Local Backlight Dimming for Liquid Crystal Displays via Convolutional Neural Network

Junho Jo *, Jae Woong Soh*, Jae Sung Park †, and Nam Ik Cho *
* Dept. of Electrical and Computer Eng., INMC, Seoul National University, Seoul, Korea
† Samsung Electronics, Ltd., Suwon, Korea
E-mail: {jottue, soh90815}@ispl.snu.ac.kr; nicho@snu.ac.kr

***Abstract*—This paper presents a new local backlight dimming (LBD) for liquid crystal displays (LCD) method based on a convolutional neural network (CNN). Many previous LBD algorithms controlled the backlight intensity relying on hand-crafted features within a local block, that is, statistical information of pixel values in each block. However, they have a lack of generalization ability due to the use of hand-crafted features, which are usually not adaptive to the input properties. Also, they usually disregarded the diffusion property of the backlight that may affect the neighboring blocks. In this respect, we propose a CNN-based LBD algorithm to alleviate these problems. To address the lack of generalization ability of hand-crafted features, we adopt a CNN-based approach that learns the features and thus provides appropriate backlight intensities for the given inputs. Also, the diffusion property of light and leakage property of liquid crystal are considered when training the network, thereby alleviating the loss of details while achieving the high contrast ratio. Experiments show that the proposed method outperforms both quantitatively and qualitatively compared to the other LBD algorithms. Specifically, for the images from the DIV2K dataset, the proposed method achieves at least 1dB enhancement in PSNR, showing the generalization performance.**

I. Introduction

Currently, two types of displays, namely organic light-emitting diode (OLED) display and the liquid crystal display (LCD), are the most dominant ones in the market. The OLED display is a self-emitting device, which directly illuminates the light with the intensity of the corresponding pixel value. Hence, it can render fine color differences and a perfectly black screen, and thus achieves the infinite contrast ratio. On the contrary, the LCD is not a self-emitting device, and thus the backlight is required to display the contents. Technically, the displayed contents on LCD are derived by the product of liquid crystal (LC) transmittance and backlight luminance. However, due to the structural limitation of LC, the LCD has light leakage property, which means that LC transmittance cannot be zero. Hence, the LCD cannot make the perfect-black area, and thus the overall contrast is generally worse than the self-emitting devices. However, thanks to several advantages compared to the OLED display such as in lifetime, durability, and price, it still has a large market share, and thus there have been many efforts to overcome the above-stated drawbacks of LCD.

One of the main approaches to reduce the influence of leakage property is the backlight dimming technique. In principle, this method dims the backlight at the dark area and conversely compensates for the LC transmittance to be reciprocally proportional to the dimming ratio. As a result, the displayed image is less distorted while reducing the influence of leakage property. Nowadays, the light-emitting diode (LED) has replaced cold-cathode fluorescent lamps (CCFL) as a backlight unit (BLU). The BLU is usually divided into blocks so that we can locally and individually control the intensities of LEDs in each block according to image contents. This strategy is named as local backlight dimming (LBD), which greatly increases the contrast ratio while reducing the power consumption [1]–[6]. However, the LBD also often brings some degradations, such as loss of details and halo artifacts. First, when the backlight is significantly dimmed, the LC transmittance value is overcompensated, which results in saturation and thus loss of details. Conversely, boosting the backlight intensities usually preserves the details, but the diffusion of light and the leakage property of LC bring out halo artifacts, *i.e.*, blooming around the intensity-boosted blocks. Therefore, finding the optimal backlight intensities for the given image is essential so that the LCD can display less-distorted details and real black level.

To handle these problems, several algorithms have been proposed, addressing their uniqueness in the decision criteria of dimming ratio [1]–[6]. Many previous LBD algorithms rely on hand-crafted features on the only local block, that is, statistical information of pixel values in each block (a segment of BLU). For instances, the most straightforward method is to control the backlight intensity of each block based on the mean or maximum values of the pixels in each block. Chen *et al.* [1] used a weighted average of the histogram of each image block to calculate the initial luminous intensity. Hsia *et al.* [2] used variance values of a block to detect the region that contains edges, which needs enhancement of BLU to preserve edges. However, the aforementioned hand-crafted-feature-based algorithms [1]–[3], [6] determined the intensity of the local LED segment only with the corresponding (local) block information. Since the light of a BLU is diffused to the neighboring areas, the diffusion effects must also be considered to determine the LED intensity. In other words, not only the local information but also the neighboring block information should be considered to determine the LED intensity. Taking the diffusion into consideration, there have been several attempts to formulate the LBD as an optimization problem to minimize the distortion of an image after the

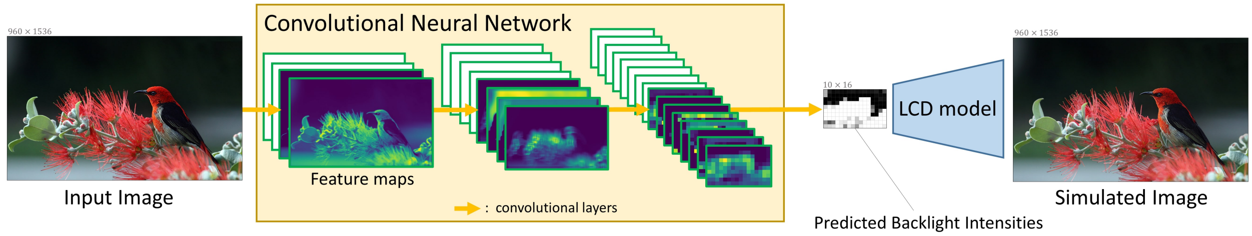

Fig. 1: We proposed a CNN-based local backlight dimming algorithm. To estimated the backlight intensities of each LED, we design our network as an autoencoder model, which takes an image as the input. Additionally, for end-to-end learning of network, we formulate the LCD model as fully differentiable model, taking simulation process of LCD into consideration.

local dimming [4], [5]. However, these approaches require burdensome computation and lack of generalization ability due to heuristically defined parameters.

