Single Image HDR Reconstruction Using a CNN with Masked Features and Perceptual Loss

使用具有蒙版特征和感知损失的CNN进
行单图像HDR重建

MARCEL SANTANA SANTOS, Centro de Informática, Universidade Federal de Pernambuco
TSANG ING REN, Centro de Informática, Universidade Federal de Pernambuco
NIMA KHADEMI KALANTARI, Texas A&M University

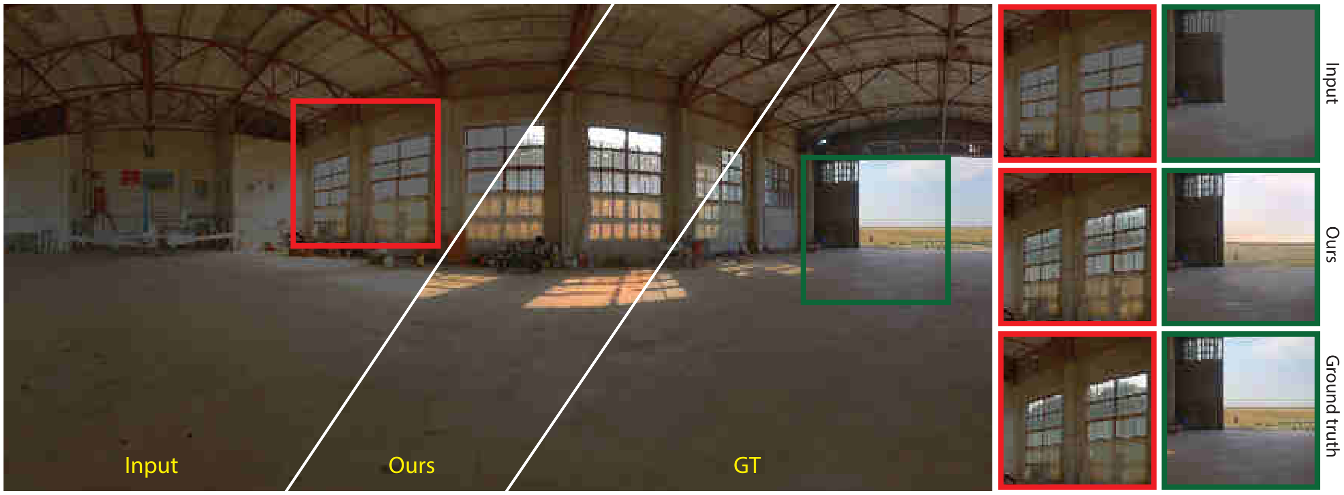

Fig. 1. We propose a novel deep learning system for single image HDR reconstruction by synthesizing visually pleasing details in the saturated areas. We introduce a new feature masking approach that reduces the contribution of the features computed on the saturated areas, to mitigate halo and checkerboard artifacts. To synthesize visually pleasing textures in the saturated regions, we adapt the VGG-based perceptual loss function to the HDR reconstruction application. Furthermore, to effectively train our network on limited HDR training data, we propose to pre-train the network on inpainting task. Our method can reconstruct regions with high luminance, such as the bright highlights of the windows (red inset), and generate visually pleasing textures (green insert). See Figure 7 for comparison against several other approaches. All images have been gamma corrected for display purposes.

Digital cameras can only capture a limited range of real-world scenes' luminance, producing images with saturated pixels. Existing single image high dynamic range (HDR) reconstruction methods attempt to expand the range of luminance, but are not able to hallucinate plausible textures, producing results with artifacts in the saturated areas. In this paper, we present a novel learning-based approach to reconstruct an HDR image by recovering the saturated pixels of an input LDR image in a visually pleasing way. Previous deep learning-based methods apply the same convolutional filters on well-exposed and saturated pixels, creating ambiguity during training and leading to checkerboard and halo artifacts. To overcome this problem, we propose a feature masking mechanism that reduces the contribution of the features from the saturated areas. Moreover, we adapt the VGG-based perceptual loss function to our application to be able to synthesize visually pleasing textures. Since the number of HDR images for training is limited, we propose to train our system in two stages. Specifically, we first train our system on a large number of images for image inpainting task and then fine-tune it on HDR reconstruction. Since most of the HDR examples contain smooth regions that are simple to reconstruct, we propose a sampling strategy to select challenging training patches during the HDR fine-tuning stage. We demonstrate through experimental results that our approach can reconstruct visually pleasing HDR results, better than the current state of the art on a wide range of scenes.

