Deep Tone Mapping Operator for High Dynamic Range Images

Aakanksha A Rana, Praveer Singh, Giuseppe Valenzise, Frédéric Dufaux, Nikos Komodakis, Aljosa Smolic

Deep Tone Mapping Operator for High Dynamic Range Images

Aakanksha Rana*, Praveer Singh*, Giuseppe Valenzise, Frederic Dufaux, Nikos Komodakis, Aljosa Smolic

***Abstract*—A computationally fast tone mapping operator (TMO) that can quickly adapt to a wide spectrum of high dynamic range (HDR) content is quintessential for visualization on varied low dynamic range (LDR) output devices such as movie screens or standard displays. Existing TMOs can successfully tone-map only a limited number of HDR content and require an extensive parameter tuning to yield the best subjective-quality tone-mapped output. In this paper, we address this problem by proposing a fast, parameter-free and scene-adaptable deep tone mapping operator (DeepTMO) that yields a high-resolution and high-subjective quality tone mapped output. Based on conditional generative adversarial network (cGAN), DeepTMO not only learns to adapt to vast scenic-content (*e.g.*, outdoor, indoor, human, structures, etc.) but also tackles the HDR related scene-specific challenges such as contrast and brightness, while preserving the fine-grained details. We explore 4 possible combinations of Generator-Discriminator architectural designs to specifically address some prominent issues in HDR related deep-learning frameworks like blurring, tiling patterns and saturation artifacts. By exploring different influences of scales, loss-functions and normalization layers under a cGAN setting, we conclude with adopting a multi-scale model for our task. To further leverage on the large-scale availability of unlabeled HDR data, we train our network by generating *targets* using an objective HDR quality metric, namely Tone Mapping Image Quality Index (TMQI). We demonstrate results both quantitatively and qualitatively, and showcase that our DeepTMO generates high-resolution, high-quality output images over a large spectrum of real-world scenes. Finally, we evaluate the perceived quality of our results by conducting a pair-wise subjective study which confirms the versatility of our method.**

***Index Terms*—High Dyanmic Range images, tone mapping, generative adversarial networks.**

I. Introduction

Tone mapping is a prerequisite in the high dynamic range (HDR) imaging [1], [2], [3], [4] pipeline to print or render HDR content for low dynamic range displays. With the unprecedented demands of capturing/reproducing scenes in high-resolution and superior quality, HDR technology is growing rapidly [5], [6], [7]. Although HDR display systems have advanced in the last few decades (for *e.g.*, Sim2, Dolby Vision, etc), they still necessitate some sort of tone mapping operation because of limited technical capabilities of the materials used in these displays. Additionally, due to high manufacturing costs, the absolute majority of screens still have limited dynamic range and rely largely on Tone Mapping Operators (TMOs) for desired top-quality presentation.

Several TMOs have been designed over the last two decades, promising the most faithful representation of real-world luminosity and color gamut for high-quality output. However, in practice, such TMOs are limited to successfully tone map only limited number of HDR images due to their parametric sensitivity [8], [9]. For instance, a TMO capable of mapping a bright daytime scene might not map a dark or evening scene equally well. In fact, one needs to manually tweak in an extensive parametric space for every new scene, in order to achieve the best possible results while using any such TMO. Thus, the entire process of finding the most desirable high-resolution tone-mapped output is not only slow, tedious and expensive, but is almost impractical when there is a large variety of HDR content being generated from numerous capturing devices.

This raises a natural question whether a more *adaptive* tone mapping function can be formulated which can quickly alter itself to wide variability in real-world HDR scenes to reproduce the best subjective quality output without any perceptual damage to its content on a high-resolution display. With the recent success of deep learning [10] and wide scale availability of HDR data, it is now possible to learn a model with such complex functionalities for effective tone mapping operation.

In this paper, we propose an end-to-end deep learning (DL) based tone-mapping operator (DeepTMO) for converting any given HDR scene into a tone-mapped LDR output which is of high resolution [1024x2048] and superior subjective quality. Based upon a conditional generative adversarial network (cGAN) [11], [12], the DeepTMO model directly inputs 32-bit *linear* HDR content and reproduces a realistically looking tone-mapped image, aiming to mimic the original HDR content under a limited range [0-255]. DeepTMO is trained to cater a wide range of scenic-content for *e.g.*, indoor/outdoor scenes, scenes with structures, human faces, landscapes, dark and noisy scenes, etc.

The motivation for generative adversarial networks (GAN) in the DeepTMO design stems from their tremendous success in several image-to-image translation studies [13]. Such models have shown to overcome the problem of spatially blurred-out resulting images with a simple L_1/L_2 loss function. Furthermore, instead of optimizing parameters for a given TMO [14] for a particular scene [15], [2], our objective is to

A. Rana and A. Smolic are with V-SENSE, Trinity College Dublin, Ireland
G. Valenzise and F. Dufaux are with Laboratoire des Signaux et Systémes, CNRS, CentraleSupelec, Université Paris-Sud.
P. Singh and N. Komodakis are with LIGM/IMAGINE, Ecole des Ponts ParisTech, Université Paris-Est.
*Equal contributors. This publication has emanated from research conducted with the financial support of Science Foundation Ireland (SFI) under the Grant Number 15/RP/2776. The work presented in this document was also supported by BPIFrance and Région Ile de France, in the framework of the FUI 18 Plein Phare project. We gratefully acknowledge the support of NVIDIA Corporation with the donated GPU used for this research.

design a model which is *adaptable* to different scenes-types (such as day/night, outdoor/indoor, etc.), thus encompassing all their desired characteristics. Altogether, this is difficult for a naive loss-function to satisfy. Moreover, designing such a cost function is quite complex [16], and needs expert knowledge. Therefore, we overcome this challenge by *learning* an 'adversarial loss' that encapsulates all the desired features from all ideal tone-mapped images by using the underlying training data; thereby eradicating the need of manually handcrafting such a loss function.

GANs are capable to generate better quality images compared to the state-of-the-art models, however, there are still some prominent issues such as tilling patterns, local blurring and saturated artifacts (see Fig. 5 (a)). To handle these problems in a high-resolution output image, we explore the DeepTMO architectural design by comparing the single-scale and multi-scale variants of both generator and discriminator. We subsequently showcase how a multi-scale version of the generator-discriminator architecture helps in predicting artifact-free tone mapped images, which are both structurally consistent with input HDR and simultaneously preserve fine-grained information recovered from different scales.

The DeepTMO model is effectively a multi-scale architecture having a 2-scale generator and a 2-scale discriminator, both of which are conditioned on the *linear* HDR input. Both generator and discriminator compete with each other. The generator is trying to fool discriminator by producing high subjective quality tone mapped images for the given input HDR, while the discriminator trying to discriminate between real and synthetically generated HDR-LDR image pairs. Our basic discriminator architecture is similar to PatchGAN [17], [18] which classifies patches over the entire image and averages over all of them to yield the final image score. Similarly our basic generator architecture comprises of an encoder-decoder network where the input HDR is given first to an encoder resulting in a compressed representation which is then passed to the decoder yielding finally a tone mapped image.

To train our model, we accumulate our dataset from freely available HDR image sources. Ideally, the training dataset should be created through a subjective evaluation considering all possible tone mapping operators for all available HDR scenes. However, conducting such a subjective evaluation is highly cumbersome and unfeasible. Thus, it necessitates the requirement of an objective quality assessment metric which can quantify the tone mapping performance of each TMO for any given scene. For our task, we select a well known metric namely Tone Mapped Image Quality Index (TMQI). We first rank 13 widely used TMOs using the TMQI metric for each HDR input. We then select the topmost scoring tone-mapped image as our *target* output.

In a nutshell, we

1) propose a fast, parameter-free DeepTMO, which can generate high-resolution and foremost subjective quality tone-mapped outputs for a large variety of *linear* HDR scenes, including indoor, outdoor, person, structures, day and night/noisy scenes.
2) explore 4 possible cGANs network settings: (a) Single-scale-Generator (Single-G) and Single-scale-Discriminator (Single-D), (b) Multi-scale-Generator (Multi-G) and Single-D, (c) Single-G and Multi-scale-Discriminator (Multi-D), (d) Multi-G and Multi-D, thus discussing the influence of scales and finally proposing a multi-scale generator-discriminator model for our problem.
3) detail the impact of different loss functions and normalization layers while elaborating how each step helps in improving the overall results by tackling different artifacts.
4) provide quantitative and qualitative comparison of our model with best tone mapped outputs over 105 images and also validate our technique through a pair-wise subjective study.

II. Related Work

HDR imaging technology has been a subject of interest over the past decades, inspiring to capture and reproduce a wide range of colors and luminous intensities of the real world on a digital canvas. Normally, the information stored in HDR content is represented using a 32-bit floating point format. But to cope with conventional displays, such scenes are often tone-mapped to an LDR format with available TMOs. A great variety of TMOs addressing different perceptual objectives have been proposed in the past years. In the following, we give a quick review of the tone mapping literature and then would touch upon various deep learning techniques for HDR imaging.