In recent years, convolutional neural networks (CNNs) have become popular in many fields such as pattern recognition and computer vision, as they outperform traditional methods using hand-crafted features [7]–[9], [16], [17]. In this respect, we propose a CNN-based LBD algorithm which performs better than the above-stated conventional methods. In detail, to address the lack of generalization ability of hand-crafted features, we adopt a learning-based network that learns features for finding the optimal intensity of BLU. Precisely, we design a network as an autoencoder model [10], which takes an image as the input and produces the intensities of BLUs. By adopting the CNN, the proposed method can exploit rich representations of image contents. Also, unlike the conventional methods [1]–[3] that find the backlight intensity using hand-crafted features in pixel values within the corresponding block, our CNN-based method estimates the intensity using learned features in a wider range of pixel values. Hence, it also naturally considers the diffusion of light from the neighboring blocks.In summary, we consider deep features from neighboring blocks when computing the backlight intensity for the given input image. Experiments show that the proposed algorithm achieves plausible results in terms of objective measures.

II. Previous Works

Many LBD algorithms [1]–[3], [6] are based on the statistical information of pixel values in each block. For example, Chen *et al.*'s method [1] mainly includes two steps: calculation of initial intensities of LEDs and dark scene enhancement. In the first step, for considering the major gray levels, a weighted average of the histogram of each image block was used to calculate the initial backlight intensity. In the second step for dark scene enhancement, the initial LED intensities were enhanced more in the case of a very dark scene. Hsia *et al.* [2] proposed an LBD algorithm to improve the edge information. They noted that using the average value of each block reduces image details because the compensated LC transmittance value can be easily saturated. To alleviate this problem, blocks containing the large edge magnitude are detected by high variance values, and the initial backlight intensities of the corresponding block are enhanced. By doing this, the backlight intensity of the block containing large edges is not too dimmed so that it does not lose image details. Also, there are several attempts to formulate LBD as an optimization problem to minimize the distortion of an image after the local dimming [4], [5]. Cha *et al.* [4] presented an optimization-based LBD method that minimizes power consumption under a given allowable distortion for the LCDs with edge-lit LED backlight. To reduce the image quality fluctuation in this method, an inequality constrained optimization problem is formulated, and the steepest descent method is used to solve the optimization problem.

III. Backgrounds for the Proposed Method

A. Autoencoder

To address the lack of generalization ability when using hand-crafted features, we adopt a learning-based network, specifically an autoencoder (AE) model [10]. The AE is a type of artificial neural network that aims to learn a representative encoding of a set of data, typically for the purpose of non-linear dimensionality reduction [10]. The AE is mainly composed of two parts: encoder and decoder. They can be expressed with two mapping functions $\phi : \mathbf{R}^m \rightarrow \mathbf{R}^n$ and $\psi : \mathbf{R}^n \rightarrow \mathbf{R}^m$. The encoder stage of the AE takes the input $x \in X \subset \mathbf{R}^m$ and outputs the latent vector $z \in Z \subset \mathbf{R}^n$. The decoder stage of the AE attempts to reconstruct the original input, producing $\hat{x} \in X$ from the latent z. These non-linear mapping functions are trained to minimize reconstruction errors such as mean squared errors (MSE):

$$\phi, \psi = \arg\min ||x - \psi \circ \phi(x)||^2. \tag{1}$$

In other words, the AE learns an approximation of identity function so that its output $\hat{x}$ becomes as similar as to x. By imposing other constraints on the network such as limiting the dimension of Z space or formulating a deterministic decoder part, we can allocate specific meaning to the z. Mostly, the latent space Z has lower dimensionality than the input space X, $m > n$, then the $z = \phi(x)$ is the compressed representation of the input x.

In our case, the task is to estimate the intensities of a fixed number of LEDs z from the input image x, where the dimension of z is much smaller than x, given a constraint that the displayed image $\hat{x}$ through the decoder should be as similar as to the input image x. Hence, we model the estimation of

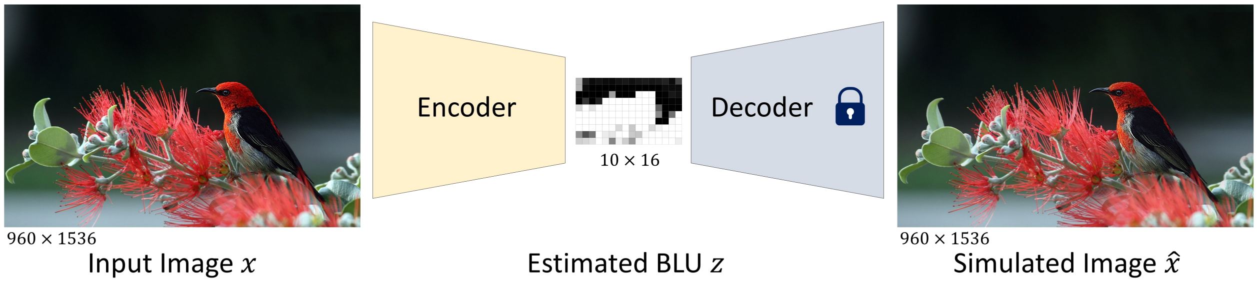

Fig. 2: Overall architecture of proposed network. Lock icon at the decoder denotes that it is a not-trainable function. The decoder is the process of making an output image with the BLU and given pixel values.