CCS Concepts: • **Computing methodologies → Computational photography**.

Additional Key Words and Phrases: high dynamic range imaging, convolutional neural network, feature masking, perceptual loss

ACM Reference Format:
Marcel Santana Santos, Tsang Ing Ren, and Nima Khademi Kalantari. 2020. Single Image HDR Reconstruction Using a CNN with Masked Features and Perceptual Loss. *ACM Trans. Graph.* 39, 4, Article 80 (July 2020), 10 pages. https://doi.org/10.1145/3386569.3392403

1 INTRODUCTION

The illumination of real-world scenes is high dynamic range, but standard digital cameras sensors can only capture a limited range of luminance. Therefore, these cameras typically produce images with

Authors' addresses: Marcel Santana Santos, Centro de Informática, Universidade Federal de Pernambuco, mss8@cin.ufpe.br; Tsang Ing Ren, Centro de Informática, Universidade Federal de Pernambuco, tir@cin.ufpe.br; Nima Khademi Kalantari, Texas A&M University, nimak@tamu.edu.

under/over-exposed areas. A large number of approaches propose to generate a high dynamic range (HDR) image by combining a set of low dynamic range images (LDR) of the scene at different exposures [Debevec and Malik 1997]. However, these methods either have to handle the scene motion [Hu et al. 2013; Kalantari and Ramamoorthi 2017; Kang et al. 2003; Oh et al. 2014; Sen et al. 2012; Wu et al. 2018] or require specialized bulky and expensive optical systems [McGuire et al. 2007; Tocci et al. 2011]. Single image dynamic range expansion approaches avoid these limitations by reconstructing an HDR image using one image. These approaches can work with images captured with any standard camera or even recover the full dynamic range of legacy LDR content. As a result, they have attracted considerable attention in recent years.

Several existing methods extrapolate the light intensity using heuristic rules [Banterle et al. 2006; Bist et al. 2017; Rempel et al. 2007], but are not able to properly recover the brightness of saturated areas as they do not utilize context. On the other hand, recent deep learning approaches [Eilertsen et al. 2017; Endo et al. 2017; Lee et al. 2018a] systematically utilize contextual information using convolutional neural networks (CNNs) with large receptive fields. However, these methods usually produce results with blurriness, checkerboard, and halo artifacts in saturated areas.

In this paper, we propose a novel learning-based technique to reconstruct an HDR image by recovering the missing information in the saturated areas of an LDR image. We design our approach based on two main observations. First, applying the same convolutional filters on well-exposed and saturated pixels, as done in previous approaches, results in ambiguity during training and leads to checkerboard and halo artifacts. Second, using simple pixel-wise loss functions, utilized by most existing approaches, the network is unable to hallucinate details in the saturated areas, producing blurry results. To address these limitations, we propose a feature masking mechanism that reduces the contribution of features generated from the saturated content by multiplying them to a soft mask. With this simple strategy, we are able to avoid checkerboard and halo artifacts as the network only relies on the valid information of the input image to produce the HDR image. Moreover, inspired by image inpainting approaches, we leverage the VGG-based perceptual loss function, introduced by Gatys et al. [2016], and adapt it to the HDR reconstruction task. By minimizing our proposed perceptual loss function during training, the network can synthesize visually realistic textures in the saturated areas.

Since a large number of HDR images, required for training a deep neural network, are currently not available, we perform the training in two stages. In the first stage, we train our system on a large set of images for the inpainting task. During this process, the network leverages a large number of training samples to learn an internal representation that is suitable for synthesizing visually realistic texture in the incomplete regions. In the next step, we fine-tune this network on the HDR reconstruction task using a set of simulated LDR and their corresponding ground truth HDR images. Since most of the HDR examples contain smooth regions that are simple to reconstruct, we propose a simple method to identify the textured patches and only use them for fine-tuning.