A. Tone Mapping Operators for HDR Content

TMOs have been widely explored in the literature, principally based upon how they handle the contrast, color and luminosity in a given HDR image [19]. However, they have been classified into several categories under different sets of criteria [6], [5]. Primarily, they have been grouped into *global* and *local* approaches, relying on how these mapping functions operate on an image. The global methods such as [20], [21], [22] apply the same compression function to all the pixels of an image. For the *local* techniques such as [23], [24], [25], a tone-mapped pixel depends on the values of its neighboring pixels. Even though global approaches are faster to compute, their resulting LDR outputs do not maintain adequate contrast in the images; thus the scene appears somewhat washed out. The local tone mapping functions, conversely, do not face these issues and are generally capable enough of handling contrast ratios, meanwhile preserving local details. However, these operators result in some prominent 'halo' effects around the high frequency edges, thereby giving unnatural artifacts in the scenes. Another category of TMOs [26], [27], [28] includes designs which are inspired from the human visual system, can models the attributes such as adaptation with time, and can discriminate at high contrast stimuli and gradient sensitivities. Nonetheless, all these existing TMOs have been designed to target independently, multiple different objectives [6], [9], such as simulating human visual properties, honest reproduction of scenes, best subjective preference or even for computer vision applications [29]. However, in our work, we mainly focus

towards designing a TMO aiming for "best subjective quality output".

Several small scale perceptual studies have been performed using varied criteria such as with reference or without reference [30], [8], [31] to compare these classical and newly developed TMOs for different perceptual objectives. Even though these subjective studies are ideal to analyze TMO's performance, the process is bounded to use a limited number of content and TMOs due to practical considerations. As an alternate solution, objective metrics such as [31], [32] have been proposed to automate the evaluation. TMQI is a state-of-the-art objective metric and has been widely used for several TMO optimization studies [2], [15]. It assesses the quality of images on 1) structural fidelity which is a multi scale analysis of the signals, and 2) naturalness, which is derived using the natural image statistics. Both these crucial properties of human perception are combined to define a subjective quality score.

a) Learning-based methods: Parametric sensitivity of hand-crafted TMOs is a well-known phenomenon which impacts the subjective quality of the resulting output. As a result, this emphasizes 'scene-dependence' of such tone mapping designs *i.e.*, for a given subjective quality task, TMOs have to be fine tuned for each individual scene type. To this end, some optimization based tone mapping frameworks [2], [15] have been designed where the parameters of a specific TMO are optimized for a given image. However, the parameter fine-tuning process for each scene separately is time consuming and limits its real-time applicability. Additionally, it somehow questions the 'automatic' nature of tone mapping [9] for their applicability on a wide variety of real-world scenes.

B. CNNs for HDR Scenes

Recently, CNNs have been utilized extensively for multiple HDR imaging tasks such as reconstructing HDR using a single-exposure LDR [33], [34], [35], [36], predicting and merging various high and low exposure images for HDR reconstruction [37] or yielding HDR outputs from dynamic LDR inputs [38]. CNNs have also been modeled to learn an input-output mapping as done for de-mosaicking and de-noising by [39] or learning an efficient bilateral grid for image enhancement [40]. [41] have recently proposed a deep bilateral tone mapper, but it works only for 16-bit linear images and not for conventional 32-bit HDR images. A recent work [42] addresses the end-to-end tone mapping problem where the model is trained for a given scene. This is somewhat similar approach to parameter-tuning where the model is calibrated for only one given scene at a time. Therefore, the problem of designing a fast, parameter-free, end-to-end TMO which can effectively tone map wide variety of real-world high-resolution content for high quality display in real time, still holds relevance.

As observed in the past CNN studies, the quality of resulting output depends heavily on the choice of the loss function. Formulating a loss function that constrains the CNN to yield sharp, top quality tone-mapped LDR from their corresponding linear-valued HDR is complex and an ill posed problem. Our work doesn't encounter such issues as we utilize a GAN based architecture.

C. Generative Adversarial Networks

GANs [11] have attended lots of attention owing to their capability of modeling the underlying target distribution by forcing the predicted outputs to be as indistinguishable from the target images as possible. While doing this, it implicitly learns an appropriate loss function, thus eliminating the requirement of hand crafting one by an expert. This property has enabled them to be utilized for wide variety of image processing tasks such as super-resolution [18], photo-realistic style-transfer [43] and semantic image in-painting [44]. For our task, we employ GAN under a conditional setting, commonly referred as cGAN [12], where the generated output is conditioned on the input image. Recently, cGAN based frameworks have been designed for the inverse problem of generating HDR images from single LDR images [45], [46].

One distinctive feature of cGAN frameworks is that they learn a structured loss where each output pixel is conditionally dependent on one or more neighboring pixels in the input image. Thus, this effectively constrains the network by penalizing any possible structural difference between input and output This property is quite useful for our task of tone-mapping where we only want to compress the dynamic range of an HDR image, keeping the structure of the output similar to the input HDR. For this specific reason, cGANs have been quite popular for image-to-image translation tasks, where one representation of a scene is automatically converted into another, given enough training pairs [13] or without them under unsupervised settings [47], [48], [49]. However, a major limitation of using cGANs is that it is quite hard to generate high resolution images due to training instability and optimization issues. The generated images are either blurry or contain noisy artifacts such as shown in Fig. 5 (a). In [50], motivated from perceptual loss [43], the authors derive a direct regression loss to generate high-resolution 2048×1024 images, but their method fails to preserve fine-details and textures. In [51], authors have recently shown significant improvement on the quality of high-resolution generated outputs through a multi-scale generator-discriminator design. A similar work for converting HDR to LDR using GANs [52] has also appeared recently where authors oversimplifies the tone-mapping problem by testing only on small 256x256 image crops. Essentially, such an approach may not substantially capture the full luminance range present in HDR images, and thereby overlooks the basic goal of TMO by working on full dynamic range scenes. We, however, showcase our findings using their adopted architectures from [13] on 1024×2048 HDR images in the supplementary material.

In summary, we motivate the DeepTMO design with these given findings, and discuss the impact of scales for both generator and discriminator, while showcasing their ability to generate high-resolution tone-mapped outputs.

III. Algorithm

A. Problem Formulation

We propose a fast DeepTMO model with the prime objective of producing high-resolution and high-quality tone-mapped images for vast variety of real-world HDR images.

Ideally, our model should automatically adapt for each scene without any external parameter tuning. To this end, we propose to imbibe different desired tone mapping characteristics depending upon scene type, content, brightness, contrast etc., to yield high perceptual quality output. In the following paragraphs, we will briefly discuss the formulation of our DeepTMO model.

a) Linear Domain Input: For our models, we directly work on linear values. We performed the scaling to [0,1] with very high-precision (32-bit float precision), thereby, not impacting the overall output brightness. This way, we could simply automate the entire pipeline by making the network learn itself from the unaltered high-dynamic information of the scene. Additionally, we also experimented with log-scaling the input HDR before performing the tone mapping operation, specifically to test for halo-effects in high-exposure regions such as the sun in Fig.15. Note that we did some experimental studies with different input normalization techniques. More details can be found in supplementary material.

b) Color Reproduction: Classical TMOs firstly perform the dynamic range compression in the luminance channel only and then, the colors are reproduced in the post-processing stage. This partially accounts to ease the computational complexity of the tone-mapping operation. We follow a similar paradigm, employing the common methodology for color reproduction [22] given as $C_{out} = \frac{C_{in}}{L_{in}} \cdot L_{out}$, where C_{out} and L_{out} are output color and luminance images while C_{in} is the input HDR color image.

(a) $\mathcal{L}_1$-loss (b) Deep-TMO (Single-Scale)

Fig. 1: Comparison between CNN (encoder-decoder) with $\mathcal{L}_1$-loss and DeepTMO (single-scale). Inlets in row 2 show that DeepTMO yields sharp and high resolution output, whereas the CNN results in blurred outputs.

c) Motivation for GANs: To achieve the desired TMO, one solution is to use a simple $\mathcal{L}_1$ or perceptual ($\mathcal{L}_{prp}$) loss function [43] with an encoder-decoder architecture as utilized in the past by various inverse-TMOs for generating HDR scenes from single-exposure [33] or multi-exposure [37] LDR images. However, such naive loss functions suffer from either overall spatial blurring (evident in $\mathcal{L}_1$ loss in Fig. 1) or over-compression of contrast (evident in $\mathcal{L}_{prp}$ loss in Fig. 2). This is mainly because a CNN architecture learns a mapping from all possible dynamic range values available in the wide

(a) $\mathcal{L}_{prp}$-loss (b) Deep-TMO (Single-Scale)

Fig. 2: Comparison between CNN (encoder-decoder) with $\mathcal{L}_{prp}$-loss (perceptual) and DeepTMO (single-scale).

variability of training-set scenes to a range [0,255]. Thus, the trained model effectively predicts a fairly mean luminance value for most of the pixels in output images to minimize the overall loss function. Another simple idea could be to use TMQI directly as loss function. However, due to the mathematical design of TMQI's naturalness component and characteristic discontinuity, TMQI cannot be directly used a loss function for back-propagation in a DL framework. In fact, the alternate methodology proposed by authors in [2], which optimizes a given TMO using TMQI, is also impossible to be imbibed into an end-to-end DL pipeline, as it treats both SSIM and naturalness separately using two different optimization strategies.

Given the goal of our TMO, designing an effective cost function manually for catering to wide variability of tone-mapping characteristics under different scenic-content is quite a complex task. An alternate solution could be to *learn* such a loss function. The use of GAN is an apt choice here, as it learns an adversarial loss function by itself (the loss being the discriminator network), that encapsulates all the desired features for an ideal TMO encoded in the underlying training data, thereby eradicating the need of manually designing a loss function. An added advantage of GAN is that it facilitate to obtain perceptually superior tone-mapped solutions residing in the subspace of natural images as compared to reproducing closer to mean valued or blurred outputs in case of ordinary $\mathcal{L}_1$ / $\mathcal{L}_{prp}$ loss functions.

Aiming an artifact-free high-resolution tone-mapped output, we begin investigating the choice of architecture from single-scale to its multi-scale variant for both generator and discriminator in the following sections.

