $\mathcal{S}_\epsilon[x] = sgn(x)\max(|x| - \epsilon, 0)$ and is applied element-wise in (6)

$$\begin{aligned} \mathbf{Z}^{(t+1)} \leftarrow \mathbf{Z}^{(t)} + \mu^{(t)}(\nabla\mathbf{T}^{(t+1)} - \mathbf{G}^{(t+1)}); \\ \mu^{(t+1)} \leftarrow \mu^{(t)}\rho, \rho > 1 \end{aligned} \tag{7}$$

(5), (6) and (7) are used to implement the exact solver in code.

For refinement on the initial illumination map, we use the gradient of the initial illumination map to initialize the weight matrix W as

$$\mathbf{W}_h(x) \leftarrow \frac{1}{|\nabla_h \hat{\mathbf{T}}(x)| + \epsilon}; \mathbf{W}_v(x) \leftarrow \frac{1}{|\nabla_v \hat{\mathbf{T}}(x)| + \epsilon} \tag{8}$$

Once we have the refined illumination map, we apply gamma correction on it via $\mathbf{T} \leftarrow \mathbf{T}^\gamma$ and obtain the refined image from (1). Image quality can be further enhanced by applying denoising techniques. We apply the BM3D algorithm on only the Y channel of the refined image $\mathbf{R}$ to get the denoised image $\mathbf{R}_d$. Since noise can be different at different patches of the input, to avoid imbalanced processing we reconstruct the final image $\mathbf{R}_f$ as

$$\mathbf{R}_f \leftarrow \mathbf{R} \circ \mathbf{T} + \mathbf{R}_d \circ (\mathbf{1} - \mathbf{T}) \tag{9}$$

B. DT-CWT

Following the methodology adopted in [2], the entire image processing is on the V-channel of the image, as explained in Fig. 1. DT-CWT is then applied on it to yield two low-pass sub-images and six high-pass sub-bands, after which different contrast enhancement techniques are used for the low and high pass subbands.

1) Filtering for Low Pass Sub-bands: In order to compress the global dynamic range of the image, we use the function as given in [2]

$$L_g(x,y) = \frac{\log\left(\frac{L_w(x,y)}{\overline{L_w}} + 1\right)}{\log\left(\frac{L_{wmax}}{\overline{L_w}} + 1\right)} \tag{10}$$

where $L_g(x,y)$ is the global luminance output, $L_w(x,y)$ is the luminance value of the input image, L_{wmax} is the maximum luminance value of input, and $\overline{L_w}$ is defined as

$$\overline{L_w} = exp(\frac{1}{N}\sum_{xy} \log(\delta + L_w(x,y)) \tag{11}$$

where N is the total number of pixels in an image and δ is a constant to avoid the occurrence of singularities.

A local adaptive algorithm based on Retinex theory is applied as follows

$$L_{out}(x,y) = a(x,y)\log\left(\frac{L_g(x,y)}{H_g(x,y)} + \beta\right) \tag{12}$$