BLU through the encoder of the AE with trainable parameters of CNN, while we design the decoder part to simulate the mechanism of LCD. The overall architecture of the proposed network is shown in Fig. 2.

B. Convolutional Neural Network

We formulate the encoder stage of the proposed network using a convolutional neural network (CNN), which enables to capture rich representations of images and thus now widely used in various computer vision tasks [9]. The operation in a convolution layer can be represented as,

$$h_{out} = \sigma(W * h_{in} + c), \tag{2}$$

where W and c denote a set of learnable filter coefficient and bias, respectively. The convolution layer's output h_{out} is obtained by convolving $(*)$ layer's input h_{in} with W and addition with c, followed by a non-linear activation function σ. The output from a layer can be interpreted as an activation map, which shows the presence of specific features or patterns in the image. As one layer feeds its output into the next, extracted features hierarchically and progressively become more complex. By using the CNN, it is able to recognize patterns with extreme variability, with robustness to distortions and simple geometric transformations. Therefore, CNNs outperform traditional methods based on hand-crafted features in a variety of applications [7]–[9], [17]. In this respect, we formulate the encoder stage of our autoencoder using a CNN which determines the backlight intensity. Details on the architecture of the encoder stage are shown in Fig. 3 and will be handled in Section IV-A.

IV. Proposed Method

Without loss of generality, we formulate the LBD problem as the estimation of the brightness of 10×16 LEDs z from a 960×1536 RGB input image x. The overall architecture of the proposed network is shown in Fig. 2. For the arrangement of LEDs, we consider only the direct-lit backlight where the backlights are placed behind the screen with the equivalent block size for convenience. In other words, we divide BLUs into 10×16 blocks for local dimming control so that one unit covers to 96×96 pixels. These numbers are chosen just for an instance of many possible implementations, and we believe

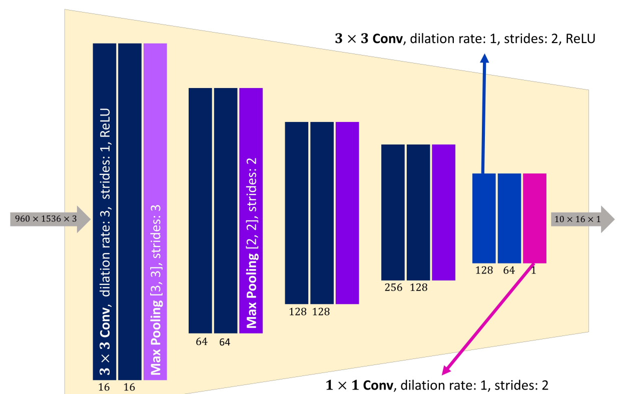

Fig. 3: Details of the encoder stage of proposed network.

that our method will also work for a large variation of these settings (block numbers, the arrangement of the light source, image size, *etc.*).

A. Structure of Encoder Part

As we already stated in Section III-A, to find a compressed representation of the input image, we design the encoder stage of the proposed network as shown in Fig. 3. Unlike previous methods [1]–[3] which handled only gray level such as Y channel or maximum value among RGB, we feed the original 3-channel RGB image. Hence, the encoder receives $960 \times 1536 \times 3$ input images as the input, and generates 10×16 compressed latent representative z. To be precise, the encoder is composed of 8 convolution layers, rectified linear unit (ReLU) as the non-linear activation function, and 4 max-pooling layers with different strides. We also adopt the dilated convolution filter [11] instead of the plain convolution, for enlarging the receptive field. For efficiently exploiting surround contextual information with a small number of parameters, the dilated filter pads zero between the components of filters with dilation factor. Finally, we apply a sigmoid function for the final output of the encoder to be bounded from 0 to 1, which is the normalized intensity of the BLU.

