Adversarial and adaptive tone mapping operator: multi-scheme generation and multi-metric evaluation

Xingdong Cao,[a] Kenneth Lai,[a] Michael Smith,[b] and Svetlana Yanushkevich[a,*]

[a]University of Calgary, Biometric Technologies Laboratory, Department of Electrical and Software Engineering, Calgary, Alberta, Canada

[b]University of Calgary, Department of Electrical and Software Engineering, Calgary, Alberta, Canada

Abstract. Tone mapping is one of the main techniques to convert high-dynamic range (HDR) images into low-dynamic range (LDR) images. We propose to use a variant of generative adversarial networks to adaptively tone map images. We designed a conditional adversarial generative network composed of a U-Net generator and patchGAN discriminator to adaptively convert HDR images into LDR images. We extended previous work to include additional metrics such as tone-mapped image quality index (TMQI), structural similarity index measure, Fréchet inception distance, and perceptual path length. In addition, we applied face detection on the Kalantari dataset and showed that our proposed adversarial tone mapping operator generates the best LDR image for the detection of faces. One of our training schemes, trained via 256 × 256 resolution HDR–LDR image pairs, results in a model that can generate high TMQI low-resolution 256 × 256 and high-resolution 1024 × 2048 LDR images. Given 1024 × 2048 resolution HDR images, the TMQI of the generated LDR images reaches a value of 0.90, which outperforms all other contemporary tone mapping operators. [DOI: 10.1117/1.JEI.30.4.043020]

Keywords: adaptive tone mapping; high-dynamic range images; generative adversarial networks.

Paper 210080 received Feb. 12, 2021; accepted for publication Aug. 3, 2021; published online Aug. 19, 2021.

1 Introduction

The dynamic range of an image is described as the variation of luminance in different parts of the image.[1] The majority of real-life images are of low dynamic range (LDR) and are generally represented by an 8-bit integer per pixel format.[2] In contrast, high dynamic range (HDR) uses more bits (16/32) to quantify the pixel values. Even though HDR images can better describe a scene, most common 8-bit display methods are not compatible with HDR images. A cost-effective method of displaying HDR images is to convert them into LDR images as opposed to using a 16-bit display setting.

Many tone mapping operators (TMOs) have been proposed and have shown incredible progress in many scenarios. Even though tone mapping is one of the most common ways to perform HDR to LDR conversion, TMOs have many limitations, such as generalization, parameter turning, expert knowledge, and model instability.

The main research question of this work is: Is it possible to propose a TMO that can adaptively tone-map all HDR images with different contents? In this paper, we seek to answer this question by exploring deep learning techniques. We propose a specific deep learning network, a conditional generative adversarial network (cGAN),[3] to adaptively convert an HDR image into an LDR image. Our proposed model is training via HDR–LDR image pairs containing assorted content, including natural scenarios, indoor/outdoor scenes, regular/irregular geometric shapes, colorful/monochrome objects, and drastic luminance changes.

*Address all correspondence to Svetlana Yanushkevich, syanshk@ucalgary.ca

In general, the implementation of any generative adversarial networks (GANs) requires an objective loss function. In deep learning networks, the loss function measures the difference between the output and input images. Common loss functions are the absolute (called $\mathcal{L}_1$) or squared (called $\mathcal{L}_2$). In this work, we implement a unique network composed of general cGAN loss, feature matching loss, and perceptual loss. Combining these losses allows the proposed adversarial tone mapping operator (adTMO) to learn the distribution of ideally tone-mapped images.

For low-resolution image-to-image translation tasks, cGAN has shown great success in generating high-quality target images.[4] However, for high-resolution image-to-image translation tasks, many problems exist. These problems require complex models to combat tilling patterns, local blurring, and saturated artifacts.[5,6] One of the main deterrences of using high-resolution images is the amount of resources required for training, specifically the amount of time required for convergence. In our work, we explore the possibility of using low-resolution images to train a cGAN model ("U-Net" G and PatchGAN D). We extended the work on adTMO[7] to include additional metrics such as structural similarity index measure (SSIM), perceptual path length (PPL), Fréchet inception distance (FID), and multi-scale structural similarity index measure (MS-SSIM), as well as the performance metrics for face detection. We show that adTMO outperforms most other TMOs when testing on low- and high-resolution HDR images.

This paper aims to design a smart TMO that can adaptively convert complex scenic HDR images into LDR images. The main contributions of our work are listed as follows.

1. We propose adTMO, a variant of cGAN capable of adaptively generating high-resolution and high-quality LDR images.
2. We explore different training and testing schemes, in order to find the best possible combination to generate the highest quality images.
3. We evaluate the performance of adTMO and other TMOs using metrics such as SSIM and FID. In addition, we look at the performance of face detection applied to the different tone-mapped images.

This paper is organized as follows: Section 2 provides a literature review related to TMOs, cGAN, and metrics used for evaluating image-to-image translation tasks. Section 3 describes the architecture of adTMO and the different training/testing schemes we apply. Section 4 details the databases used for training and the preprocessing and postprocessing steps applied to the images. Section 5 summarizes the results of adTMO. Section 6 concludes our paper.

2 Related Work

In this section, we provide a short review of tone-mapping literature, cGAN, and metrics used for evaluating image-to-image translation tasks.

2.1 *TMOs*

Over the past 20 years, different TMOs have been designed to convert HDR images into LDR images. They can be divided into two categories, global TMOs and local TMOs, based on how they work on image pixels. Global TMOs, such as Larson et al.[8] and Drago et al.,[9] apply the same function on all pixels of an image. Global TMOs take less time to convert HDR images, but the output LDR images have reduced contrast. Local TMOs, e.g., Chiu et al.[10] and Tumblin et al.,[11] calculate the output pixel value based on the input and its neighboring pixels. Local TMOs can preserve the local structure and generate good contrast but at a cost of more computation time. In addition, most TMOs can only deal with some specific scenarios and do not generalize well with regard to image content.

2.2 *Generative Adversarial Networks*

First proposed by Goodfellow in 2014,[12] GAN has shown great success in many fields. GAN consists of a generator model (G) and a discriminator model (D). The goal of G is to generate fake samples that are real enough to fool D. For D, its goal is to distinguish real samples from collected databases and fake samples generated by G. By training G and D simultaneously, they can compete with each other and achieve an equilibrium allowing G to implicitly learn the distribution of real samples from the collected databases, without the need of complex loss functions.

In this paper, we adopt cGAN,[3] so that the goal of G changes to generating fake samples under new conditions. Many low-resolution image-to-image translation tasks, such as semantic labels to photos and architectural labels to photo, adopt cGAN to generate target images and achieve satisfactory results.[4] Patel et al.[13] conducted a similar work using cGAN to convert HDR images into LDR images, but they only tested with 256×256 resolution image crops. A complex multi-scale architecture for high-resolution image-to-image tasks is proposed by Wang et al.[5] and Rana et al.[6] Those proposed networks required high-resolution training images and took many resources including memory and time to train. It took a week to train the multi-scale network[6] using a 12-GB NVIDIA Titan-X GPU on a Intel Xeon e7 core i7 machine.

Due to the downsampling process in the generation part of cGAN, it is challenging for the input images to preserve the fine details. A bilateral filter is a common method to perform edge-preserving and noise-reducing operations which can be adopted to preserve the finer details of an image.[14] A method that optimizes the bilateral filtering method to have a constant time O(1) was proposed by Porikli.[15] Others proposed to preserve edges in images include global image smoothing based on the weighted least squares (WLS)[16] and guided image filter.[17] Extended work on WLS was conducted by Min et al.[18] to create a fast variant, achieving comparable results but requiring much less computational time. Optimization to the guided image filtering technique was performed by incorporating an edge-aware weighting into the guided filter, which greatly reduced the halo artifacts in images.[19]

Zheng et al.[20] proposed to create a hybrid model that consists of both a model-driven and data-driven approach to generate a higher quality image. In this paper, we have mainly focused on the data-driven approach via the use of cGAN. However, there is an immense value in a hybrid model; thus we plan to create such a hybrid model in future works by integrating the model-driven portion into our data-driven model.