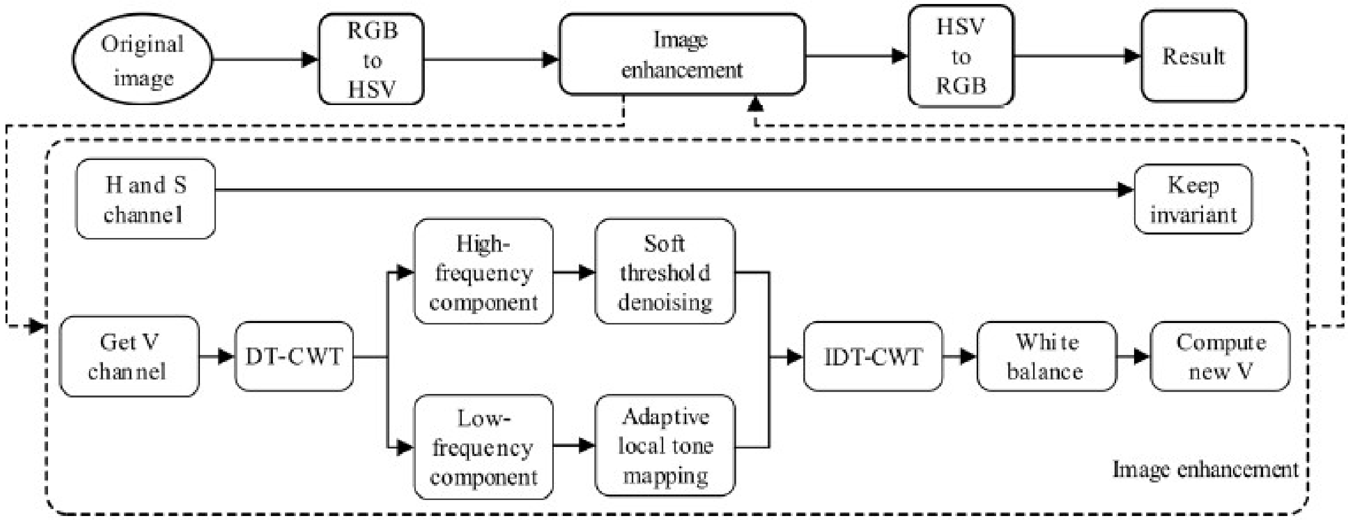


Fig. 1: DTCWT workflow [2]

Where $\alpha(x, y) = 1 + \eta \frac{L_g(x,y)}{L_{g,max}}$ and $\beta = \lambda \overline{L_g}$ where $\eta = 36$ and $\lambda = 10$. $H_g(x, y)$ is the output of the $L_g(x, y)$ applied to the guided filter and is given by

$$H_g(x, y) = \frac{1}{|\omega|} \sum_{\epsilon_x, \epsilon_y \in \omega(x,y)} [a(\epsilon_x, \epsilon_y) L_g(x, y) + b(\epsilon_x, \epsilon_y)] \tag{13}$$

where ϵ_x, ϵ_y are coordinates of neighborhood pixels, $\omega(x, y)$ is a square window with radius r and center (x, y). $a(\epsilon_x, \epsilon_y)$ and $b(\epsilon_x, \epsilon_y)$ are linear coefficients, given by

$$a(\epsilon_x, \epsilon_y) = \frac{\mu_2(\epsilon_x, \epsilon_y) - \mu^2(\epsilon_x, \epsilon_y)}{\sigma^2(\epsilon_x, \epsilon_y) + \tau} \tag{14}$$

$$b(\epsilon_x, \epsilon_y) = \mu(\epsilon_x, \epsilon_y) - a(\epsilon_x, \epsilon_y)\mu(\epsilon_x, \epsilon_y) \tag{15}$$

where $\mu(\epsilon_x, \epsilon_y)$ and $\sigma^2(\epsilon_x, \epsilon_y)$ are the mean and variance of neighbourhood in L_g, $\mu_2(\epsilon_x, \epsilon_y)$ is the mean of L_g^2 in $\omega(x, y)$ and τ is the regularization parameter, set as 0.01.

2) Filtering for High Pass Sub-bands: Since the high-pass coefficients contain most of the noise along with other sharp details, soft-thresholding is applied to retain smoothness and improve distortion, expressed as follows:

$$Y = \begin{cases} sgn(X)(|X| - T) & |X| > T \\ 0 & |X| \leq T \end{cases} \tag{16}$$

where Y is the high-frequency coefficient after thresholding, X is the high-frequency coefficient after DTCWT, T is the threshold value.

Finally for contrast enhancement, the 'Whitebalance' [6] operation is applied on the V-channel, which is done by picking new maximum and minimum quantiles from the pixel intensity histogram, saturating the pixels outside the new range and finally rescaling the image back via an affine transform. (12) and (16) are used in code along with Whitebalance as described in [6]

C. ML

Fifty 13x13 patches are randomly extracted from the entire training dataset. A central patch of smaller window size $w \times w$ (eg. 3×3, 5×5, 7×7 etc.) is chosen from the bigger patch which forms the input data to the convolution neural network (CNN) and the central pixel of the enhanced image is the target. The CNN architecture is shown in Fig 2 where each conv2d layer is followed by ReLU activation function and sigmoid is applied on the output of the last layer. The dataset is split into 70:30 ratio for training and validation and the model is trained for 100 epochs. The loss function used is normalised root mean square error and batch training is performed. Model with minimum loss on validation dataset is chosen for inference on test data. The results are finally evaluated on 15 test images. While testing, the image is first padded and for each $w \times w$ patch in the image, iterating through rows and columns, single pixel value is predicted by the trained CNN.