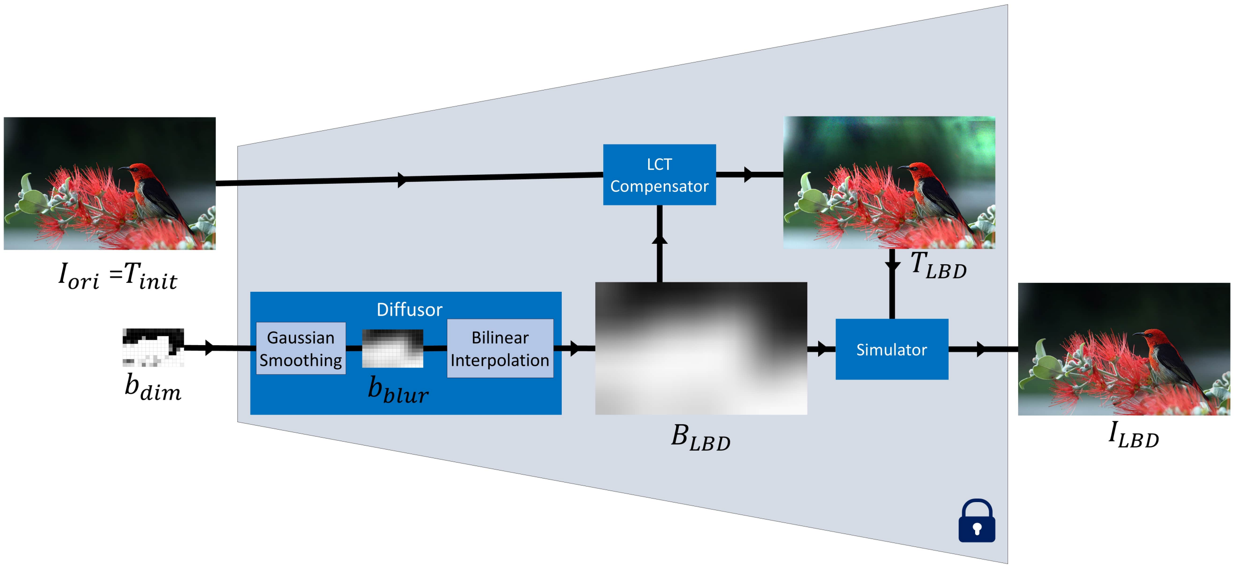

Fig. 4: Overall architecture of decoder stage of proposed network. The decoder is mainly composed of 3 parts: Diffusor, LC transmittance Compensator, and Simulator. Referred notations are handled in Section 4.2.

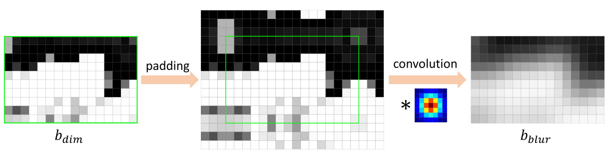

Fig. 5: Illustration of Gaussian smoothing. First, we pad boundary of b_{dim} with mirror reflections, considering the fact that the light is blocked and cannot be spread at the boundary. Secondly, padded BLUs are convolved with 7×7 Gaussian filter for blurring.

B. Structure of Decoder Part

To give a constraint for the 10×16 latent array to work as BLUs, we design decoder part concerning the physics of LCD. As shown in Fig. 4, the LCD model is composed of 3 parts: diffusor, LC transmittance compensator, and simulator.

1) Diffusor: Since the LED is a discrete light source, multi-layered optical diffuser films are necessary to disperse the LED light. This makes the illumination spread over the regions as a plane light source so that the blocking artifact around the backlight block is reduced. To model the effect of the optical diffuser film, we utilize two operations, namely Gaussian smoothing and bilinear interpolation. First, the Gaussian smoothing operation is to model the constructive diffusion property of light, which is given by

$$b_{\text{blur}}(c,r) = \sum_{k=-3}^{3} \sum_{l=-3}^{3} b_{\text{dim}}(c+k, r+l) \times G(k,l), \quad (3)$$

where

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}, \quad (4)$$

where b_{dim} and b_{blur} denote dimmed and blurred backlight intensities at (c,r)-th block, respectively. Also, $c \in \{1,2,\ldots,10\}$ and $r \in \{1,2,\ldots,16\}$ represent column and row index of a block, respectively. We use 7×7 Gaussian filter with $\sigma = 1.5$. At the edge of the panel, the light has different diffusion property from the fact that the light is blocked and cannot be spread. Therefore, when convolving the Gaussian smoothing filter, we pad boundary with mirror reflections of the BLUs as shown in Fig. 5. Furthermore, with the assumption that each LED is located at the center of the corresponding block, we model the optical diffuser films using a bilinear interpolation [18]. As shown in Fig. 4, this model eventually gives the uniformly-dispersed luminance of BLUs (B_{LBD}) as a plane light source.

2) LC transmittance compensator: The displayed image on LCD is derived by the product of backlight luminance with LC transmittance. Before applying LBD, we assume that the initial luminance of BLU is at peak luminance (=1), which means that all light sources are equally turned on. Formally, the original pixel value of the LCD panel at the position (i,j) is modeled as

$$I^c_{ori}(i,j) = T^c_{init}(i,j) \times B_{init}(i,j) \quad (5)$$

where I^c_{ori}, T^c_{init}, and B_{init} denote the original pixel value of the c-th channel, initial LC transmittance of the c-th channel, and backlight intensity, respectively. Before the application of LBD, all the backlights are fully turned on, which corresponds to $B_{init}(i,j) = 1$ for all (i,j). Hence, the pixel value before the LBD is

$$I^c_{ori} = T^c_{init}(i,j). \quad (6)$$

When the backlight luminance is dimmed, to keep the same luminance value with the original image, the LC transmittance must be compensated to be reciprocally proportional to the dimming ratio. For this, we define the compensation factor C as

$$C(i,j) = \frac{B_{init}(i,j)}{B_{\text{LBD}}(i,j)} = \frac{1}{B_{\text{LBD}}(i,j)} \quad (7)$$

where B_{LBD} is the dimmed lacklight in $(0,1]$. However, since the LC transmittance is bounded, we have to impose a saturation constraint through the clip function. Then, the compensated LC transmittance after the LDB is expressed as

$$T^c_{\text{LBD}}(i,j) = f_{clip}(C(i,j) \times T^c_{init}(i,j)), \quad (8)$$

where $f_{clip}(\cdot)$ denotes the clip function which bounds the input values from 0 to 1. When the backlight is overly dimmed ($B_{\text{LBD}} \ll 1$), conversely the compensation factor is boosted ($C \gg 1$), and then, the compensated LC transmittance ($C \times T^c_{init} > 1$) values are saturated due to the clip function. As a result, the simulated image cannot fully reconstruct the original value, which results in loss of details. To prevent the loss of details from saturation, several algorithms [1], [6] used an experimental lookup table (LUT) which contains pre-determined compensation factors according to the relation between I_{ori} and B_{LBD}. However, in order to show the intrinsic performance of the models, the LUT is excluded in all experiments.