2.3 *Evaluation for Image-to-Image Translation Task*

Evaluation of image-to-image translation tasks remains an open question. SSIM was proposed by Wang et al.[21] to compare the structural information based on the human visual system. SSIM is commonly used to compare the similarity between the generated images and the ground-truth images. It is defined by Wang et al.[21] as follows:

$$\mathrm{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}, \tag{1}$$

where μ is the mean with respect to x or y, σ is the variance with respect to x or y, and C_1, C_2 are the constants defined as $(0.01L)^2$ and $(0.03L)^2$ (L is the dynamic range of the pixels), respectively.

Based on SSIM, a metric called multi-scale structural similarity (MS-SSIM)[22] was designed to incorporate the variations of viewing conditions.

FID[23] was proposed to capture the similarity between the generated and ground-truth images. To compute FID, both the generated and real images are propagated through a pretrained Inception V3 model[24] and their difference from the last pooling layer is used. A smaller FID represents higher similarity, that is given an FID of 0, two images are identical. The FID is defined as follows:

$$\mathrm{FID} = \|\mu_r - \mu_g\|^2 + \mathrm{tr}\left(\Sigma_r + \Sigma_g - 2\sqrt{\Sigma_r\Sigma_g}\right), \tag{2}$$

where μ represents the mean for the real (r) and generated (g) images, Σ represents the covariance for the real (r) and generated (g) images, and tr is the trace linear function.

Similar to FID, PPL[25] uses the pretrained VGG16[26] as embeddings to calculate the perceptual similarity between two images. As with FID, a smaller PPL means that two images have a greater perceptual similarity.

Evaluating the performance of TMOs is also an issue for tone mapping operations. One intuitive solution is a subjective evaluation, which involves human participants ranking LDR images generated by different TMOs based on their subjective preference. Such subjective evaluation takes a lot of time and energy, with the results unstable across different participant groups.[27] Another solution is objective metrics, e.g., tone-mapped image quality index (TMQI)[28] and TMQI-II,[29] widely used in tone-mapping optimization studies.[6,30] TMQI represents a form of indexing that considers the naturalness of tone-mapped LDR images, and structural fidelity between the HDR and tone-mapped LDR images expressed as[28]

$$\mathrm{TMQI}(\mathbf{H},\mathbf{L}) = a[S(\mathbf{H},\mathbf{L})]^{\alpha} + (1-a)[N(\mathbf{L})]^{\beta}, \tag{3}$$

where **H** and **L** denote the original HDR image and the tone-mapped LDR image, S and N denote the structural fidelity and statistical naturalness measures, respectively. α and β control the sensitivities of S and N, and $0 \le a \le 1$ adjusts the relative weights between S and N. In this paper, we use the default α, β, and a, recommended by Yeganeh and Wang.[28]

3 Proposed Method

In this section, we will detail our proposed adTMO to convert HDR images into LDR images, the architecture of our G and D, the objective function we use, and the different training/testing schemes we deploy.

3.1 *cGAN-Based adTMO*

In this paper, we construct adTMO based on the principle of cGAN[3] that can translate HDR images into LDR images. Figure 1 shows the training pipeline of our proposed adTMO. We train D using (HDR, LDR) pairs where D is trying to predict (HDR, RealLDR) pair as real and predict (HDR, FakeLDR) pair as fake. G is trying to generate FakeLDR that is real enough so that D is unable to distinguish FakeLDR from RealLDR. We train G and D simultaneously, specifically, in each iteration, we train D twice with weight set to 0.5 [once using the (HDR, RealLDR) pair, and once using the (HDR, FakeLDR) pair].

3.2 *Network Architectures*

We adopt the network architectures from Isola et al.,[4] where G is a U-Net[31] and D is a 70×70 PatchGAN,[32] both using convolution-BatchNorm-LeakyRelu[33] blocks with $\alpha = 0.2$.

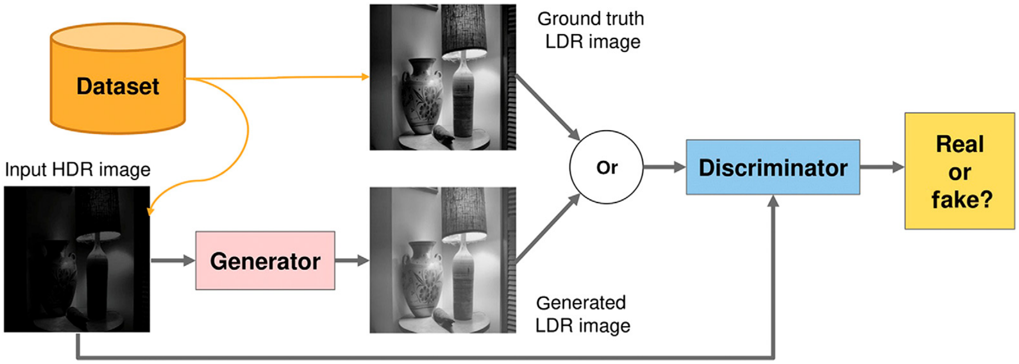


Fig. 1 Training pipeline of cGAN. *D* is trained to distinguish ground truth LDR image from the generated LDR image. *G* is trained to generate LDR image that is real enough to fool *D*.

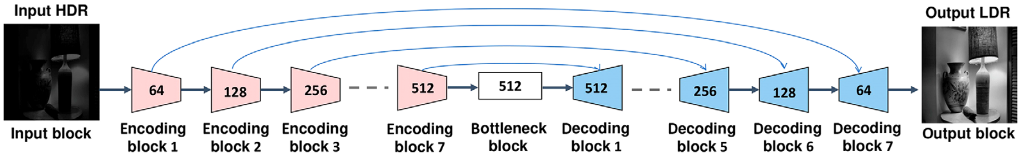


Fig. 2 Architecture of the U-Net generator with one input block, seven encoding blocks, one bottleneck block, seven decoding blocks, and one output block. There is a direct skip connecting each encoding–decoding pair.

3.2.1 *Generator architecture*

Figure 2 shows the architecture of our G, which is a U-Net consisting of one input block, seven encoding blocks, one bottleneck, seven decoding blocks, and one output block. Each encoding block will down-sample image size by $1/4$ (1/2 of width and 1/2 of height) of the previous block with strides $= 2$, and each decoding block will up-sample the previous block by 4 times. We added direct connections between the encoding and decoding blocks in order to preserve some of the finer details that may have been lost during the downsampling process. This direct connection, also called skip connection, allows for the gradient of the later layers to propagate back to the earlier layers. Such propagation prompts the model to learn, more efficiently, the mapping between the input and output layers, allowing for the finer details to be recovered from the downsampling process. For the i'th decoding block, we add a direct skip from the last i'th encoding block and concatenate the two blocks in channel before applying the LeakyRelu activation function. The filter size is set to 4×4 for all blocks. The filter number is set to 64 for the first encoding block and doubles for each of the next encoding block until it reaches 512, then remains unchanged. The filter number for each decoding block is the same as the encoding block with which it connects. For the bottleneck block, the filter number is set to 512, and the activation function is ReLU. For the output block, the filter number is set to 1 and the activation function is sigmoid. We can feed our G with images of different sizes given it is fully convolutional.