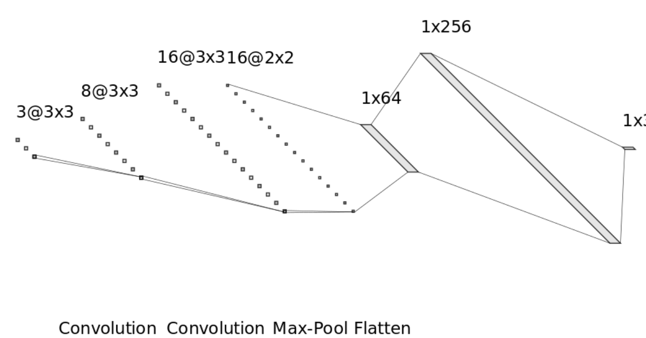


Fig. 2: CNN architecture

IV. Experiments and Results

For impelemtation of low light image enhancement based on DT-CWT [2] one level of decomposition is applied for both the trees. Though the algorithm does improve overall illumination, a lot of noise is present in the enchanced image. The image is of not satisfactory quality. Hence we also implement another algorithm LIME [7] based on illumination map estimation. This method produces very good results. In LIME, strategy II described in [7] is used for weight initialisation. α is chosen as 0.15 by default for all the experminets and ρ is found by grid search. For $\rho = 2.0$ the results are good and inference time is faster as compared to other values. Furthermore, we compare LIME against a machine learning model as well. The simple CNN trained on 3x3 image patches does not perform better than LIME as well. The model does learn to increase the global illumination of the image to some extent but also introduces high frequency noise in the predicted image which apparently is absent in original low light images. Only visual assessment of the quality of images for all the strategies described has been made and LIME performs by far the best among all others with significantly reduced computation costs (**26 seconds** while CNN takes **1 minute**). We present results of LIME, DT-CWT and CNN in Fig. 3 for images in the test folder of LOL dataset.

V. Learning, Conclusions, and Future Work

We started with low light image enhancement method based on retinex theory and DT-CWT. This method is able to improve illumination and make the objects visible but also introduces undesirable noise in the output image. This motivates us to look for other fast and robust low light image enhancement techniques. LIME is one such method that produces visually adequate and acceptable results. Implementing Low-Light Image enhancement via Illumination Map Estimation (LIME) reveals how non machine learning models can also perform efficiently and effectively without the need of large datasets for training or GPUs for computation and hence can be easily incorporated with any hand held devices or light applications. The algorithm descirbed does not assume any specific properties of image and enhances the image while denoising in case noise is present. It is no doubt this technique can be used to preprocess many vision based applications ranging from edge detection, feature matching, object recognition and tracking, night surveillance and thus improve the performance.

Contribution of Team Members

- **Mitali** Fully implemented the exact solver in LIME algorithm, wrote code for Lowpass enhancement in the DT-CWT algorithm, implemented the CNN model and collected results. Wrote sections II, IV and V of report.
- **Suraj**: Implemented all helper functions in LIME algorithm, wrote code for Highpass enhancement and other helper functions in DT-CWT. Wrote sections I and III of report.

References

[1] E. H. Land, "The retinex theory of color vision," *Scientific American*, vol. 237, no. 6, pp. 108–128, 1977.

[2] M. xiang Yang, G. jin Tang, X. hua Liu, L. qian Wang, Z. guan Cui, and S. huai Luo, "Low-light image enhancement based on retinex theory and dual-tree complex wavelet transform," *Optoelectronics Letters*, vol. 14, no. 6, pp. 470–475, 2018.

[3] C.-Y. Tsai and C.-H. Chou, "A novel simultaneous dynamic range compression and local contrast enhancement algorithm for digital video cameras," *EURASIP Journal on Image and Video Processing*, no. 1, 2011.

[4] C. Wei, W. Wang, W. Yang, and J. Liu, "Single low-light ghost-free image enhancement via deep retinex model," *KSII Transactions on Internet and Information Systems*, vol. 15, no. 5, 2021.