3) Light Leakage: In addition to the diffusor and compensator, it is required to consider the leakage of LC. Precisely, with the transmittance T^c_{LBD} and backlight intensity B_{LBD}, the brightness on display had to be $T^c_{LBD} \times B_{\text{LBD}}$ when there is no leakage. But since there is a small amount of leakage that is proportional to B_{LBD}, the actual brightness is modeled as

$$I^c_{\text{LBD}}(i,j) = \epsilon B_{\text{LBD}}(i,j) + (1-\epsilon) B_{\text{LBD}}(i,j) T^c_{\text{LBD}}(i,j), \quad (9)$$

where ϵ denotes the light leakage factor. It is determined to depend on the characteristics of LC material, device structure, and viewing angle. In our simulation, a constant $\epsilon = 0.03$ is used for the entire pixels for simplicity.

C. Loss function for Traning the Proposed Network

To design a loss function for our network, we investigate several loss terms which reflect the discrepancy between two vectors. Among them, inspired by the work for image restoration task [16], we adopt L2-norm loss function to simulate the displayed image from the estimated BLUs as close as to the original image. Thus, the loss function for training our network is

$$Loss = \frac{1}{2} \sum_{i=1}^{w} \sum_{j=1}^{h} \sum_{c=1}^{3} (I^c_{ori}(i,j) - I^c_{LBD}(i,j))^2 \quad (10)$$

where w and h denote the width and height of an image, respectively.

D. Dataset

For training and test, we use DIV2K dataset [13] which is released for single image super-resolution. The DIV2K dataset consists of high resolution (2K) RGB images with a large diversity of contents, composed of 800 images for training and 100 images for validation. Additionally, we manually crawled 536 RGB images from the Internet for paying special attention to the image quality, and also for the diversity of contrast. Including these images in training dataset, we augment the dataset with 1,336 images, so that our dataset covers a wide range of contrast. For efficiency, we train with sub-images randomly cropped from the original dataset with the size of 480×768, which is one-quarter of the target image size. Moreover, we augment the dataset by randomly masking the sub-images for the robustness of the network, providing training images with extremely high contrast with irregular shapes. Finally, we construct a dataset with 106,084 cropped sub-images for the training. Typical examples are shown in Fig. 6.

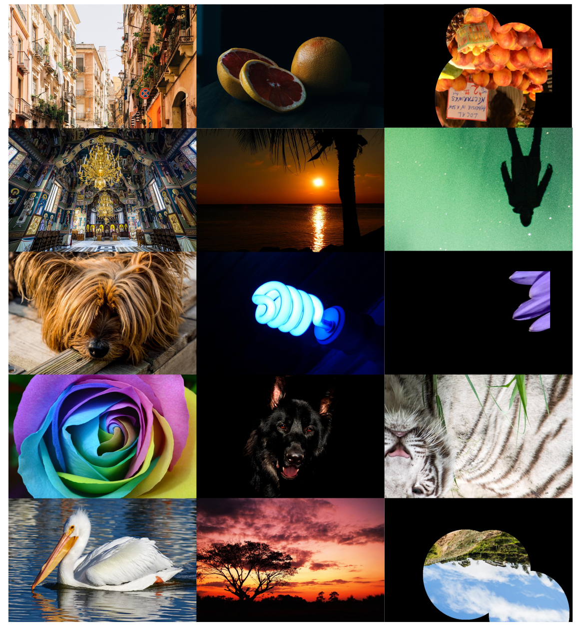

Fig. 6: Typical examples of used dataset. From the first to the third columns, images are examples of DIV2K, manually crawled, and augmented images used for training, respectively.

V. Experimental Results

A. Training Method

During training, we estimate 5×8 BLUs from 480×768 sub-images. The size is enough for the network to consider surrounding contextual information. We have trained the network using Adam optimizer [18] with the mini-batch size of 32. We used 0.0005 as the initial learning rate with 0.8 decay rate in every 20 epochs. Dilation rate of the filters in the encoder is empirically set to 3. Additionally, we applied weight decay regularization [14] to prevent overfitting.