3.2.2 *Discriminator architecture*

Figure 3 shows the architecture of our D. This is a 70×70 PatchGAN consisting of one input layer, five encoding blocks, and one output block. The input layer concatenates the input HDR and LDR image in the color channel. Each of the first four encoding blocks will down-sample image size to $1/4$ of the previous block with strides $= 2$. For the last encoding block, we set strides $= 1$, leaving the image size unchanged. The number of filters for each encoding blocks is defined as follows 64, 128, 256, 512, and 512. The output block has 1 filter, with strides $= 1$, a sigmoid activation and outputs a 16×16 matrix. Each value in the output matrix maps to a 70×70 receptive field in the input layer, identifying this patch as either real or fake.

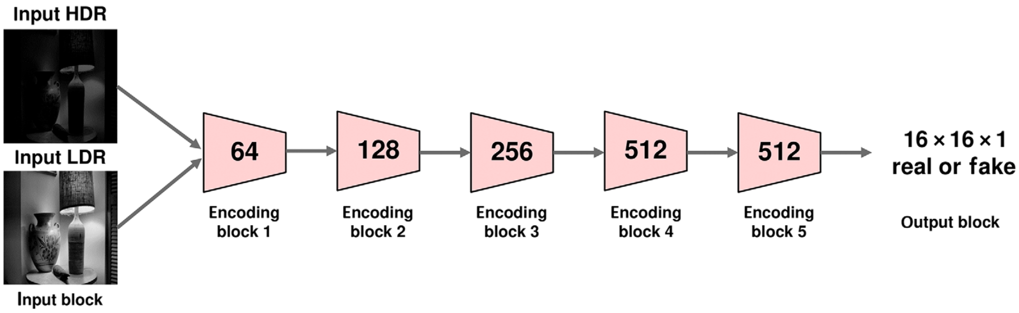


Fig. 3 Architecture of the PatchGAN discriminator. Each value in the output matrix identifies a 70×70 receptive field in the input layer as either real or fake.

3.3 *Objective Function*

As discussed earlier, the goal of G is to convert an HDR image into its tone-mapped LDR version, and the goal of D is to distinguish the generated LDR image from the ground-truth LDR image. The objective of cGAN[3] can therefore be written as

$$\begin{aligned}\mathcal{L}_G(G,D) &= \mathbb{E}_{(x)}\ \log(1-D(x,G(x)))\\ \mathcal{L}_D(G,D) &= -\mathbb{E}_{(x,y)}\ \log\ D(x,y) - \mathbb{E}_{(x)}\ \log(1-D(x,G(x))),\end{aligned} \tag{4}$$

where G tries to minimize $\mathcal{L}_G(G,D)$, and D tries to minimize $\mathcal{L}_D(G,D)$.

In addition to the cGAN loss, we incorporated a feature matching loss $\mathcal{L}_{\text{FM}}$ based on D. We extract features from multiple layers of D and attempt to match these intermediate representations between the real and generated LDR image, i.e., we minimize the difference between the features via the L1 norm:

$$\mathcal{L}_{\text{FM}}(G,D) = \mathbb{E}_{(x,y)}\sum_{i=1}^{M}\frac{1}{U_i}[\|D^{(i)}(x,y) - D^{(i)}(x,G(x))\|_1], \tag{5}$$

where $D^{(i)}$ denotes the i'th layer with U_i activations of D, and M is the number of layers of D. In this experiment, we chose five convolution layers in the five encoding blocks of D.

Additionally, we appended the perceptual loss $\mathcal{L}_{\text{prp}}$ used by Johnson et al.,[34] which consists of the features computed from every single layer of the pretrained Inception V3 network,[24] given by

$$\mathcal{L}_{\text{prp}}(G) = \mathbb{E}_{(x,y)}\sum_{i=1}^{N}\frac{1}{V_i}[\|F^{(i)}(y) - F^{(i)}(G(x))\|_1], \tag{6}$$

where $F^{(i)}$ denotes the i'th layer with V_i activations of the Inception V3 network, and N is the selected number of layers in the Inception V3 network. In this experiment, we empirically choose five activation layers of the Inception V3 network as F to calculate $\mathcal{L}_{\text{prp}}$.

With $\mathcal{L}_{\text{FM}}$ and $\mathcal{L}_{\text{prp}}$, we are able to keep both low-level image characteristics and high-level perceptual information. Combining these losses together, our final objective is expressed as

$$G_{\text{loss}} = \mathcal{L}_G(G,D) + \alpha\mathcal{L}_{\text{FM}}(G,D) + \beta\mathcal{L}_{\text{prp}}(G) \quad D_{\text{loss}} = \mathcal{L}_D(G,D), \tag{7}$$

where α and β control the weight of $\mathcal{L}_{\text{FM}}$ and $\mathcal{L}_{\text{prp}}$ with respect to $\mathcal{L}_{\text{cGAN}}$. Here we set $\alpha = 10$ and $\beta = 10$, recommended by Rana et al.[6]

3.4 *Training and Testing*

We deploy different training and testing scheme combinations to achieve better performance.

3.4.1 *Training*

We adopt three training schemes.

- *Training scheme A (see purple box in* Fig. 4). All HDR images were resized into 256 × 256 resolution, and TMOs were used to generate tone-mapped LDR images. The generated 748 HDR-LDR image pairs were used to train our adTMO.
- *Training scheme B (see blue box in* Fig. 4). This scheme required resizing the HDR images into 1024 × 1024 resolution and using TMOs to generate tone-mapped LDR images. The next step was to randomly crop the corresponding 256 × 256 resolution regions from HDR images and LDR images. We generated 23,936 HDR-LDR image pairs to train the adTMO.
- *Training scheme C*. The resized and cropped 256 × 256 resolution images were combined from training schemes A and B to provide all together 24,684 training pairs.

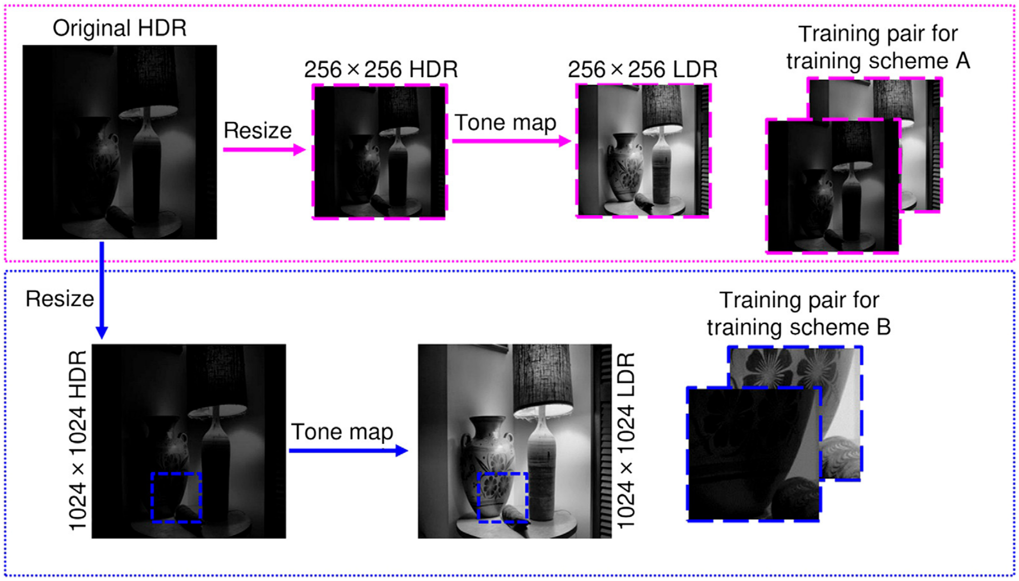


Fig. 4 The purple and blue boxes, respectively, show how we generate training pairs for training schemes A and B.