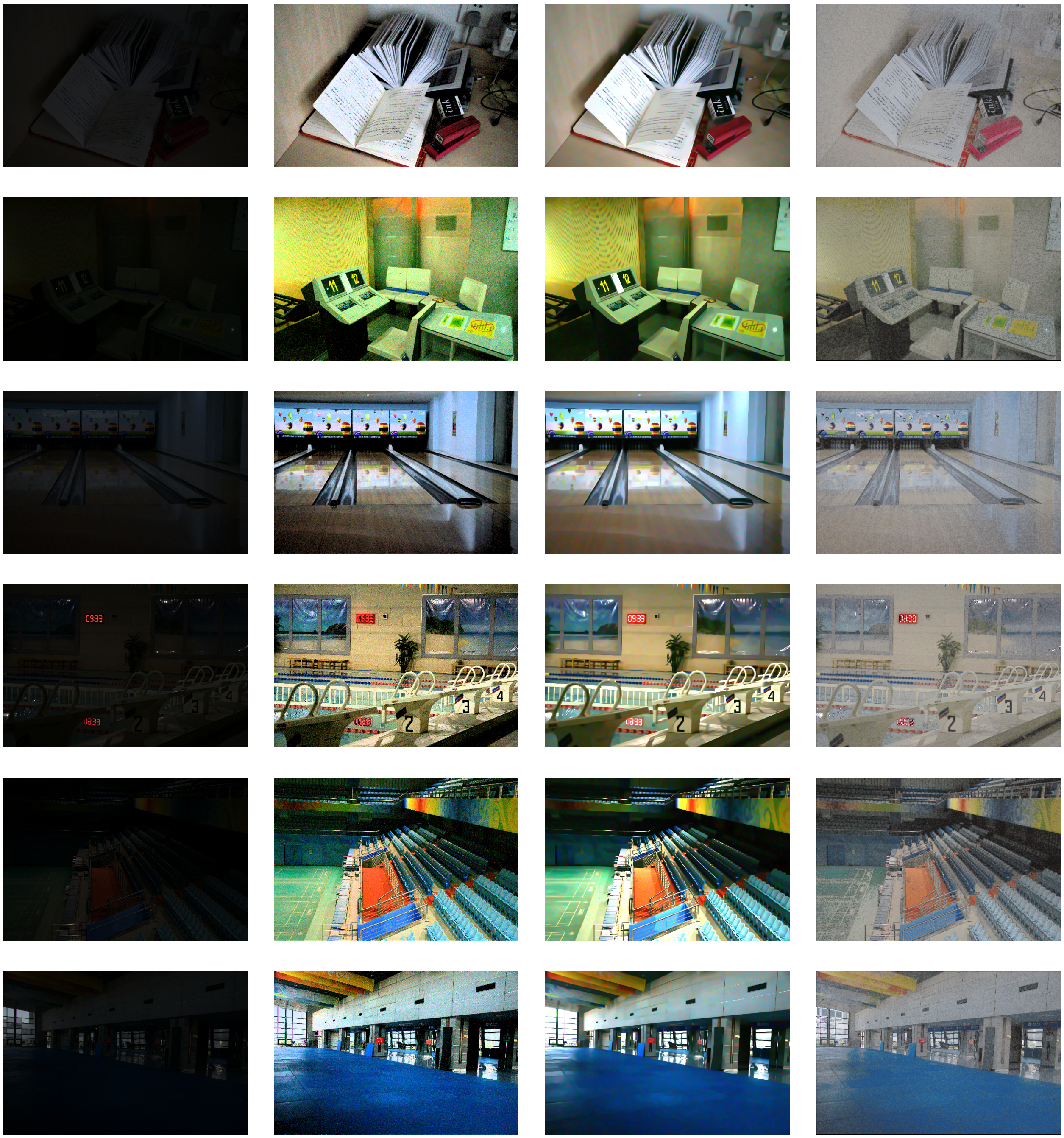
[5] (2021) Bmvc2018 deep retinex decomposition. [Online]. Available: https://daooshee.github.io/BMVC2018website/

[6] N. Limare, J.-L. Lisani, J.-M. Morel, A. B. Petro, and C. Sbert, "Simplest color balance," *Image Processing On Line*, vol. 1, pp. 297–315, 2011.

[7] X. Guo, Y. Li, and H. Ling, "Lime: Low-light image enhancement via illumination map estimation," *IEEE Transactions on Image Processing*, vol. 26, no. 2, pp. 982–993, 2017.

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

Fig. 3: Comparing different methods for low light image enhancement. (a) Low Light test images in LOL dataset. Results using (b) DT-CWT. (c) LIME. (d) CNN.



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

Fig. 3: Comparing different methods for low light image enhancement. (a) Low Light test images in LOL dataset. Results using (b) DT-CWT. (c) LIME. (d) CNN.