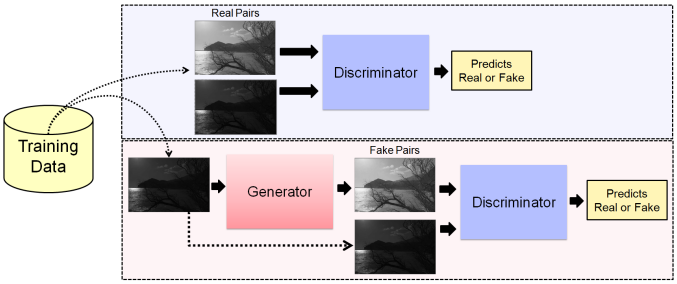


Fig. 3: DeepTMO Training Pipeline.

B. DeepTMO (Single-Scale)

Fig. 3 depicts an overview of our training algorithm. For our DeepTMO model, we basically employ a cGAN frame-

work [12] which implicitly learns a mapping from an observed HDR image x to a tone mapped LDR image y, given as: $G : x \longrightarrow y$. The architecture is composed of two fundamental building blocks namely a discriminator (D) and a generator (G).

The input to G consists of an $H \times W \times C$ size HDR image normalized between $[0, 1]$. We consider $C = 1$ *i.e.*, only the luminance channel is given as an input. Its output is a tone-mapped image (top row of fake pair in Fig. 3) of same size as the input. D on the other hand, takes luminance channels of HDR and tone mapped LDR images as input pairs, and predicts whether they are real tone-mapped images or fake. It is trained in a supervised fashion, by employing a training dataset of input HDR and their corresponding *target* tone-mapped images (real-pair in Fig. 3). We detail the complete methodology to build our target dataset in Section IV. An additional advantage of conditioning on an HDR input is that it empowers D to have some pre-information to make better reasoning for distinguishing between a real or fake tone mapped images, thus accelerating its training.

Next, we discuss the architectures for single-scale generator (Single-G) and single-scale discriminator (Single-D) which are our adaptations from past studies [43], [47] which show impressive results for style transfer and super-resolution tasks on LDR images. Further on, in the subsequent sections, we will reason as to why opting for their multi-scale versions aids in further refining the results.

a) Generator Architecture (Single-G): The Single-G architecture is an encoder-decoder architecture as shown in Fig. 4a. Overall, it consists of a sequence of 3 components: the convolution front end $G^{(Front)}$, a set of residual blocks $G^{(Res)}$ and the deconvolution back end $G^{(Back)}$. $G^{(Front)}$ consists of 4 different convolution layers which perform a subsequent down-sampling operation on their respective inputs. $G^{(Res)}$ is composed of 9 different residual blocks each having 2 convolution layers, while $G^{(Back)}$ consists of 4 convolution layers each of which up-samples its input by a factor of 2. During the down-sampling, $G^{(Front)}$ compresses the input HDR, thus keeping the most relevant information. $G^{(Res)}$ than applies multiple residual corrections to convert the compressed representation of input HDR to one that of its target LDR counterpart.

Finally, $G^{(Back)}$ yields a full size LDR output from this compressed representation through the up-sampling operation.

b) Discriminator Architecture (Single-D): The Single-D architecture resembles a 70×70 PatchGAN [13], [17], [18] model, which aims to predict whether each 70×70 overlapping image patch is real or fake, as shown in Fig. 4b. The main motivation of choosing a PatchGAN discriminator over a full-image size discriminator is that it contains much less parameters allowing it to be easily used for any-size images in a fully convolutional manner. This is pertinent for our problem setting where we involve very high resolution images. An added advantage of a PatchGAN discriminator is that while working on patches, it also models the high-frequency information by simply restricting its focus upon the structure in local image regions. The Single-D is run across the entire image, and all the responses over various patches are averaged out to yield the final prediction for the image. Note that the input to D is a concatenation of the HDR and its corresponding LDR image.

Although the Single-G and Single-D architecture yields high-quality reconstructions at a global level, yet it results in noisy artifacts over some specific areas such as bright light sources as shown in Fig. 5a. In a way, it necessitates modifying both single-scale versions of G and D to cater not only to coarser information, but at the same time, paying attention to finer level details, thus resulting in a much more refined tone-mapped output.

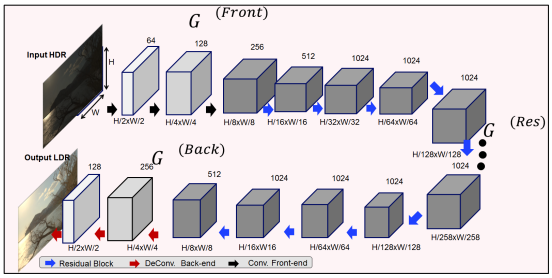
C. DeepTMO (Multi-Scale)

While generating high resolution tone-mapped images, it is quite evident now that we need to pay attention towards low-level minute details as well as high-level semantic information. To this end, motivated from [51], we alter the existing DeepTMO (single-scale) model, gradually incorporating step-by-step a multi-scale discriminator (Multi-D) and a multi-scale generator (Multi-G) in the algorithmic pipeline. Different from [51], our adaptation (a) utilizes a 2-scaled discriminator, (b) incorporates a different normalization layer in the beginning given by $\frac{(x-x_{min})}{(x_{max}-x_{min})}$, scaling pixels between [0,1] with a high 32-bit floating point precision, (c) inputs specifically a single luminance channel input with 32-bit pixel-depth linear HDR values.

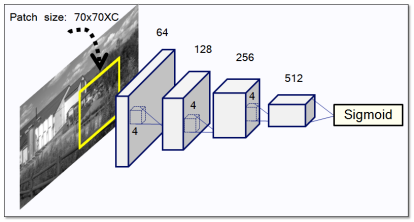
In the following, we detail the multi-scale versions of G and D. We showcase the impact through step-wise substitution of the Single-D with its Multi-D variant, and then the Single-G as well with its Multi-G counterpart.

a) Multi-D: Correctly classifying a high-resolution tone-mapped output as real or fake is quite challenging for Single-D. Even though an additional loss term effectively removes noisy artifacts at a global scale in the image (illustrated later in Section III-D), we still witness repetitive patterns in specific localized regions while using Single-D (for *e.g.*, seen around high illumination sources like inside/outside the ring of table lamp in Fig. 5a and on the ring of the lamp in Fig 5c). One easy way to tackle this problem is by focusing the discriminator's attention to a larger receptive field which is possible either through a deeper network or larger convolution kernels. However, it would in-turn demand a higher memory-bandwidth, which is already a constraint for training high-resolution HDR images. Thus, we basically retain the same network architecture for the discriminator as used previously, but rather apply it on two different scales of input *i.e.*, the original and the $2\times$ down-sampled version, calling the two discriminators D_o and D_d respectively.

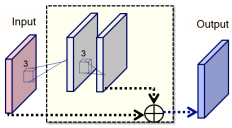
Both D_o and D_d are trained together to discriminate between real and synthetically generated images. D_d, by working on a coarser scale, focuses on a larger area of interest in patches throughout the image. This feature subsequently aids G to generate more globally consistent patch-level details in the image. D_o on the other hand, operating at a much finer scale than D_d, aids in highlighting more precise finer nuances in patches, thus enforcing G to pay attention towards very minute details too at the time of generation. Thus, by



(a) Generator (Single Scale).

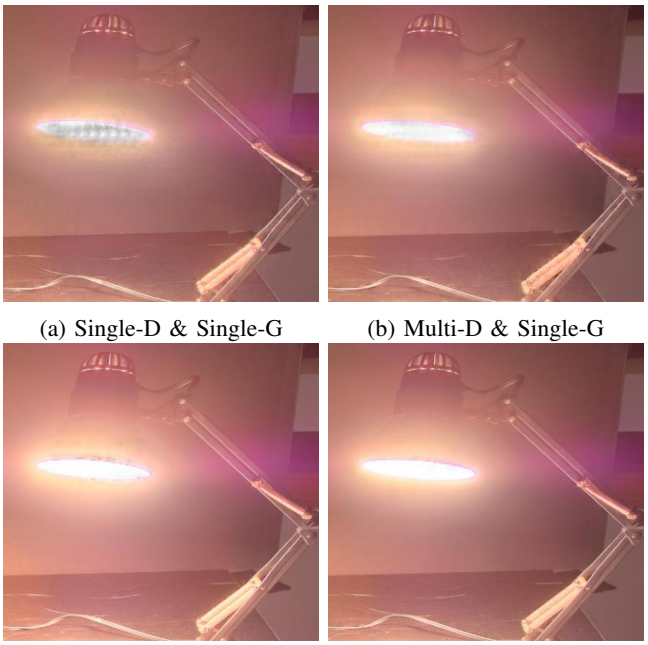


(b) Discriminator (Single Scale).



(c) Residual Blocks

Fig. 4: DeepTMO (single-scale) generator and discriminator architecture. The generator in (a) is an encoder-decoder architecture. Residual blocks in (c) consist of two sequential convolution layers applied to the input, producing a residual correction. Discriminator in (b) consists of a patchGAN [13], [17], [18] architecture which is applied patch wise on the concatenated the input HDR and tone mapped LDR pairs. More details in Supplementary.



(a) Single-D & Single-G (b) Multi-D & Single-G

(c) Single-D & Multi-G (d) Multi-D & Multi-G

Fig. 5: Impact of Multi-scale Discriminator and Generator.