All training schemes used 256 × 256 resolution images as the training database, so the training process took less time and resources than using high-resolution images. The Adam optimizer[35] was used for all three schemes, with learing $\text{rate} = 0.0002, \beta_1 = 0.5, \beta_2 = 0.999$. We set the batch size to 1 and trained until the loss converged. The training process was deployed on an NVIDIA GeForce RTX 2080, and each training process can be finished within 30 h, which is much shorter than the 1-week training time in the muti-scale network propose by Rana et al.[6]

3.4.2 *Testing*

We deploy different testing schemes to evaluate the performance of our proposed adTMO.

- *Testing scheme W (see the red box of* Fig. 5). Test with resized 256 × 256 resolution images, we resized original HDR images into 256 × 256 resolution then fed them into G and generated the target LDR images.

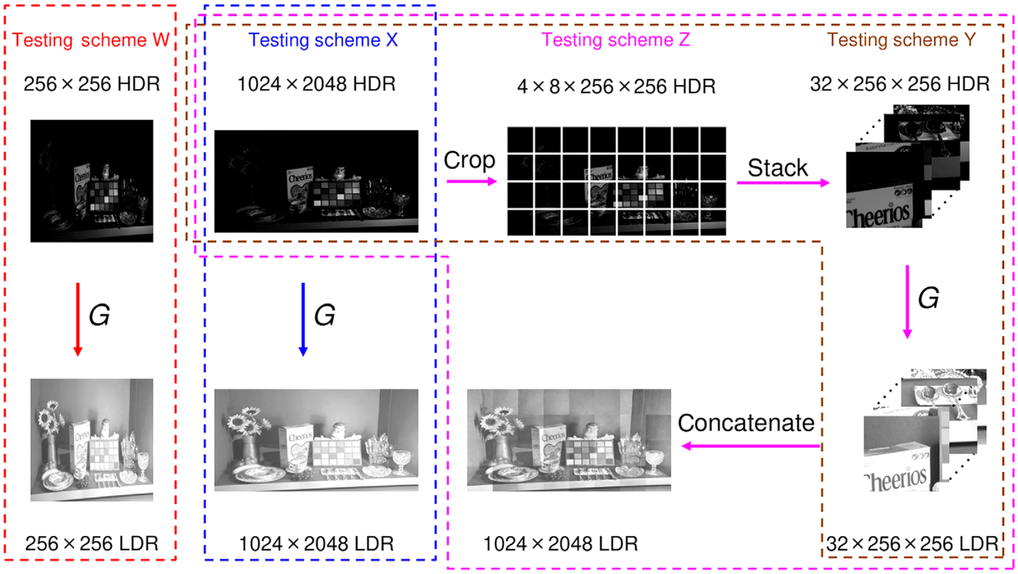


Fig. 5 The red, blue, brown, and purple boxes, respectively, show the process of test schemes W, X, Y, and Z.

- *Testing scheme X (see the blue box of* Fig. 5). Test with resized 1024 × 2048 resolution images, we resized original HDR images into 1024 × 2048 resolution then fed them into G and generated the target LDR images.
- *Testing scheme Y (see the brown box of* Fig. 5). Test with cropped 256 × 256 resolution images, we cropped 1024 × 2048 resolution HDR images into 256 × 256 resolution pieces, then fed them into G, and generated the target LDR pieces.
- *Testing scheme Z (see the purple box of* Fig. 5). Test with 4 × 8 concatenated cropped 256 × 256 resolution images, we cropped 1024 × 2048 resolution HDR images into 32 256 × 256 resolution pieces, fed them into G and generated the target LDR images, and then concatenated them together into the complete 1024 × 2048 resolution images.

4 Experimental Setup

In this section, we will detail the HDR image databases collected, how we pre- and postprocessed these databases.

4.1 *Databases*

From the many open-source HDR image databases accessible online, we selected our databases based on their content diversity, usability, resolution, and quality. Table 1 summarizes the HDR image databases we used, with the majority being high-resolution. We used 105 images from Kalantari and Ramamoorthi[45] to test adTMO, and 748 images from other 10 databases in Table 1 to train adTMO.

4.2 *Resizing*

We used two collections of 256 × 256 resolution images for training. The first set of images were the original images resized to 256 × 256 resolution (based on training scheme A), whereas the second set of images were randomly cropped from resized 1024 × 1024 images (based on training scheme B). For testing purpose, we resized HDR images into two resolutions: 256 × 256 and 1024 × 2048.

4.3 *Target LDR Images Generation*

All the collected HDR images were unlabeled, i.e., the ground-truth LDR images were unknown. To solve this problem, for each HDR image, we applied 30 different TMOs to get 30 LDR image candidates using the MATLAB HDR TOOLBOX[46] and followed the suggestion to apply GammaTMO after tone-mapping as some specific TMOs require gamma encoding. From these 30 LDR image candidates, we selected the one with the highest TMQI as the ground-truth LDR image. Table 2 summarizes the performance of each TMO when applied to the resized 256 × 256 HDR images. In Table 2, we provide the average TMQI for each TMO after applying it to the whole training set, and the number of LDR images with the highest TMQI among 30 candidates.

Table 1 HDR image databases.

Databases	# Images	# Pixels per image ($\times 10^6$)	Databases	# Images	# Pixels per image ($\times 10^6$)
Ref. 36	88	0.5	Ref. 37	92	1.8
Ref. 28	26	0.6	Ref. 38	44	14.5
Ref. 39	224	3.2	Ref. 40	15	0.3
Ref. 41	64	1.5	Ref. 42	8	2.9
Ref. 43	7	2.4	Ref. 44	180	12.9
Ref. 45	105	11.1			

Table 2 TMOs performance in tone-mapping 256×256 HDR images.

TMOs	TMQI	# LDR images with highest TMQI	TMOs	TMQI	# LDR images with highest TMQI
AshikhminTMO	0.83 ± 0.07	23	BanterleTMO	0.89 ± 0.04	24
BestExposureTMO	0.88 ± 0.05	12	BruceExpoBlend TMO	0.85 ± 0.06	9
ChiuTMO	0.86 ± 0.06	28	DragoTMO	0.89 ± 0.04	18
DurandTMO	0.87 ± 0.07	39	ExponentialTMO	0.84 ± 0.03	1
FerwerdaTMO	0.80 ± 0.11	13	GammaTMO	0.75 ± 0.15	15
KimKautzConsistent TMO	0.90 ± 0.05	37	KrawczykTMO	0.88 ± 0.07	30
KuangTMO	0.90 ± 0.05	25	LischinskiTMO	0.93 ± 0.04	89
LogarithmicTMO	0.82 ± 0.07	18	MertensTMO	0.83 ± 0.06	5
NormalizeTMO	0.88 ± 0.07	19	PattanaikTMO	0.73 ± 0.09	1
RamanTMO	0.80 ± 0.05	0	ReinhardDevlinTMO	0.86 ± 0.07	17
ReinhardTMO	0.92 ± 0.04	60	SchlickTMO	0.77 ± 0.10	2
TumblinTMO	0.83 ± 0.08	16	VanHaterenTMO	0.76 ± 0.09	2
WardGlobalTMO	0.81 ± 0.08	6	WardHistAdjTMO	0.92 ± 0.04	124
YPFerwerdaTMO	0.86 ± 0.06	38	YPTumblinTMO	0.80 ± 0.07	6
MATLAB tonemap function	0.89 ± 0.05	75			
target LDR images	0.96 ± 0.02	748			

The last row tabulates the average TMQI of the selected 748 target LDR images. Among the TMOs provided by the MATLAB HDR TOOLBOX, WardHistAdjTMO reaches the highest average TMQI and provides the most ground-truth LDR images (124 images). Apart from RamanTMO, which contributed 0 ground-truth images, all other TMOs provide at least one image for the ground-truth set.