B. Comparions with Other LBD Methods

We compare our method with three LBD algorithms: a conventional method which is based on block-wise Max value,

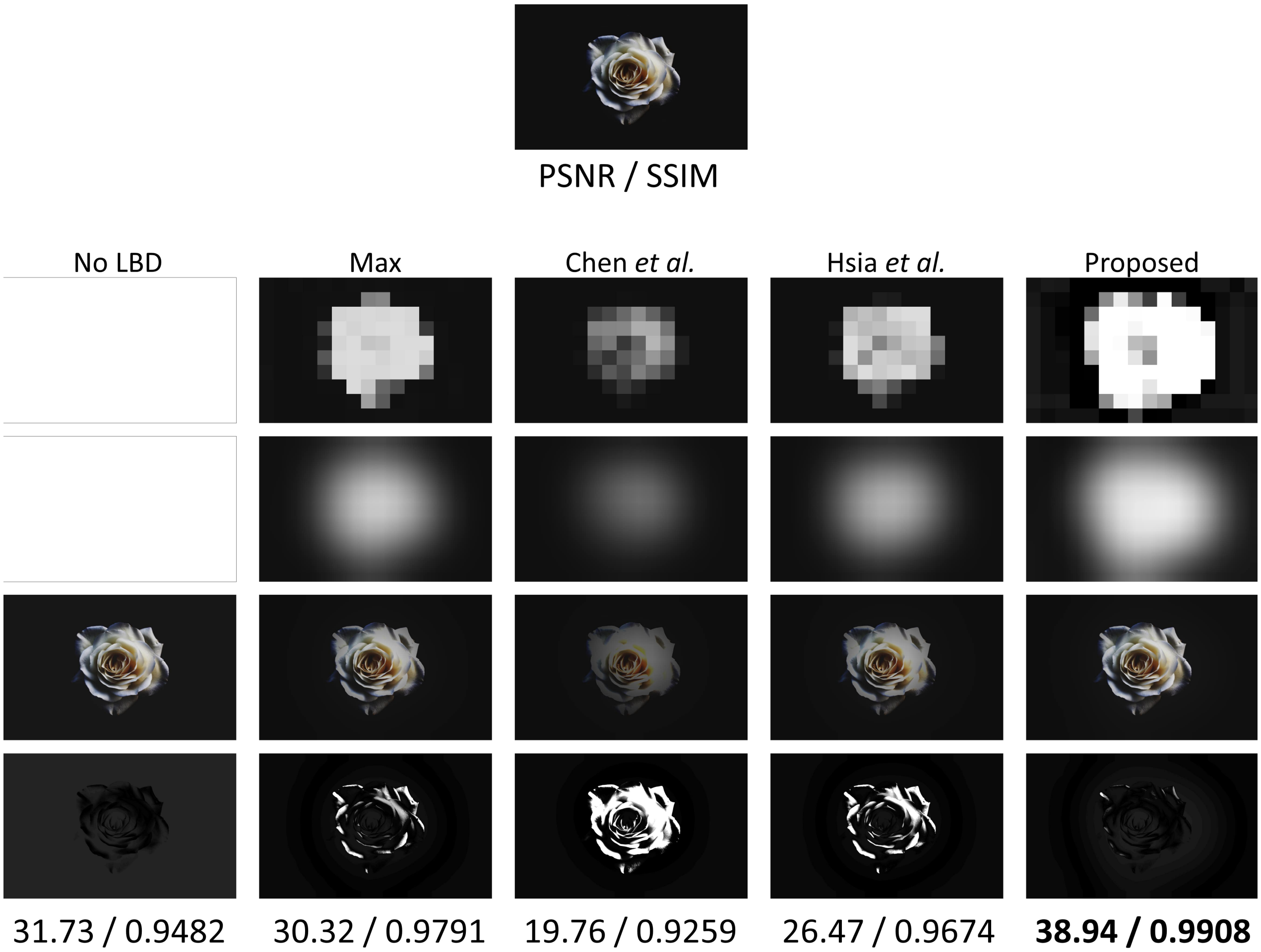

Fig. 7: Experimental results applying LBD algorithms to img0843 from DIV2K [13]. From the first to the fourth rows, images are the results of estimated BLU, diffused BLU, simulated images, and difference images from the original, respectively. For better visualization, the difference images are amplified. The best quantitative results are highlighted in bold face.

Chen *et al.* 's [1], and Hsia *et al.* 's [2]. For the latter two methods, no source codes are available, so we implemented the codes based on the original papers with MATLAB. The 10×16 BLU is estimated by each method from 960×1536 image. Then we compare the BLU estimation performance of each method with the images displayed by a common LCD model. For the quantitative comparisons, peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) [15] are used. Since the main purpose is to display the contents of the LCD panel to be close to the original, it is meaningful to use the PSNR associated with mean squared error as a measure of closeness. Also, under the assumption that human visual perception is related to the structural information from a scene [15], we also use the SSIM to assess the image quality.

1) Evaluation on DIV2K Dataset: We evaluate our method with other LBD algorithms on the DIV2K validation dataset. As shown in Fig. 7, the proposed method achieves the largest performance gain compared to the other methods. Specifically, it can be seen that other methods estimate backlight intensities only in the parts where the contents exist. However, in our results, the backlight intensities exist even if there is no content at all. This suggests that the proposed method estimates

TABLE I: Quantitative comparisons of Local Backlight Dimming algorithms

DIV2K_HR_set (100 images resized 960 1536)	Quantitative measure	
	PSNR	SSIM
Maximum method	36.91	0.9842
Chen *et al.* [1]	20.02	0.8345
Hsia *et al.* [2]	33.30	0.9757
Proposed method	**37.91**	**0.9862**