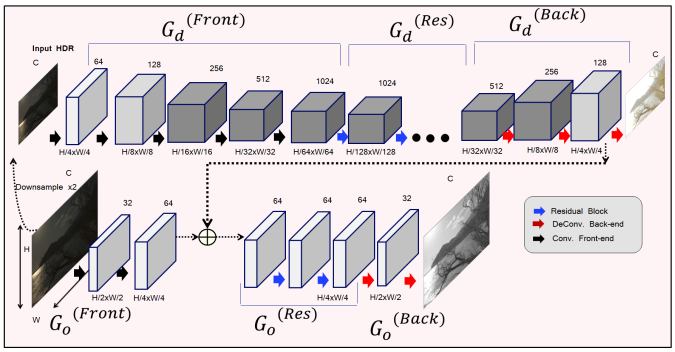


Fig. 6: DeepTMO multi-scale generator architecture. While the finer generator G_o has the original image as its input, the input to G_d is a $2\times$ down-sampled version.

introducing a Multi-D, the noisy patterns observed in Single-D, are suppressed to a large extent (for *e.g.*, as seen in Fig. 5a and Fig. 5b). However, we still witness minor traces of these artifacts due to Single-G's very own limitations, thus compelling us to switch to Multi-G. Contrary to Single-G, Multi-G reproduces outputs taking notice of both coarser and finer scales. Thus, the resultant output, having information over both scales, yields a more globally consistent and locally refined artifact-free image (for *e.g.*, as seen in Fig. 5b and Fig. 5d.

b) Multi-G: Fig. 6 illustrates the design of Multi-G. It mainly comprises of two sub-architectures, a global down-sampled network G_d and a global original network G_o. The architecture for G_d is similar to Single-G with the components, convolutional front-end, set of residual blocks and convolutional back-end being represented as: $G_d^{(Front)}, G_d^{(Res)}, G_d^{(Back)}$, respectively. G_o is also similarly composed of three components given by: $G_o^{(Front)}$, $G_o^{(Res)}$ and $G_o^{(Back)}$.

As illustrated in Fig. 6, at the time of inference, while the input to G_o is a high resolution HDR image (2048×1024), G_d receives a $2\times$ down sampled version of the same input. G_o effectively makes tone-mapped predictions, paying attention to local fine-grained details (due to its limited receptive field on a high resolution HDR input). At the same time, it also inputs from G_d, a coarser prediction (as its receptive field has a much broader view). Thus, the final generated output from $G_o^{(Back)}$ encompasses local low-level information and global structured details together in the same tone-mapped output. Hence, what we finally obtain is a much more structurally preserved and minutely refined output which is free from local noisy-artifacts, as seen in Fig. 5d.

To summarize, we showcase 4 different cGAN designs where the:

1) Single-D & Single-G architecture encounters noisy patterns due to not paying attention to finer-level details.
2) Multi-D & Single-G architecture is able to suppress patterns to some extent as observed in the previous case. This is mainly due to limited generalization capabilities of Single-G.
3) Single-D & Multi-G architecture removes patterns throughout the image, however some very localized regions still face artifacts due to the limited capacity

of Single-D.

4) Multi-D & Multi-G architecture finally yields superior quality artifact-free images.

D. Tone Mapping Objective Function

The ultimate goal of G is to convert high resolution HDR inputs to tone mapped LDR images, while D aims to distinguish real tone-mapped images from the ones synthesized by G. We train both the G and D architectures in a fully supervised setting. For training, we give a set of pairs of corresponding images $\{(x_i, y_i)\}$, where x_i is the luminance channel of the HDR input image while y_i is the luminance channel output of the corresponding tone-mapped LDR image. Next, we elaborate upon the objective function to train our DeepTMO (both single-scale and multi-scale).

The basic principle behind cGAN [12] is to model the conditional distribution of real tone-mapped images given an input HDR via the following objective:

$$\mathcal{L}_{cGAN}(G, D) = E_{x,y}[\log D(x, y)] + E_x[\log(1 - D(x, G(x)))], \tag{1}$$

where G and D compete with each other; G trying to minimize this objective against its adversary D, which tries to maximize it, i.e. $G^* = \arg\min_G \max_D \mathcal{L}_{cGAN}(G, D)$.

Since the Multi-D architecture consists of D_o and D_d, our objective for the same is:

$$G^* = \arg\min_G \max_{D_o, D_d} \sum_{s=o,d} \mathcal{L}_{cGAN}(G, D_s) \tag{2}$$

We append to the existing cGAN loss, an additional regularization term in the form of a feature matching (FM) loss $\mathcal{L}_{FM}(G, D_s)$ (similar to perceptual loss [53], [54]), given by:

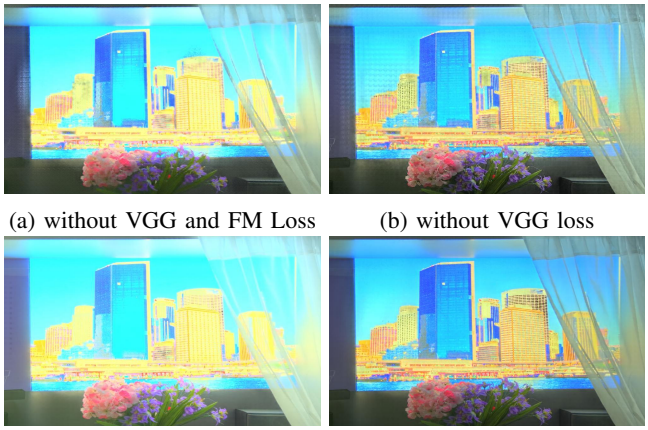
$$\mathcal{L}_{FM}(G, D_s) = E_{(x,y)} \sum_{i=1}^{M} \frac{1}{U_i} [||D_s^{(i)}(x, y) - D_s^i(x, G(x))||_1], \tag{3}$$

where D_s^i is the i^{th} layer feature extractor of D_s (from input to the i^{th} layer of D_s), M is the total number of layers and U_i denotes the number of elements in each layer. In short, we extract features from each individual D layer and match these intermediate representations over real and generated images. Additionally, we append a perceptual loss L_{prp} as used in [43], which constitutes of features computed from each individual layer of a pre-trained 19-layer VGG network [55] given by: $\mathcal{L}_{L_{prp}}(G) = \sum_{i=1}^{N} \frac{1}{V_i} [||F^{(i)}(y) - F^{(i)}(G(x))||_1]$

where $F^{(i)}$ denotes the i^{th} layer with V_i elements of the VGG network. The VGG network had been pre-trained for large scale image classification task over the Imagenet dataset [56]. Henceforth, our final objective function for a DeepTMO can be written as:

$$\begin{aligned} G^* = \arg\min_G \max_{D_o, Dd} \sum_{s=o,d} \mathcal{L}_{cGAN}(G, D_s) + \\ \beta \sum_{s=o,d} \mathcal{L}_{FM}(G, D_s) + \gamma \mathcal{L}_{L_{prp}}(G) \end{aligned} \tag{4}$$

β and γ controls the importance of $\mathcal{L}_{FM}$ and $\mathcal{L}_{L_{prp}}$ with respect to $\mathcal{L}_{cGAN}$ and both are set to 10. We illustrate the impact of both these terms in the following paragraph.



(a) without VGG and FM Loss (b) without VGG loss

(c) without FM Loss (d) with VGG and FM loss

Fig. 7: DeepTMO (single-scale) with/without FM and VGG Loss.

a) Impact of Feature Matching and Perceptual Loss term: Both $\mathcal{L}_{FM}$ and $\mathcal{L}_{L_{prp}}$ loss terms act as guidance to the adversarial loss function preserving overall natural image statistics and training without both these terms results in inferior quality throughout the image. The VGG-term primarily checks for global noisy repetitive patterns in the image and helps in suppressing them. While being applied on the full generated image, the VGG network captures both low-level image characteristics (*e.g.*, fine edges, blobs, colors etc,) and the high level semantic information through its beginning-level and later-stage network layers, respectively. Based upon these features, VGG effectively detects the corresponding artifacts as a shortcoming in the overall perceptual quality of the generated scene and hence guides to rectify them; thereby yielding a more natural image. For *e.g.*, the removal of noisy can be visualized by looking simultaneously at Fig. 7b and 7d. The FM loss term on the other hand, caters to more localized quality details like keeping a watch on illumination conditions in each sub-region. For *e.g.*, it effectively tones-down over-exposed regions of windows in the building as can be seen in Fig. 7c and 7d. This is ideally done by utilizing various feature layers of D, which are trained by focusing upon 70×70 localized image patches. Together both (VGG and FM) loss terms help in yielding a high quality overall contrast and local finer-details preserved output image (as seen in Fig. 7d).

E. Network Insight

Every component in the network plays an indispensable role in the overall tone-mapping. Starting from the convolutional front ends $G_d^{(Front)}$ and $G_o^{(Front)}$, both of which are applied directly on the linear HDR input, compress its tone and transform it to an encoded representation in an HDR space. While the convolutional layers play a critical role in down-sampling the spatial resolution by deriving meaningful feature layers using its learnt filters, the Instance Norm and activation functions (following each conv-layer) help in compressing the dynamic range of each pixel intensity. Next, the residual layers $G_d^{(Res)}$ and $G_o^{(Res)}$ can be understood as functions that map the current encoded information in HDR space to one in the LDR

space. This is essentially accomplished by adding a residual information to the current compressed form of HDR input. Finally, the $G_d^{(Back)}$ and $G_o^{(Back)}$ are applied to this encoded representation in the LDR space in order to transform it into a rich and visually pleasing LDR output. While the transposed convolution pay special attention to spatial upsampling, the activation functions maintains a compressed tone which is perceptually 'the most' appealing for a given scene.