This approach to generate target LDR images is similar to the one proposed by Cai et al.[47] to generate high-contrast images. Both our work and theirs aim to reproduce satisfactory natural LDR images. Although we focus on keeping the structural similarity from the HDR images and retaining the color naturalness, Cai et al. aimed to produce a high-contrast image from an under-/over-exposed image. Difference also exists in how to select the "ground-truth" target image. We use an objective metric TMQI to select a ground-truth LDR image, whereas Cai et al. used a subjective ranking to select a ground-truth high-contrast image.

4.4 *Normalization*

We linearly normalized the pixel value of input HDR and LDR images into [0, 1]. For input HDR images, the min/max normalization was applied:

$$v_{\text{out}} = \frac{v_{\text{in}} - v_{\min}}{v_{\max} - v_{\min}}, \tag{8}$$

where $v_{\max}$ and $v_{\min}$ are the maximum and minimum pixel values of the input HDR image, respectively. For input LDR image, we applied $v_{\text{out}} = v_{\text{in}}/255$ to do the normalization so that the pixel values of input LDR image are also in the range of [0, 1].

4.5 *Luminance Extraction and Color Reproduction*

When training and testing our proposed adTMO, we used the luminance channel rather than the RGB channels of the input images to ease the computation complexity and reduce the memory requirement. Before training, we calculated the weighted sum of the RGB channels to extract the luminance channel with the weights from Ref. 6:

$$L = 0.2959 \cdot C_R + 0.5870 \cdot C_G + 0.1140 \cdot C_B. \tag{9}$$

After generating the luminance channel from G, we used $C_{\text{out}} = C_{\text{in}} \cdot L_{\text{out}}/L_{\text{in}}$ to reproduce the RGB channels, where L_{in} and L_{out} are the input and output luminance channels, respectively, and C_{in} and C_{out} are the RGB channels of the original HDR image and the generated LDR image after color reproduction. After color reproduction, some pixel values would be larger than 255 and they were reduced to 255 to maintain the 8-bit RGB range.

5 Results

In this section, we discuss the results of our proposed adTMO, in terms of multiple metrics of the generated LDR images in different training/testing schemes.

Figure 6 demonstrates one scenario of LDR content in the RGB channels after color reproduction, in different training/testing schemes. We omit the generated LDR content in testing scheme Y because they were the images used for constructing the images in testing scheme Z. LDR images in testing scheme W [(a), (d), and (g)] have higher TMQI, but such conversion is meaningless, as many details are lost in the resizing operation. LDR images of testing scheme X, Z in training scheme A [(b), (c)] have lower TMQI with shadows around the flowers, because we only trained adTMO with resized 256 × 256 images so that many fine details from the

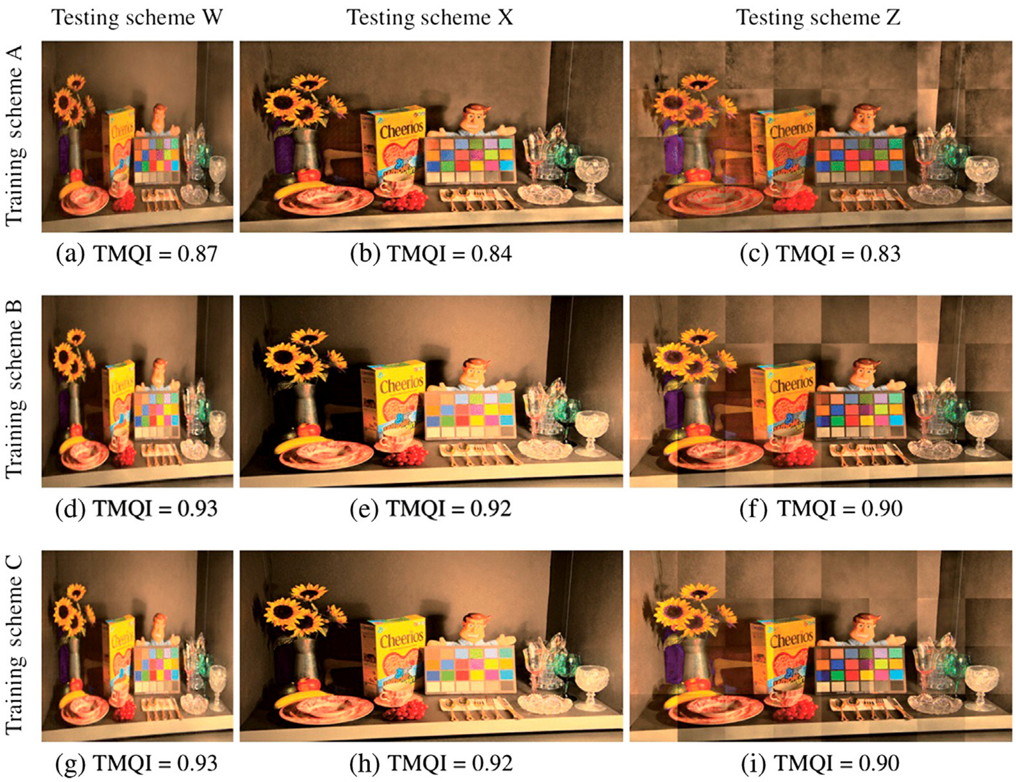
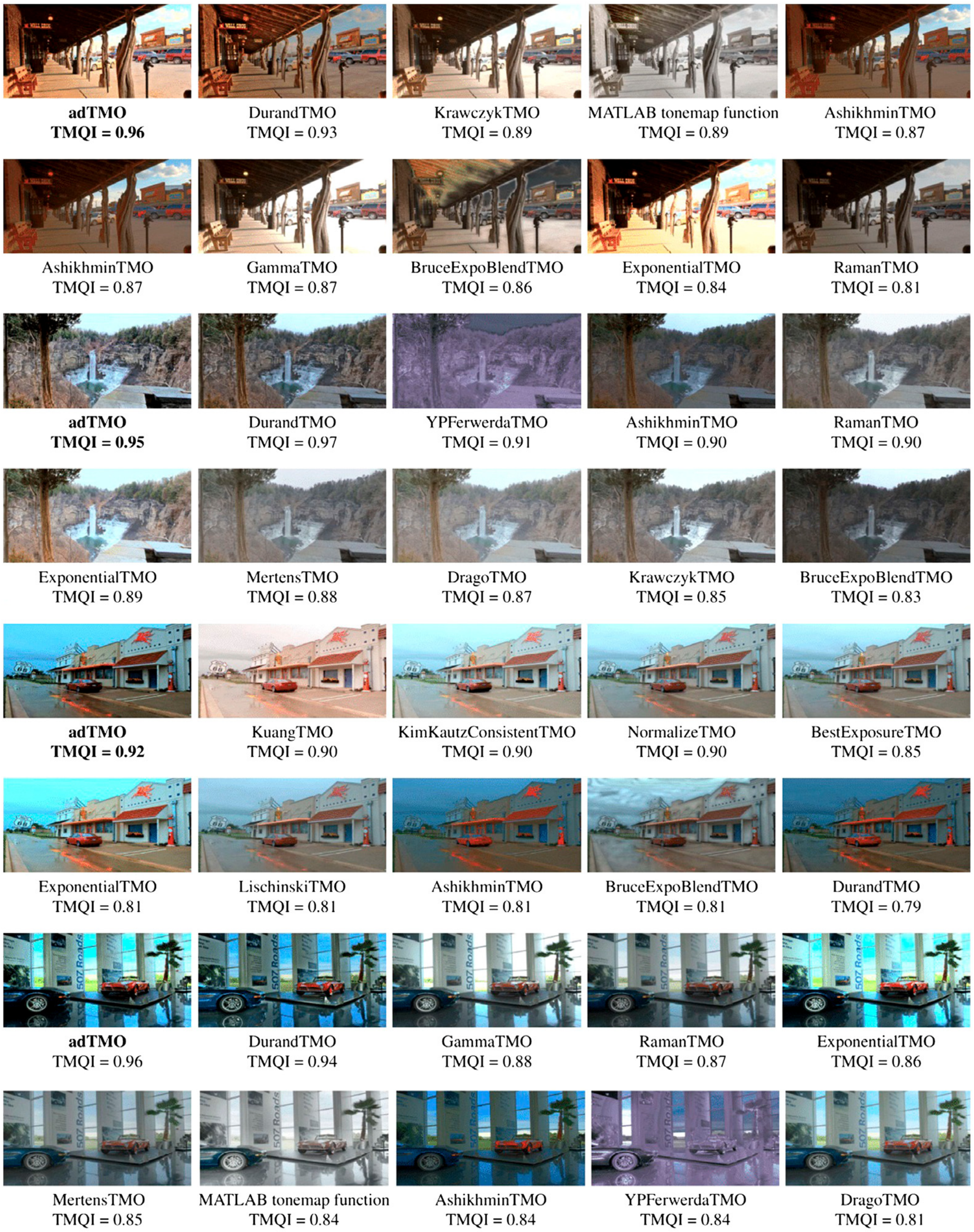