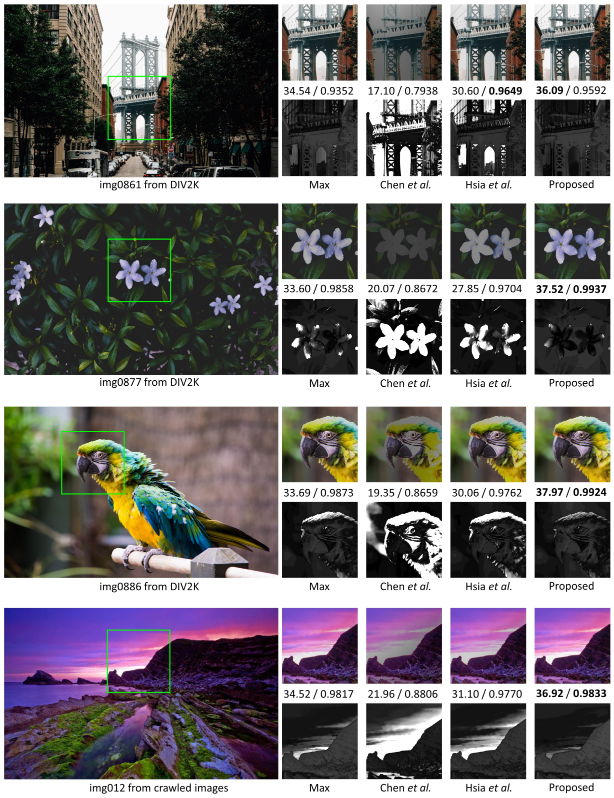

Fig. 8: Experimental results applying LBD algorithms. Gray scale images are difference image with the original. For better visualization, we multiply 10 to the difference results. The best quantitative results are highlighted in bold face.

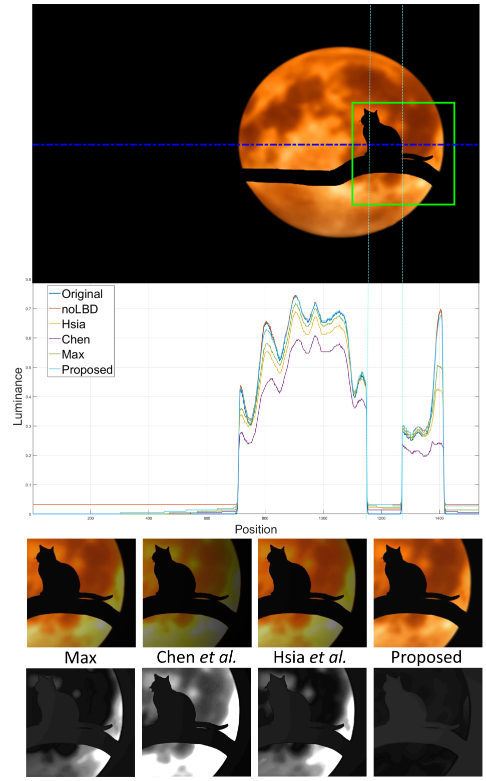

Fig. 9: Luminance profile of pixels in the dashed blue line of the image. Gray scale images are difference image with the original. For better visualization, we amplify the difference image

intensities of LEDs using not only local information but also neighbor information, unlike other LBD algorithms. As a result, our proposed method prevents the degradation of the edge contents which occurs due to the lack of light as shown in the other comparisons.

Also, the average values of quantitative results on 100 images in DIV2K validation set are shown in TABLE I. We can observe that the proposed method outperforms others in terms of PSNR and SSIM. Although the maximum value method also achieves quite high PSNR, it is prone to make halo artifacts from the leakage property of LC and the diffusion of light. Conversely, mean value based methods, such as Chen *et al.*'s and Hsia *et al.*'s, are easy to make clipping artifacts due to LC transmittance saturation [19]. Also, due to the lack of generalization of mean value based methods with heuristically-defined parameters, their results are more degraded than the maximum value methods as shown in TABLE I. On the contrary, as shown in Fig. 8, the proposed method works well for the images of various contents and environments, showing good generalization ability.

2) Luminance Profile Analysis: Fig. 9 plots the displayed luminance value of pixels in the dashed blue line of the image. The results without LBD, which is the red line in the graph, shows that the saturation of LC transmittance does not occur, *i.e.*, preserve details. But the real black is not expressed due to the light leakage. On the other hand, in the case of LBD methods, the real black is well expressed by reducing the effect of leakage, but degradations in the boundary of contents occur due to the saturation of LC transmittance. Results in the right peak of the luminance profile demonstrate the effectiveness of the proposed method compared to the others. Chen *et al.* [1] and Hsia *et al.* [2] initially estimate the intensities of LEDs based on the mean value of the block. As shown, Chen *et al.*'s [1] method shows severe degradation due to saturation. In the case of Hsia *et al.*'s [2] result, although they enhance the luminance value of the block owing to being classified as the edge of contents, it shows saturation due to the lack of light. In other words, adequate luminance cannot be covered only by an LED in a corresponding block. However, in our case, as shown in the luminance profile, degradation due to saturation is quite mitigated while suppressing the light leakage. Although some halo artifacts occur, it is believed to be negligible. Also, Fig. 10 shows activation maps of a layer in the CNN, which demonstrates that our CNN-based method fully exploits various features from an image.