TABLE I: Abbreviations

Symbols	Meaning
G_o, G_d	Generator Original, Downsampled scale
D_o, D_d	Discriminator Original, Downsampled scale
$\mathcal{L}_{cGAN}$	Adversarial Loss
$\mathcal{L}_{FM}$	Feature Matching Loss
$\mathcal{L}_{prp}$	Perceptual Loss
$\mathcal{L}_1$	L_1 Absolute Loss
H, W, C	Height, Width and Channel
β, γ	control parameters

IV. Building the HDR Dataset

In order to design a deep CNN based TMO, it is essential to obtain a large-scale dataset with a wide diversity of real-world scenes that are captured using a variety of cameras. To this end, we gather the available HDR datasets. For training the network, a total of 698 images are collected from various different sources, listed in the supplementary material. From the HDR video dataset sources, we select the frames manually so that no two chosen HDR images are similar. All these HDR images have been captured from diverse sources which is beneficial for our objective *i.e.*, learning a TMO catering a wide variety of real-world scenes.

To further strengthen the training, we applied several data augmentation techniques such as random cropping and flipping, which are discussed briefly in section V-2. We considered 105 images from the [57] for testing purposes.

A. Target Tone Mapped Images

Selecting a 'target' tone mapped image for a given HDR scene is a crucial step in training the DeepTMO. Although several subjective studies [19] built on different hypotheses have attempted to answer this question, yet they have been conducted only for very small databases of sizes upto 15-20 scenes. Such subjectively evaluated databases are limited in number and cannot be effectively used as training dataset for our DeepTMO model. Additionally, these databases have been evaluated under varying evaluation setups *i.e.*, by using different sets of TMOs and reference or no-reference settings. Hence, similar to [52], we resorted to a widely used objective-metric known as TMQI [31] to ensure a fixed target selection criterion for our problem.

As discussed in Section 2, literature of TMOs is quite extensive and practically difficult to span. Therefore, to find the target tone mapped image for each training HDR scene, we selected 13 classical TMOs: [20], [24], [58], [26], [23], [21], [22], [59], [28], [25], [27] and gamma and log mappings [19]. The selection of these tone mappings is inspired from the subjective evaluation studies [8], [31], [2], [30] which highlight the distinctive characteristics of mapping functions, which we aim to inculcate into the learning of our DeepTMO model.

For each HDR scene, we initially rank the obtained tone-mapped outputs from all the 13 TMOs using the TMQI metric. Then, the best scoring tone mapped output is selected as the 'target' for the corresponding HDR scene. Since tuning the parameters of 13 considered TMOs is a daunting task for a large set of training images, we used their default parameter settings throughout this paper. Though we acknowledge that fine-tuning TMO parameters can further boost overall performance, the process however, is almost impractical considering the large amount of training images and the vast parameter-space of the TMOs.

V. Training and Implementation Details

DeepTMO training paradigm is inspired by the conventional GANs approach, where alternate stochastic gradient descent (SGD) steps are taken for D followed by the G. We specifically utilize Least Square GANs (LSGANs), which have proven to yield [60] a much more stable learning process compared to regular GANs. For the multi-scale architecture, we first train G_d separately, and then fine tune both G_d and G_o (after freezing the weights of G_d for the first 20 epochs). For both D and G, all the weights corresponding to convolution layers are initialized using zero mean Gaussian noise with a standard deviation of 0.02, while the biases are set to 0.

(a) With Instance Norm (b) With Batch Norm

Fig. 8: Batch Normalization vs. Instance Normalization.

1) Instance Vs. Batch Norm: We use instance normalization [61], which is equivalent to applying batch normalization [62] using a batch size equal to 1.

The efficacy of the instance-norm is showcased in Fig. 8, where applying the plain batch-norm yields non-uniformity in luminance compression. While the instance normalization is trained to learn mean and standard deviation over a single-scene for the purpose of normalizing, the batch-norm learns over a full batch of input images. Thus, its mean and standard deviation is computed spatially for each pixel from a much wider range of high dynamic luminance values over the entire batch leading to uneven normalization.

Absence of batch-norm/instance-norm prevents the G/D to train properly and results in poor generation quality, thus necessitating the need for a normalization layer. All the instance normalization layers are initialized using Gaussian noise with mean 1 and 0.02 standard deviation.

2) Implementation: All training experiments are performed using the Pytorch [63] deep learning library with mini-batch SGD, where the batch size is set to 4. For multi-scale, we use batch-size 1 due to limited GPU memory. We utilize an ADAM solver [64] with initial learning rate fixed at 2×10^{-4} for the

first 100 epochs and then, allowed to decay to 0.0 linearly, until the final epoch. Momentum term β_1 is fixed at 0.5 for all the epochs. Hyper-parameters have been set to their default values and arent manipulated much due to GANs training complexity. We also employ random jitters by first resizing the original image to 700×1100, and then randomly cropping to size 512×512. For multi-scale, we resize to 1400×2200 and crop to size 1024×1024. All our networks are trained from scratch.

For all the other handcrafted TMOs, we used the MATLAB-based HDR Toolbox [19] and Luminance HDR software [1]. For each TMO, we enabled the default parametric setting as suggested by the respective authors. Training is done using a 12 Gb NVIDIA Titan-X GPU on a Intel Xeon e7 core i7 machine for 1000 epochs and takes a week.

VI. Results and Evaluation

In this section, we present the potential of our DeepTMO on a wide range of HDR scenes, containing both indoor and outdoor, human and structures, as well as day and night views. We compare our results with the best subjective outputs obtained from wide range of tone mapping methods [21], [27], [26], [22], [25], [24], [58], [59] on 105 images of test dataset [57], both qualitatively and quantitatively. In addition, we briefly discuss the specific characteristics of the proposed model, including their adaptation to content or sharpness in displaying high-resolution tone mapped outputs. Finally, we present a subjective evaluation study to access the perceived quality of the output. The size for each input image is kept fixed at 1024×2048.

Note that test scenes are different from the training set and are not seen by our model while training. Full size images and some additional results can be found in the supplementary material for better visual quality.

A. Comparison with the Best Quality Tone-Mapped Images

We begin the comparison of our DeepTMO model against the best quality tone mapped test images to assess the overall capability to reproduce high-quality images over a wide variety of scenes. To obtain the target test image, we follow a similar paradigm as provided in Section IV-A.

In Fig. 9, we demonstrate qualitative comparisons of our model with the two top scoring TMOs obtained using TMQI ranking, which includes methods like Mantiuk [27], Reinhard [59], Fattal [28], Durand [26], Drago [21], Pattnaik [24] TMO, over 7 explempary real-world scenes representing indoor/outdoor, with humans and structures, in day/night conditions. These sample scenes depict the exemplary mapping of linear HDR content using DeepTMO, where it successfully caters a wide variety of scenes as well as competes with the respective best quality outputs in terms of overall contrast preservation and visual appeal. In scene-1, a scene with human in indoor condition, we observe that our DeepTMO competes closely to the target output while preserving details in under/over exposed areas such as human face, areas under the table or outside the window. Another indoor scene-2, having shiny surfaces (indoor) and saturated outside regions (windows) demonstrate the effectiveness of our model by preserving details in these regions, yielding a high-quality output. Similar observations can be made in outdoor scenes with structures *i.e.*, in scene-3 and 4, where we notice that our DeepTMO model effectively tone-maps sharp frequency regions in overly exposed areas such as the dome of the building, the clouds in the sky or the cow's body. Landscape scene-5 has similar observations in the rising sun and dark forest regions. Although multi-scale DeepTMO design pays attention to the global and minute sub-regional information, the preservation of illumination and details in dull and overly bright regions is also due to the presence of the FM-loss term, which in turn utilizes features from different D layers. Since D is focused on localized image-patches, the FM-term implicitly understands how to compress or enhance luminance in specific regions.

More interestingly, we observe that DeepTMO suppresses noisy disturbances (*i.e.*, above the Waffle House store) in dark scene-6, which appears more pronounced in the two best performing tone-mapped images. This can be reasoned owing to the addition of VGG and FM-loss terms which guides the network to handle the noisy repetitive patterns and dark sensor-noise while preserving the natural scene statistics. Furthermore, we showcase a night time high-contrast scene-7, where our DeepTMO competes closely with the two best quality outputs while preserving the overall contrast ratio. However, we do observe the images obtained with our method have more saturated colors which we discuss later in Section VII-A.

Though in most cases our DeepTMO competes well with target images, in some cases we observe that it even outperforms them with respect to TMQI scores. Fig. 10 compares two exemplary HDR scenes from the test dataset that are mapped using the DeepTMO and their corresponding target TMOs in day and evening time-settings. In the first row, DeepTMO successfully preserves the fine details in the sky along with the waterfall and the mountains in the background. For a darker evening scene in second row, DeepTMO compensates the lighting and preserves the overall contrast of the generated scene. Even though we observe a halo ring around sun using our method (which we analyze later in Section VII-A), our TMQI score is considerably higher mainly because the TMQI metric is color-blind.