Fig. 6 The RGB channels of LDR images generated by adTMO after color reproduction. (a)–(c) are based on training scheme A; (d)–(f) are based on training scheme B; (g)–(i) are based on training scheme C. (a), (d), (g)are based on testing scheme W; (b), (e), (h) are based on testing scheme X; and (c), (f), (i) are based on testing scheme Z.

original images were lost. After we add cropped images into training databases, adTMO was able to learn how to keep the details of the original images. Therefore, the LDR images of testing scheme X in training scheme B, C [(e) and (h)] look more natural and have higher TMQI. The LDR images of testing scheme Z [(c), (f), and (i)] show "concatenated" edges, because cropping a complete image into pieces and generating their tone-mapped LDR images individually break the internal connections between these pieces. Future work is required to generate these individual images and combine them in such a way that these edges are removed while maintaining the high contrast in each individual image. Some finer details are not kept well by using the proposed adTMO. It should be noted that edge-preserving techniques such as bilateral filtering or guided image filtering have shown great promise in alleviating this problem. Further experimentation is required, and we plan in the future to incorporate these techniques into a deep-learning based TMO to create a more robust operator.



adTMO
TMQI = 0.96 | DurandTMO TMQI = 0.93 | KrawczykTMO TMQI = 0.89 | MATLAB tonemap function TMQI = 0.89 | AshikhminTMO TMQI = 0.87

AshikhminTMO TMQI = 0.87 | GammaTMO TMQI = 0.87 | BruceExpoBlendTMO TMQI = 0.86 | ExponentialTMO TMQI = 0.84 | RamanTMO TMQI = 0.81

adTMO TMQI = 0.95 | DurandTMO TMQI = 0.97 | YPFerwerdaTMO TMQI = 0.91 | AshikhminTMO TMQI = 0.90 | RamanTMO TMQI = 0.90

ExponentialTMO TMQI = 0.89 | MertensTMO TMQI = 0.88 | DragoTMO TMQI = 0.87 | KrawczykTMO TMQI = 0.85 | BruceExpoBlendTMO TMQI = 0.83

adTMO TMQI = 0.92 | KuangTMO TMQI = 0.90 | KimKautzConsistentTMO TMQI = 0.90 | NormalizeTMO TMQI = 0.90 | BestExposureTMO TMQI = 0.85

ExponentialTMO TMQI = 0.81 | LischinskiTMO TMQI = 0.81 | AshikhminTMO TMQI = 0.81 | BruceExpoBlendTMO TMQI = 0.81 | DurandTMO TMQI = 0.79

adTMO TMQI = 0.96 | DurandTMO TMQI = 0.94 | GammaTMO TMQI = 0.88 | RamanTMO TMQI = 0.87 | ExponentialTMO TMQI = 0.86

MertensTMO TMQI = 0.85 | MATLAB tonemap function TMQI = 0.84 | AshikhminTMO TMQI = 0.84 | YPFerwerdaTMO TMQI = 0.84 | DragoTMO TMQI = 0.81

Fig. 7 Qualitative comparisons of adTMO and top-9-ranked TMOs for outdoor and indoor scenes on TMQI metric.

We chose training scheme C to train the proposed adTMO, testing scheme W to tone-map 256 × 256 resolution images and testing scheme X to tone-map 1024 × 2048 resolution images given that train scheme C has the larger data set for training, and the resulting LDR images [(g) and (h)] have higher TMQI.

In Fig. 7, we demonstrate qualitative comparisons of adTMO and other top-9-ranked TMOs that produce the highest TMQI for four different scenarios, in generating 1024 × 2048 resolution

Table 3 Qualitative comparisons of adTMO and all other TMOs for 256 × 256 resolution images on SSIM, MS-SSIM, FID, and PPL metrics. The bold values indicate the metric where adTMO performs the best amongst all other TMOs.

TMOs	TMQI	SSIM	MS-SSIM	FID	PPL
AshikhminTMO	0.85 ± 0.07	0.71	0.73	103.2	327.5
BanterleTMO	0.89 ± 0.05	0.72	0.73	91.3	242.6
BestExposureTMO	0.90 ± 0.05	0.81	0.82	92.7	210.4
BruceExpoBlendTMO	0.88 ± 0.07	0.78	0.81	87.4	154.5
ChiuTMO	0.86 ± 0.06	0.71	0.75	98.0	201.7
DragoTMO	0.89 ± 0.05	0.76	0.78	93.4	296.1
DurandTMO	0.90 ± 0.06	0.78	0.79	88.3	164.1
ExponentialTMO	0.84 ± 0.04	0.73	0.76	121.9	219.5
FerwerdaTMO	0.84 ± 0.09	0.75	0.77	108.3	285.1
GammaTMO	0.80 ± 0.07	0.62	0.68	118.4	439.5
KimKautzConsistentTMO	0.90 ± 0.05	0.78	0.78	84.2	138.6
KrawczykTMO	0.86 ± 0.08	0.70	0.72	104.7	248.6
KuangTMO	0.89 ± 0.06	0.78	0.79	94.5	238.5
LischinskiTMO	0.93 ± 0.05	0.82	0.83	74.3	159.2
LogarithmicTMO	0.88 ± 0.07	0.76	0.78	98.2	223.8
MertensTMO	0.87 ± 0.06	0.71	0.73	96.2	194.4
NormalizeTMO	0.87 ± 0.08	0.73	0.76	101.4	245.8
PattanaikTMO	0.77 ± 0.02	0.60	0.63	164.9	468.1
RamanTMO	0.85 ± 0.07	0.69	0.71	116.7	280.2
ReinhardDevlinTMO	0.84 ± 0.04	0.71	0.72	113.8	202.7
ReinhardTMO	0.92 ± 0.05	0.80	0.81	80.5	143.8
SchlickTMO	0.84 ± 0.09	0.70	0.72	104.6	257.3
TumblinTMO	0.86 ± 0.04	0.70	0.72	108.5	236.1
VanHaterenTMO	0.82 ± 0.04	0.68	0.70	115.7	275.9
WardGlobalTMO	0.89 ± 0.06	0.80	0.81	92.5	193.6
WardHistAdjTMO	0.93 ± 0.04	0.80	0.81	70.3	152.9
YPFerwerdaTMO	0.86 ± 0.06	0.72	0.74	98.2	204.2
YPTumblinTMO	0.81 ± 0.03	0.68	0.71	102.5	257.4
YPWardGlobalTMO	0.87 ± 0.06	0.71	0.74	98.4	201.5
MATLAB tonemap function	0.87 ± 0.04	0.74	0.76	129.5	286.3
Proposed adTMO	0.92 ± 0.05	0.80	0.82	**68.2**	163.2

images. In most scenarios, including indoor/outdoor, irregular geometric shape, large colors range, and drastic luminance changes, our adTMO outperforms all other TMOs on TMQI metric. As well, the LDR images generated by adTMO do not suffer contrast problems like other LDR images. Tables 3 and 4 list different metrics mentioned in Sec. 2 of the test dataset tone-mapped

Table 4 Qualitative comparisons of adTMO and all other TMOs for 1024 × 2048 resolution images on SSIM, MS-SSIM, FID, PPL, and face detection accuracy metrics. The bold values indicate the metric where adTMO performs the best amongst all other TMOs.