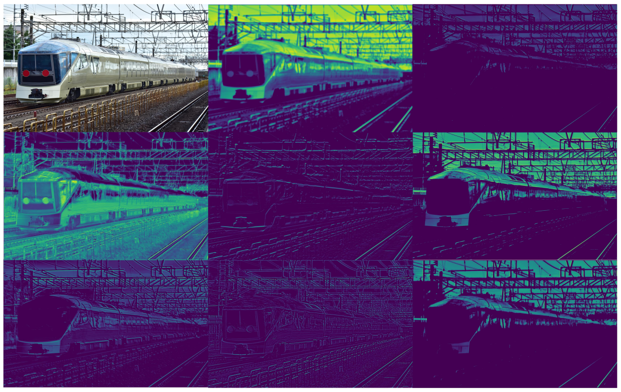

Fig. 10: Visualization of some activation maps of the second convolution layer of our encoder. The figure shows that our method exploits various directions of edges, textures, dark regions, bright regions, *etc.*

VI. Discussion

We have proposed a CNN-based local backlight dimming algorithm. Unlike the conventional methods that locally control the BLU, the proposed method considers the surrounding contextual information by using a large receptive field size of CNN. Also, the diffusion property of light and leakage property of LC are included in the backpropagation path, which eventually makes the displayed content of LCD as close as to the original. Experimental results show that the proposed method yields outstanding qualitative results with less loss of details and a higher contrast ratio. Also, it is shown from the visualized results that the proposed method outperforms other methods in terms of perceptual quality. Finally, we note that the complexity of CNN can be an issue for its practical usage in the LCD panels. Hence, we also considered minimizing the computation complexity and memory by squeezing the number of filters and the depth of layers. The number of parameters for this light implementation is $108K$, which is much less than the well-known conventional VGG16 [17] that requires $138,358K$ for classification tasks. We leave it as a future work to further squeeze or quantize the CNN to reduce overall computation and memory requirements while maintaining the performance. Our codes and datasets will be made publicly available on a website.

Acknowledgment

This research was financially supported by the Ministry of Trade, Industry, and Energy (MOTIE), Korea, under the “Regional Innovation Cluster Development Program(R&D, P0002072)” supervised by the Korea Institute for Advancement of Technology (KIAT), and the BK21 FOUR program of the Education and Research Program for Future ICT Pioneers, Seoul National University in 2020.”

References

[1] Chen, Hanfeng, et al. ”39.3: Locally Pixel-Compensated Backlight Dimming for Improving Static Contrast on LED Backlit LCDs.” SID Symposium Digest of Technical Papers. Vol. 38. No. 1. Oxford, UK: Blackwell Publishing Ltd, 2007.

[2] Hsia, Shih-Chang, et al. ”High-performance local dimming algorithm and its hardware implementation for LCD backlight.” Journal of Display Technology 9.7 (2013): 527-535.

[3] Park, Jae Sung, Insung Hwang, and Nam Ik Cho. ”Human visual perception-based localized backlight scaling method for high dynamic range LCDs.” Journal of the Society for Information Display 24.8 (2016): 471-486.

[4] Cha, Seungwook, et al. ”An optimized backlight local dimming algorithm for edge-lit LED backlight LCDs.” Journal of Display Technology 11.4 (2015): 378-385.

[5] Burini, Nino, et al. ”Modeling power-constrained optimal backlight dimming for color displays.” Journal of Display Technology 9.8 (2013): 656-665.

[6] Cho, Hyunsuk, and Oh-Kyong Kwon. ”A backlight dimming algorithm for low power and high image quality LCD applications.” IEEE Transactions on Consumer Electronics 55.2 (2009): 839-844.

[7] He, Kaiming, et al. ”Deep residual learning for image recognition.” Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[8] Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. ”U-net: Convolutional networks for biomedical image segmentation.” International Conference on Medical image computing and computer-assisted intervention. Springer, Cham, 2015.

[9] Yamashita, Rikiya, et al. ”Convolutional neural networks: an overview and application in radiology.” Insights into imaging 9.4 (2018): 611-629.

[10] Baldi, Pierre. ”Autoencoders, unsupervised learning, and deep architectures.” Proceedings of ICML workshop on unsupervised and transfer learning. 2012.

[11] Yu, Fisher, and Vladlen Koltun. ”Multi-scale context aggregation by dilated convolutions.” arXiv preprint arXiv:1511.07122 (2015).

[12] Hulze, Hendriek Groot, and Pierre deGreef. ”50.2: Power savings by local dimming on a LCD panel with side lit backlight.” SID symposium digest of technical papers. Vol. 40. No. 1. Oxford, UK: Blackwell Publishing Ltd, 2009.

[13] Agustsson, Eirikur, and Radu Timofte. ”Ntire 2017 challenge on single image super-resolution: Dataset and study.” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. 2017.

[14] Ng, Andrew Y. ”Feature selection, L 1 vs. L 2 regularization, and rotational invariance.” Proceedings of the twenty-first international conference on Machine learning. ACM, 2004.

[15] Wang, Zhou, et al. ”Image quality assessment: from error visibility to structural similarity.” IEEE transactions on image processing 13.4 (2004): 600-612.

[16] Zhao, Hang, et al. ”Loss functions for image restoration with neural networks.” IEEE Transactions on computational imaging3.1 (2016): 47-57.

[17] Simonyan, Karen, and Andrew Zisserman. ”Very deep convolutional networks for large-scale image recognition.” arXiv preprint arXiv:1409.1556 (2014).

[18] Parker, J. Anthony, Robert V. Kenyon, and Donald E. Troxel. ”Comparison of interpolating methods for image resampling.” IEEE Transactions on medical imaging 2.1 (1983): 31-39..

[19] Huang, Yuge, et al. ”Prospects and challenges of mini-LED and micro-LED displays.” Journal of the Society for Information Display (2019).