One possible explanation of such outcomes is the ability of the generator to learn the manifold of all available best tone mapping operators and subsequently developing a superior tone mapping functionality (from this manifold), which yields optimal output depending upon the scene. In other words, this manifold learning can be observed as a loose formulation built over the ideal characteristics (both global and local) desired for tone-mapping of different scene-types present in the training-set. In fact, learning such a complex mapping functionality is non-trivial by using a global TMQI metric score alone. This further confirms the goal of our training strategy.

a) Quantitative Analysis: To further demonstrate the high-quality mapping capability of DeepTMO models on all the 105 real world scenes, in Fig. 11, we show a distribution plot of the number of scenes against the TMQI Scores. For

[1] http://qtpfsgui.sourceforge.net/

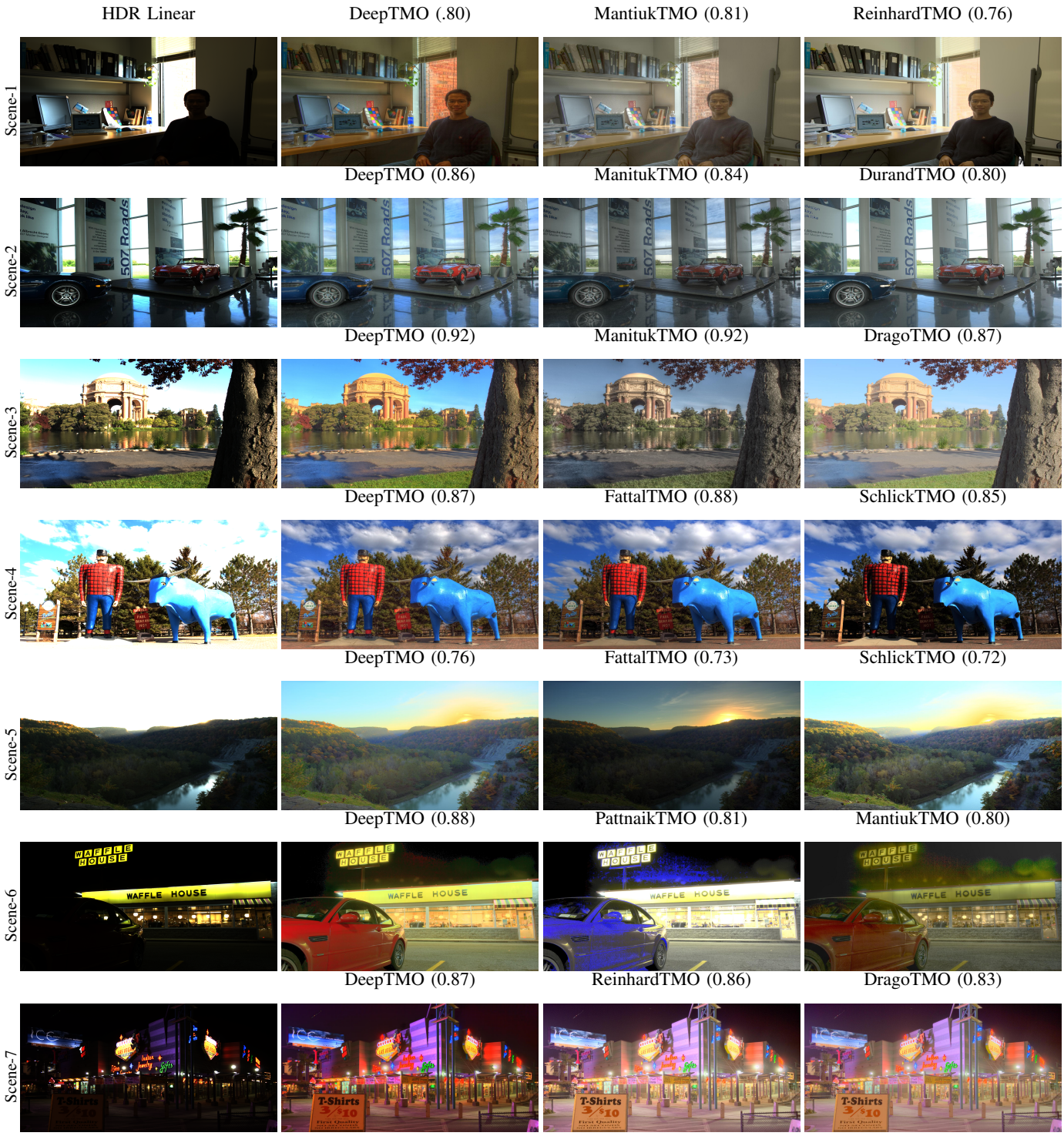


Fig. 9: Comparison between our DeepTMO outputs and outputs from top-2 ranked tone-mapped scenes on TMQI metrics for a variety of real-world scenes including indoor, scenes with structures, landscape, dark/noisy scenes. In brackets we show corresponding TMQI scores.

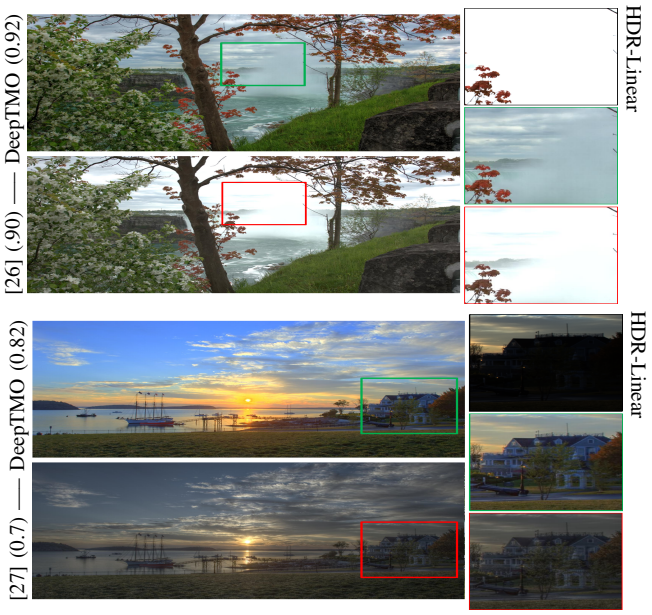


Fig. 10: Comparison between DeepTMO and targets, highlighting the zoom-ins with the corresponding HDR-linear input.

completeness, we also provide scores achieved by target tone-mapped outputs. The curves clearly show that the generated tone mapped images for DeepTMO compete closely with the best available tone mapped images on the objective metrics with DeepTMO fairing the best amongst all.

We provide quantitative analysis in Table II, to showcase the performance of our proposed model with the existing approaches. For each method, the TMQI scores are averaged over 105 scenes of the test dataset. The final results show that our proposed tone mapping model adapts for the variety of scenes and hence, achieves highest score. Please note that standard TMOs were applied with default parameter settings and hence results may improve for them by parameter optimization. Still performance of our fully automatic approach is highly competitive.

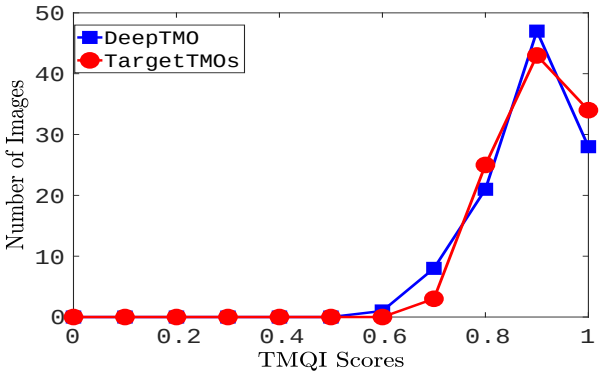


Fig. 11: Quantitative performance comparison of best performing DeepTMO with the target TMOs.

b) Computation Time: Inference is performed on test-images of size 1024×2048 and takes on an average 0.0187 sec. for single-scale and 0.0209 sec. for multi-scale designs, as shown in Figure 13.

TABLE II: *Quantitative Results*. Mean TMQI scores on the test-set of 105 images.

TMOs	TMQI
Tumblin [23] TMO	0.69 ±0.06
Chiu [25] TMO	0.70 ±0.05
Ashikh [58] TMO	0.70 ±0.06
Ward [20] TMO	0.71 ±0.07
Log [19] TMO	0.72 ±0.09
Gamma [19] TMO	0.76 ±0.07
Pattnaik [24] TMO	0.78 ±0.04
Schlick [22] TMO	0.79 ±0.09
Durand [26] TMO	0.81 ±0.10
Fattal [28] TMO	0.81 ±0.07
Drago [21] TMO	0.81 ±0.06
Reinhard [59] TMO	0.84 ±0.07
Mantiuk [27] TMO	0.84 ±0.06
DeepTMO (Single G - Single G)	0.79 ±0.06
DeepTMO (Single G - Multi D)	0.81 ±0.05
DeepTMO (Multi G - Single D)	0.80 ±0.07
DeepTMO (Multi G - Multi D)	**0.88 ±0.06**

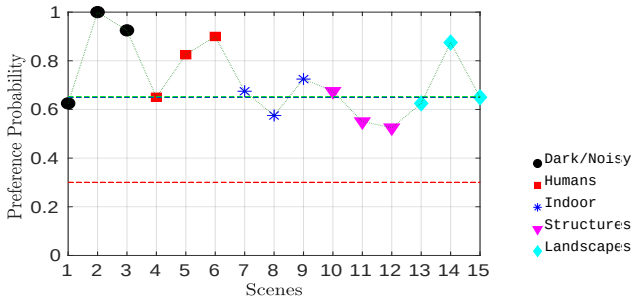


Fig. 12: Subjective Test Results. Preference probability of our DeepTMO over best performing target TMOs for 15 scenes representing 5 different scene categories.

B. Quality Evaluation

We performed a subjective pairwise comparison to validate the perceived quality of our tone-mapped images. 20 people participated in this subjective study, with age range of 23-38 years, normal or corrected-to-normal vision.

1) Test Environment and Setup: The tests were carried out in a room reserved for professional subjective tests with ambient lighting conditions. A Dell UltraSharp 24 Monitor (DELL U2415) was used for displaying images with screen resolution 1920×1200 at 59 hz. The desktop background window was set at 128 gray value.