TMOs	TMQI	SSIM	MS-SSIM	FID	PPL	Face detection acc. (%)
AshikhminTMO	0.82 ± 0.09	0.69	0.70	114.6	254.8	70.5
BanterleTMO	0.84 ± 0.08	0.67	0.69	104.5	239.6	87.6
BestExposureTMO	0.85 ± 0.07	0.73	0.73	102.4	218.5	88.6
BruceExpoBlendTMO	0.81 ± 0.07	0.70	0.71	96.5	204.5	83.8
ChiuTMO	0.78 ± 0.06	0.64	0.68	104.9	208.7	78.1
DragoTMO	0.84 ± 0.07	0.69	0.71	98.6	175.3	85.7
DurandTMO	0.89 ± 0.07	0.75	0.77	104.7	264.9	87.6
ExponentialTMO	0.83 ± 0.05	0.70	0.71	142.7	304.6	73.3
FerwerdaTMO	0.76 ± 0.11	0.70	0.72	123.8	175.0	70.5
GammaTMO	0.78 ± 0.08	0.61	0.66	121.5	275.1	73.3
KimKautzConsistentTMO	0.85 ± 0.07	0.75	0.76	97.4	204.6	81.9
KrawczykTMO	0.81 ± 0.10	0.68	0.69	119.5	259.0	80.0
KuangTMO	0.85 ± 0.08	0.72	0.74	101.3	237.1	81.0
LischinskiTMO	0.89 ± 0.07	0.80	0.81	87.5	174.2	88.6
LogarithmicTMO	0.82 ± 0.08	0.72	0.74	103.9	222.5	80.0
MertensTMO	0.84 ± 0.08	0.68	0.71	99.4	194.9	77.1
NormalizeTMO	0.82 ± 0.09	0.68	0.70	105.4	223.7	78.1
PattanaikTMO	0.70 ± 0.06	0.58	0.61	195.8	479.1	4.8
RamanTMO	0.82 ± 0.08	0.64	0.66	124.7	275.9	77.1
ReinhardDevlinTMO	0.79 ± 0.05	0.69	0.70	115.3	246.1	76.2
ReinhardTMO	0.86 ± 0.07	0.75	0.76	87.1	196.4	87.6
SchlickTMO	0.79 ± 0.08	0.66	0.68	119.6	219.4	75.2
TumblinTMO	0.80 ± 0.06	0.67	0.69	125.2	231.8	78.1
VanHaterenTMO	0.77 ± 0.06	0.62	0.65	128.5	274.0	76.2
WardGlobalTMO	0.82 ± 0.07	0.75	0.76	96.3	162.4	81.9
WardHistAdjTMO	0.89 ± 0.06	0.76	0.76	77.5	186.5	83.8
YPFerwerdaTMO	0.86 ± 0.08	0.71	0.73	107.5	201.4	76.2
YPTumblinTMO	0.75 ± 0.05	0.66	0.67	111.8	214.8	70.5
YPWardGlobalTMO	0.80 ± 0.06	0.64	0.67	105.7	197.5	81.0
MATLAB tonemap function	0.84 ± 0.06	0.70	0.72	141.6	308.1	88.6
Proposed adTMO	0.90 ± 0.06	**0.80**	**0.81**	79.5	187.4	**90.5**

by 30 TMOs and the proposed adTMO. We modify the PPL so that it can be used to evaluate TMOs. Specifically, the PPL is calculated as follows:

$$\text{PPL} = \mathbb{E}[\frac{1}{\epsilon^2} d(g\{lerp[f(z_1), f(z_2); t]\}, g\{lerp[f(z_1), f(z_2); t + \epsilon]\})]. \tag{10}$$

where $f(z)$ represent the function mapping latent space to style vector in adTMO, t is uniformly distributed between 0 and 1, $lerp$ represents for linear interpolation, g is the generator function to create image, d measures the perceptual distance between the images, and ϵ is set as 10^{-4} here. In generating 256×256 resolution images, our proposed adTMO outperforms all other TMOs with regard to the metric FID and outperforms most of TMOs with regard to other metrics. In generating 1024×2048 resolution images, our proposed adTMO outperforms all other TMOs with regard to the metrics TMQI, SSIM, and MS-SSIM and outperforms most other TMOs with regard to FID and PPL. We also divided the images into two sets, one for indoor scenes and another for outdoor scenes. Both reach high TMQI (0.89 and 0.90) for 1024×2048 resolution images. Our deep learning-based tone mapping algorithm uses a mixture of best features from other TMOs. In the absence of interactive parameter adjustment as it is not always available, our approach offers the best TMQI.

In addition to the above-mentioned metrics, we also applied a face detection technique to the generated 1024×2048 LDR images to measure the face detection accuracy as HDR-LDR translation is often used in security and healthcare applications. The face detection accuracy is defined as $\text{acc} = \text{TP}/(\text{TP} + \text{FN})$, where TP and FN represent the number of faces that are detected and not detected, respectively. The face detector used in this paper is the Haar cascades face detector,[48] and the test set we used for evaluation is by Kalantari and Ramamoorthi,[45] which consists of HDR images containing human faces. Our proposed adTMO reaches the highest face detection accuracy compared with other TMOs. The main reason contributing to this is that we use the pretrained Inception V3 network[24] to derive the perceptual loss, so our generated LDR images look more natural, and the face detector trained on natural images can achieve higher accuracy in LDR images generated by our adTMO. Overall, adTMO output has the highest quality, regarding high-resolution 1024×2048 images and is comparable to the results for 256×256 images.

6 Conclusion

We propose an adTMO, which can adaptively generate high-resolution and high-quality LDR images. We explore different training and testing schemes and find the best possible combination to generate the highest quality images. We use multiple metrics including TMQI, SSIM, MS-SSIM, and face detection accuracy to measure the performance of the proposed adTMO. When testing on low-resolution LDR images, our adTMO has the highest performance on the FID metric across all other TMOs. When testing on high-resolution LDR images, our adTMO has the highest performance on TMQI, SSIM, MS-SSIM, and face detection accuracy over all other TMOs. Looking specifically at the TMQI metric, the proposed adTMO achieves a TMQI of 0.90 ± 0.06, which is superior to the DeepTMO's[6] 0.88 ± 0.06. In addition, we have the advantage in the training time where we spend 30 h for training, which is much short than DeepTMO's 1 week.

Acknowledgments

This project was partially supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) through the grant "Biometric-enabled Identity management and Risk Assessment for Smart Cities," and the Mitacs Globalink Graduate Fellowship, Canada.

References

1. F. Mccollough, *Complete Guide to High Dynamic Range Digital Photography*, Lark Books (2008).

2. E. Reinhard et al., *High Dynamic Range Imaging: Acquisition, Display, and Image-Based Lighting*, Morgan Kaufmann (2010).
3. M. Mirza and S. Osindero, "Conditional generative adversarial nets," arXiv:1411.1784 (2014).
4. P. Isola et al., "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 1125–1134 (2017).
5. T.-C. Wang et al., "High-resolution image synthesis and semantic manipulation with conditional GANs," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 8798–8807 (2018).
6. A. Rana et al., "Deep tone mapping operator for high dynamic range images," *IEEE Trans. Image Process.* **29**, 1285–1298 (2019).
7. X. Cao et al., "Adversarial and adaptive tone mapping operator for high dynamic range images," in *Proc. IEEE Symp. Ser. Comput. Intell. (SSCI)*, pp. 1814–1821 (2020).
8. G. W. Larson, H. Rushmeier, and C. Piatko, "A visibility matching tone reproduction operator for high dynamic range scenes," *IEEE Trans. Vis. Comput. Graphics* **3**(4), 291–306 (1997).
9. F. Drago et al., "Adaptive logarithmic mapping for displaying high contrast scenes," *Proc. Comput. Graphics Forum* **22**(3), 419–426 (2003).
10. K. Chiu et al., "Spatially nonuniform scaling functions for high contrast images," in *Proc. Graphics Interface*, Canadian Information Processing Society, pp. 245–245 (1993).
11. J. Tumblin, J. K. Hodgins, and B. K. Guenter, "Two methods for display of high contrast images," *ACM Trans. Graphics* **18**(1), 56–94 (1999).
12. I. Goodfellow et al., "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, pp. 2672–2680 (2014).
13. V. A. Patel, P. Shah, and S. Raman, "A generative adversarial network for tone mapping HDR images," in *Proc. Conf. Comput. Vision, Pattern Recognit., Image Process. and Graphics*, Springer, pp. 220–231 (2017).
14. C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Sixth Int. Conf. Comput. Vision (IEEE Cat. No. 98CH36271)*, IEEE, pp. 839–846 (1998).
15. F. Porikli, "Constant time o (1) bilateral filtering," in *IEEE Conf. Comput. Vision and Pattern Recognit.*, IEEE, pp. 1–8 (2008).
16. Z. Farbman et al., "Edge-preserving decompositions for multi-scale tone and detail manipulation," *ACM Trans. Graphics* **27**(3), 1–10 (2008).
17. K. He, J. Sun, and X. Tang, "Guided image filtering," in *Comput. Vision-ECCV 2010*, K. Daniilidis, P. Maragos, and N. Paragios, Eds., Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 1–14 (2010).
18. D. Min et al., "Fast global image smoothing based on weighted least squares," *IEEE Trans. Image Process.* **23**(12), 5638–5653 (2014).
19. Z. Li et al., "Weighted guided image filtering," *IEEE Trans. Image Process.* **24**, 120–129 (2015).
20. C. Zheng et al., "Single image brightening via multi-scale exposure fusion with hybrid learning," *IEEE Trans. Circuits Syst. Video Technol.* **31**(4), 1425–1435 (2020).
21. Z. Wang et al., "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Process.* **13**(4), 600–612 (2004).
22. Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *Proc. Thirty-Seventh Asilomar Conf. Signals, Syst. & Comput.s*, Vol. 2, pp. 1398–1402 (2003).
23. M. Heusel et al., "GANs trained by a two time-scale update rule converge to a local nash equilibrium," in *Proc. Adv. Neural Inf. Process. Syst.*, pp. 6626–6637 (2017).
24. C. Szegedy et al., "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 2818–2826 (2016).
25. T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 4401–4410 (2019).
26. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv:1409.1556 (2014).

27. P. Ledda et al., "Evaluation of tone mapping operators using a high dynamic range display," *ACM Trans. Graphics* **24**(3), 640–648 (2005).
28. H. Yeganeh and Z. Wang, "Objective quality assessment of tone-mapped images," *IEEE Trans. Image Process.* **22**(2), 657–667 (2012).
29. K. Ma et al., "High dynamic range image compression by optimizing tone mapped image quality index," *IEEE Trans. Image Process.* **24**(10), 3086–3097 (2015).
30. K. Debattista, "Application-specific tone mapping via genetic programming," *Proc. Comput. Graphics Forum* **37**(1), 439–450 (2018).
31. O. Ronneberger, P. Fischer, and T. Brox, "U-net: convolutional networks for biomedical image segmentation," *Lect. Notes Comput. Sci.* **9351**, 234–241 (2015).
32. C. Li and M. Wand, "Precomputed real-time texture synthesis with markovian generative adversarial networks," in *Proc. Eur. Conf. Comput. Vision*, Springer, pp. 702–716 (2016).
33. A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," arXiv:1511.06434 (2015).
34. J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. Eur. Conf. Comput. Vision*, Springer, pp. 694–711 (2016).
35. D. P. Kingma and J. Ba, "Adam: a method for stochastic optimization," arXiv:1412.6980 (2014).
36. F. Xiao et al., "High dynamic range imaging of natural scenes," in *Proc. Color and Imaging Conf.*, Society for Imaging Science and Technology, pp. 337–342 (2002).
37. M.-A. Gardner et al., "Learning to predict indoor illumination from a single image," arXiv:1704.00090 (2017).
38. P. Stanczyk and C. Phillips, "openexr images," 2020, https://github.com/AcademySoftwareFoundation/openexr-images.
39. B. Funt and L. Shi, "The rehabilitation of maxRGB," in *Proc. Color and Imaging Conf.*, Society for Imaging Science and Technology, pp. 256–259 (2010).
40. P. Modin, "HDR Vault Image Set (Version 1.0.0). Zenodo," https://zenodo.org/record/1245790#.YRazFYgzY2w (2018).
41. M. D. Fairchild, "The HDR photographic survey," in *Proc. Color and Imaging Conf.*, Society for Imaging Science and Technology, pp. 233–238 (2007).
42. Pfstools Google Group, "Pfstools HDR image gallery," http://pfstools.sourceforge.net.
43. Max Planck Institut Informatik, "HDR source image gallery," 2018, http://resources.mpi-inf.mpg.de/hdr/gallery.html.
44. W. J. Adams et al., "The southampton-york natural scenes (SYNS) dataset: statistics of surface attitude," *Sci. Rep.* **6**, 35805 (2016).
45. N. K. Kalantari and R. Ramamoorthi, "Deep high dynamic range imaging of dynamic scenes," *ACM Trans. Graphics* **36**(4), 1–12 (2017).
46. F. Banterle et al., *Advanced High Dynamic Range Imaging*, CRC Press (2017).
47. J. Cai, S. Gu, and L. Zhang, "Learning a deep single image contrast enhancer from multi-exposure images," *IEEE Trans. Image Process.* **27**(4), 2049–2062 (2018).
48. P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recognit.*, Vol. 1, p. I–I (2001).

Xingdong Cao received his BSc degree in electrical engineering from Zhejiang University, Zhejiang, China, in 2019. He is currently an MSc student under the supervision of professor Svetlana Yanushkevich at the Biometric Technologies Laboratory, Department of Electrical and Software Engineering, University of Calgary, Calgary, Alberta, Canada. His research interests include applying machine learning technologies to the biometrics field.

Kenneth Lai received his BSc and MSc degrees from the University of Calgary, Calgary, Alberta, Canada, in 2012 and 2015, respectively, where he is currently pursuing his PhD in the Department of Electrical and Software Engineering. His areas of interest include biometrics and its application to security and health care systems.

Michael Smith is a professor emeritus in electrical and software engineering at Schulich School of Engineering, University of Calgary, Calgary, Canada, with research interests in software engineering and customized real-time digital signal processing algorithms in the context of mobile embedded systems and biomedical instrumentation. He is a senior member of IEEE.

Svetlana Yanushkevich received her Dr.Tech.Sc. (Dr. Habilitated) degree from the Warsaw University of Technology in 1999. She is currently a professor in the Department of Electrical and Software Engineering at the University of Calgary. She is directing the Biometric Technologies Laboratory and conducting research in the area of biometric-based authentication technologies. She is a senior member of IEEE.