Each stimuli included a pair of tone mapped images for a given scene, where each pair always consisted of an image produced by DeepTMO and the other one obtained using the best-performing tone mapping functions based on the TMQI rankings. To cater a wide variety of content, we selected 15 scenes from 105 test-set images, representing 5 different categories (3 scenes per category) namely, i) Humans, ii) Dark/Noisy, iii) Indoor, iv) Structures, and v) Landscapes.

2) Procedures: We conducted a pair-wise subjective experiment where the observer was asked to choose an image by showing a pair of images side-by-side. The option same was not included to force users to choose one of the stimuli. Each participant was asked to select an image which is more

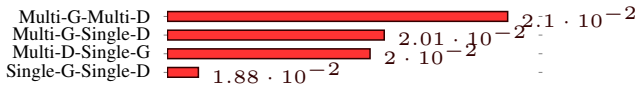


Fig. 13: *Computation time in seconds.*

realistic and appealing to him/her. Participants were provided with unlimited time to make their decision and record their choice. The experiment was divided into a training and test session, where training involved each participant being briefed to familiarize with the subjective quality evaluation task. Each observer compared a pair of image twice, having each tone-mapped image displayed on both sides (*e.g.*, DeepTMO vs. first-best tone mapped and first-best tone mapped vs. DeepTMO).

3) Results: In order to quantify the results of pairwise subjective tests, we scaled the winning frequencies of the model to the continuous quality-scores using the widely known Bradley-Terry (BT) model in [65]. The scaling is performed using the statistical analysis proposed in [66] to determine whether the perceived visual quality difference of the compared models is statistically significant. The preference probability for our method $Pref - Prob_{(DeepTMO)}$ is mathematically given as:

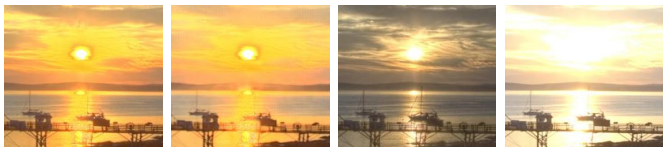
$$Pref - Prob_{(DeepTMO)} = \frac{w_{DeepTMO}}{N} + \frac{t}{2 \cdot N} \qquad (5)$$

where $w_{DeepTMO}$ is the winning frequency of our proposed model, t is the tie frequency and N is the total number of participants. The statistical model relies on the hypothesis that each compared TMO in the pairwise test shares equal probability of occurrence *i.e.*, 0.5 and hence, follows a Binomial distribution. Based on the initial hypothesis, a Binomial test was performed on the collected data and the critical thresholds were obtained by plotting the cumulative distribution function of the Binomial distribution. By setting 95% as the level of significance, if we receive 13 $(B(13, 20, 0.5) = 0.9423)$ or more votes for our proposed method, we consider our tone-mapped image to be significantly favored in terms of subjective quality. Similarly, by setting 5% as the significance level, if we receive 6 $(B(6, 20, 0.5) = 0.0577)$ or less votes for our proposed method, we consider our tone-mapped image to be least favored in terms of subjective quality.

The results of the pair-wise subjective quality experiment are shown in Fig. 12. The two lines (blue and red) mark probabilities of high $(13/20 = .65)$ and low $(6/20 = .30)$ favor-abilities respectively. Looking at the results, we observe that DeepTMO images have been significantly preferred over best TMQI rated tone mapped images for most of the scenes, for different possible categories. In general, we observed that subjects preferred our tone-mapped LDR scenes which preserve the contrast well. Based on some informal post-experiment interviews, we found that best TMQI rated target images, preserving fine details were least realistic and more like paintings to observers. A small set of images used in subjective tests is shown in Fig. 9.

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

Fig. 14: Top TMQI scoring TMOs showing not-so-visually desirable outputs. (a) DeepTMO output, (b), (c) and (d) are 3 top ranking TMO output.



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

Fig. 15: Halo effect. (a) DeepTMO output, (b) DeepTMO trained with log-scaled values, (c) and (d) 2 top ranking TMO outputs.

VII. Conclusion, Limitations and Future work

Designing a fast, automated tone-mapping operator that can reproduce best subjective quality outputs from a wide range of linear-valued HDR scenes is a daunting task. Existing TMOs address some specific characteristics, such as overall contrast ratio, local fine-details or perceptual brightness of the scene. However, the entire process of yielding high-quality tone-mapped output remains a time-consuming and expensive task, as it requires an extensive parameter tuning to produce a desirable output for a given scene.

To this end, we present an end-to-end parameter-free DeepTMO. Tailored in a cGAN framework, our model is trained to output realistically looking tone-mapped images, that duly encompass all the various distinctive properties of the available TMOs. We provide an extensive comparison among various architectural design choices, loss functions and normalization methods, thus highlighting the role that each component plays in the final reproduced outputs. Our DeepTMO successfully overcomes the frequently addressed blurry or tiling effects in recent HDR related works [9], [37], a problem of significant interest for several high-resolution learning-based graphical rendering applications as highlighted in [9]. By simply learning an HDR-to-LDR cost function under a multi-scale GANs framework, DeepTMO successfully preserves desired output characteristics such as underlying contrast, lighting and minute details present in the input HDR at the finest scale. Lastly, we validate the versatility of our methodology through detailed quantitative and qualitative comparisons with existing TMOs.

(a) DeepTMO (b) s=0.5 (c) s=0.3

Fig. 16: Color Correction. (a) DeepTMO, (b) and (c) are the color corrected DeepTMO controlled by parameter s from [67].

A. Limitations and Future Work

a) Target Selection: Though DeepTMO successfully demonstrates versatility in addressing wide variety of scenes, its expressive power is limited by the amount of available training data and quality of its corresponding 'target'. As noted in Section I, due to unavailability of subjectively annotated 'best tone mapped images' for HDR scenes, we resort to an objective TMQI metric to build the corresponding target LDR. However, the metric itself is not as perfect as the human visual system. We illustrate this point in Fig. 14. The images ranked lower by TMQI metric in column 3 and 4 are somehow more interesting than their best-ranked counterpart in column 2. Such samples can eventually restrict the generation power of our model.

Another specific case includes 'Halo' artifacts or rings around high illumination regions such as the sun shown in Fig. 15, where DeepTMO (column 1) is compared with the top TMQI scoring outputs in column 3 and column 4. This is mainly due to the inadequate amount of training data consisting of such samples, and the presence of their overly saturated 'target' counterparts. As a result, D has very little information about effectively tone-mapping such regions, and thus is unable to guide G to effectively eradicate such effects at generation time. To handle such artifacts, we additionally experimented using a log-scale input (column 2) where we observe that even log-scale values do no rectify such effects, thus necessitating the need of adequate training samples.

An alternative future work to address this problem, can be to weakly rely on these 'noisy' tone-mapped ground truths images by utilizing a weakly supervised learning paradigm [68]. We can also learn HDR-to-LDR mapping in a completely unsupervised fashion without giving any input-output pairs [47]. This would allow the network to decide by itself which is the best possible tone-mapped output simply by independently modeling the underlying distribution of input HDR and output tone mapped images.

b) Color Correction: Color is an important aspect while rendering high quality subjective tone-mapped outputs. Our proposed method has been trained for efficient luminance compression in HDR scenes and uses the classical color ratios to produce the resulting tone-mapped outputs. Although it provides best subjective quality outputs in most cases, it sometimes can result into overly saturated colors which might look unnatural and perceptually unpleasant. One simple solution could be to simply plug-in existing color correction methods [67] to obtain the desired output. An example is shown in Fig. 16, where color correction has been carried out using the method as proposed in [67], which is given by $C_{out} = ((\frac{C_{in}}{L_{in}} - 1) \cdot s + 1) \cdot L_{out}$, where s is the color saturation control. Alternately, another interesting solution could be to learn a model to directly map the content from HDR color-space to an LDR colored tone mapped output.

REFERENCES

[1] A. Pardo and G. Sapiro, "Visualization of high dynamic range images," *IEEE Transactions on Image Processing*, vol. 12, no. 6, pp. 639–647, June 2003.

[2] K. Ma, H. Yeganeh, K. Zeng, and Z. Wang, "High dynamic range image compression by optimizing tone mapped image quality index," *IEEE Transactions on Image Processing*, vol. 24, Oct 2015.

[3] D. Gommelet, A. Roumy, C. Guillemot, M. Ropert, and J. L. Tanou, "Gradient-based tone mapping for rate-distortion optimized backward-compatible high dynamic range compression," *IEEE Transactions on Image Processing*, pp. 5936–5949, Dec 2017.

[4] A. Rana, G. Valenzise, and F. Dufaux, "Learning-based Adaptive Tone Mapping for Keypoint Detection," in *IEEE International Conference on Multimedia & Expo (ICME'2017)*, Hong Kong, China, Jul. 2017.

[5] F. Dufaux, P. Le Callet, R. Mantiuk, and M. Mrak, *High Dynamic Range Video: From Acquisition, to Display and Applications*. Academic Press, 2016.

[6] J. McCann and A. Rizzi, *The art and science of HDR imaging*, 01 2012.

[7] A. Rana, G. Valenzise, and F. Dufaux, "Learning-Based Tone Mapping Operator for Image Matching," in *IEEE International Conference on Image Processing (ICIP'2017)*. Beijing, China: IEEE, 2017.

[8] P. Ledda, A. Chalmers, T. Troscianko, and H. Seetzen, "Evaluation of tone mapping operators using a high dynamic range display," *ACM Transactions on Graphics (TOG)*, pp. 640–648, 2005.

[9] G. Eilertsen, R. Wanat, R. K. Mantiuk, and J. Unger, "Evaluation of Tone Mapping Operators for HDR-Video," *Computer Graphics Forum*, 2013.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[11] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014.

[12] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[13] P. Isola, J. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," *CVPR*, 2017.

[14] A. Rana, G. Valenzise, and F. Dufaux, "Optimizing tone mapping operators for keypoint detection under illumination changes," in *2016 IEEE 18th International Workshop on Multimedia Signal Processing (MMSP)*, Sep. 2016, pp. 1–6.

[15] K. Debattista, "Applicationspecific tone mapping via genetic programming," *Computer Graphics Forum*, vol. 37, no. 1, pp. 439–450, 2017.

[16] A. Rana, J. Zepeda, and P. Perez, "Feature learning for the image retrieval task," in *Computer Vision - ACCV 2014 Workshops*. Springer International Publishing, 2015, pp. 152–165.

[17] C. Li and M. Wand, "Precomputed real-time texture synthesis with markovian generative adversarial networks," in *European Conference on Computer Vision*. Springer, 2016, pp. 702–716.

[18] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," *arXiv preprint arXiv:1609.04802*, 2016.

[19] F. Banterle, A. Artusi, K. Debattista, and A. Chalmers, *Advanced High Dynamic Range Imaging: Theory and Practice*, February 2011.

[20] G. W. Larson, H. Rushmeier, and C. Piatko, "A visibility matching tone reproduction operator for high dynamic range scenes," *IEEE Transactions on Visualization and Computer Graphics*, pp. 291–306, Oct. 1997.

[21] F. Drago, K. Myszkowski, T. Annen, and N. Chiba, "Adaptive logarithmic mapping for displaying high contrast scenes," *Computer Graphics Forum*, pp. 419–426, 2003.

[22] C. Schlick, "An adaptive sampling technique for multidimensional integration by ray-tracing," in *Photorealistic Rendering in Computer Graphics*. Springer, 1994.

[23] J. Tumblin, J. K. Hodgins, and B. K. Guenter, "Two methods for display of high contrast images," *ACM Trans. Graph.*, pp. 56–94, Jan. 1999.

[24] S. Pattanaik and H. Yee, "Adaptive gain control for high dynamic range image display," in *Proceedings of the 18th Spring Conference on Computer Graphics*, ser. SCCG '02. ACM, 2002, pp. 83–87.

[25] K. Chiu, M. Herf, P. Shirley, S. Swamy, C. Wang, and K. Zimmerman, "Spatially nonuniform scaling functions for high contrast images," in *Proceedings of Graphics Interface '93*, ser. GI '93, Toronto, Ontario, Canada, 1993, pp. 245–253.

[26] F. Durand and J. Dorsey, "Fast bilateral filtering for the display of high-dynamic-range images," in *29th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '02, 2002.

[27] R. Mantiuk, K. Myszkowski, and H. P. Seidel, "A perceptual framework for contrast processing of high dynamic range images," *ACM Trans. Appl. Percept.*, vol. 3, no. 3, Jul. 2006.

[28] R. Fattal, D. Lischinski, and M. Werman, "Gradient domain high dynamic range compression," *ACM Trans. Graph.*, vol. 21, no. 3, pp. 249–256, Jul. 2002.

[29] A. Rana, G. Valenzise, and F. Dufaux, "Learning-based tone mapping operator for efficient image matching," *IEEE Transactions on Multimedia*, pp. 1–1, 2018.

[30] M. Cadik, M. Wimmer, L. Neumann, and A. Artusi, "Evaluation of HDR tone mapping methods using essential perceptual attributes," *Computers and Graphics*, pp. 330 – 349, 2008.

[31] H. Yeganeh and Z. Wang, "Objective quality assessment of tone-mapped images," *IEEE Transactions on Image Processing*, vol. 22, no. 2, pp. 657–667, Feb 2013.

[32] H. Z. Nafchi, A. Shahkolaei, R. F. Moghaddam, and M. Cheriet, "Fsitm: A feature similarity index for tone-mapped images," *IEEE Signal Processing Letters*, vol. 22, no. 8, pp. 1026–1029, Aug 2015.

[33] G. Eilertsen, J. Kronander, G. Denes, R. K. Mantiuk, and J. Unger, "Hdr image reconstruction from a single exposure using deep cnns," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 6, p. 178, 2017.

[34] D. Marnerides, T. Bashford-Rogers, J. Hatchett, and K. Debattista, "Expandnet: A deep convolutional neural network for high dynamic range expansion from low dynamic range content." 2018.

[35] Y. Kinoshita and H. Kiya, "Deep inverse tone mapping using LDR based learning for estimating HDR images with absolute luminance," *CoRR*, vol. abs/1903.01277, 2019.

[36] K. Moriwaki, R. Yoshihashi, R. Kawakami, S. You, and T. Naemura, "Hybrid loss for learning single-image-based HDR reconstruction," *CoRR*, vol. abs/1812.07134, 2018.

[37] Y. Endo, Y. Kanamori, and J. Mitani, "Deep reverse tone mapping," *ACM Transactions on Graphics (Proc. of SIGGRAPH ASIA 2017)*, vol. 36, no. 6, nov 2017.

[38] N. K. Kalantari and R. Ramamoorthi, "Deep high dynamic range imaging of dynamic scenes," *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2017)*, vol. 36, no. 4, 2017.

[39] M. Gharbi, G. Chaurasia, S. Paris, and F. Durand, "Deep joint demosaicking and denoising," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 6, p. 191, 2016.

[40] J. Chen, A. Adams, N. Wadhwa, and S. W. Hasinoff, "Bilateral guided upsampling," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 6, p. 203, 2016.

[41] M. Gharbi, J. Chen, J. T. Barron, S. W. Hasinoff, and F. Durand, "Deep bilateral learning for real-time image enhancement," *ACM Trans. Graph.*, Jul. 2017.

[42] X. Hou, J. Duan, and G. Qiu, "Deep feature consistent deep image transformations: Downscaling, decolorization and HDR tone mapping," *CoRR*, 2017.

[43] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European Conference on Computer Vision*. Springer, 2016, pp. 694–711.

[44] R. Yeh, C. Chen, T. Y. Lim, M. Hasegawa-Johnson, and M. N. Do, "Semantic image inpainting with perceptual and contextual losses," *arXiv preprint arXiv:1607.07539*, 2016.

[45] S. Lee, G. Hwan An, and S.-J. Kang, "Deep recursive hdri: Inverse tone mapping using generative adversarial networks," in *The European Conference on Computer Vision (ECCV)*, September 2018.

[46] S. Lee, G. H. An, and S. Kang, "Deep chain hdri: Reconstructing a high dynamic range image from a single low dynamic range image," *IEEE Access*, pp. 49 913–49 924, 2018.

[47] J. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," *arXiv preprint arXiv:1703.10593*, 2017.

[48] M. Liu, T. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks," *arXiv preprint arXiv:1703.00848*, 2017.

[49] H. Tung, A. Harley, W. Seto, and K. Fragkiadaki, "Adversarial inversion: Inverse graphics with adversarial priors," *arXiv preprint arXiv:1705.11166*, 2017.

[50] Q. Chen and V. Koltun, "Photographic image synthesis with cascaded refinement networks," in *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, 2017.

[51] T. Wang, M. Liu, J. Zhu, A. Tao, J. Kautz, and B. Catanzaro, "High-resolution image synthesis and semantic manipulation with conditional gans," *arXiv preprint arXiv:1711.11585*, 2017.

[52] V. A. Patel, P. Shah, and S. Raman, "A generative adversarial network for tone mapping hdr images," in *Computer Vision, Pattern Recognition, Image Processing, and Graphics*, Singapore, 2018, pp. 220–231.

[53] A. Dosovitskiy and T. Brox, "Generating images with perceptual similarity metrics based on deep networks," in *Advances in Neural Information Processing Systems*, 2016, pp. 658–666.

[54] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2414–2423.

[55] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[56] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[57] M. Fairchild. (2007) The hdr photographic survey. [Online]. Available: http://www.rit-mcsl.org/fairchild/HDR.html

[58] M. Ashikhmin, "A tone mapping algorithm for high contrast images," pp. 145–156, 2002.

[59] E. Reinhard, M. Stark, P. Shirley, and J. Ferwerda, "Photographic tone reproduction for digital images," *ACM Trans. Graph.*, pp. 267–276, Jul. 2002.

[60] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley, "Least squares generative adversarial networks," in *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, 2017, pp. 2813–2821.

[61] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," *preprint arXiv:1607.08022*, 2016.

[62] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *CoRR*, vol. abs/1502.03167, 2015.

[63] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017.

[64] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[65] R. A. Bradley and M. E. Terry, "The rank analysis of incomplete block designs — I. The method of paired comparisons," *Biometrika*, vol. 39, pp. 324–345, 1952.

[66] P. Hanhart, M. Rerabek, and T. Ebrahimi, "Towards high dynamic range extensions of hevc: subjective evaluation of potential coding technologies," *Applications of Digital Image Processing XXXVIII*, p. 95990G, 2015.

[67] R. Mantiuk, R. Mantiuk, A. Tomaszewska, and W. Heidrich, "Color correction for tone mapping," *Computer Graphics Forum*, 2009.

[68] Z. Lu, Z. Fu, T. Xiang, P. Han, L. Wang, and X. Gao, "Learning from weak and noisy labels for semantic segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 3, pp. 486–500, March 2017.