Our approach can reconstruct regions with high luminance and hallucinate textures in the saturated areas, as shown in Figure 1. We demonstrate that our approach can produce better results than the state-of-the-art methods both on simulated images (Figure 7) and on images taken with real-world cameras (Figure 9). In summary, we make the following contributions:

(1) We propose a feature masking mechanism to avoid relying on the invalid information in the saturated regions (Section 3.1). This masking approach significantly reduces the artifacts and improves the quality of the final results (Figure 10).
(2) We adapt the VGG-based perceptual loss function to the HDR reconstruction task (Section 3.2). Compared to pixel-wise loss functions, our loss can better reconstruct sharp textures in the saturated regions (Figure 12).
(3) We propose to pre-train the network on inpainting before fine-tuning it on HDR generation (Section 3.3). We demonstrate that the pre-training stage is essential for synthesizing visually pleasing textures in the saturated areas (Figure 11).
(4) We propose a simple strategy for identifying the textured HDR areas to improve the performance of training (Section 3.4). This strategy improves the network ability to reconstruct sharp details (Figure 11).

2 RELATED WORK

The problem of single image HDR reconstruction, also known as inverse tone-mapping [Banterle et al. 2006], has been extensively studied in the last couple of decades. However, this problem remains a major challenge as it requires recovering the details from regions with missing content. In this section, we discuss the existing techniques by classifying them into two categories of non-learning and learning methods.

2.1 Non-learning Methods

Several approaches propose to perform inverse tone-mapping using global operators. Landis [2002] applies a linear or exponential function to the pixels of the LDR image above a certain threshold. Bist et al. [2017] approximates tone expansion by a gamma function. They use the characteristics of the human visual system to design the gamma curve. Luzardo et al. [2018] improve the brightness of the result by utilizing an operator based on the mid-level mapping.

A number of techniques propose to handle this application through local heuristics. Banterle et al. [2006] use median-cut [Debevec 2005] to find areas with high luminance. They then generate an expand-map to extend the range of luminance in these areas, using an inverse operator. Rempel et al. [2007] also utilize an expand-map but use a Gaussian filter followed by an edge-stopping function to enhance the brightness of saturated areas. Kovaleski and Oliveira [2014] extend the approach by Rempel et al. [2007] using a cross bilateral filter. These approaches simply extrapolate the light intensity by using heuristics and, thus, often fail to recover saturated highlights, introducing unnatural artifacts.

A few approaches propose to handle this application by incorporating user interactions in their system. Didyk et al. [2008] enhance bright luminous objects in video sequences by using a semi-automatic classifier to classify saturated regions as lights, reflections, or diffuse surfaces. Wang et al. [2007] recover the textures in the saturated areas by transferring details from the user-selected regions. Their approach demands user interactions that take several minutes, even for an expert user. In contrast to these methods, we propose a learning-based approach to systematically reconstruct HDR images from a wide range of different scenes, instead of relying on heuristics strategies and user inputs.

2.2 Learning-based Methods

In recent years, several approaches have proposed to tackle this application using deep convolutional neural networks (CNN). Given a single input LDR image, Endo et al. [2017] use an auto-encoder [Hinton and Salakhutdinov 2006] to generate a set of LDR images with different exposures. These images are then combined to reconstruct the final HDR image. Lee et al. [2018a] chain a set of CNNs to sequentially generate the bracketed LDR images. Later, they propose [Lee et al. 2018b] to handle this application through a recursive conditional generative adversarial network (GAN) [Goodfellow et al. 2014] combined with a pixel-wise l_1 loss.

In contrast to these approaches, a few methods [Eilertsen et al. 2017; Marnerides et al. 2018; Yang et al. 2018] directly reconstruct the HDR image without generating bracketed images. Eilertsen et al. [2017] use a network with U-Net architecture to predict the values of the saturated areas, whereas linear non-saturated areas are obtained from the input. Marnerides et al. [2018] present a novel dedicated architecture for end-to-end image expansion. Yang et al. [2018] reconstruct HDR image for image correction application. They train a network for HDR reconstruction to recover the missing details from the input LDR image, and then a second network transfers these details back to the LDR domain.

While these approaches produce state-of-the-art results, their synthesized images often contains halo and checkerboard artifacts and lacks textures in the saturated areas. This is mainly because of using standard convolutional layers and pixel-wise loss functions. Note that, several recent methods [Kim et al. 2019; Lee et al. 2018b; Ning et al. 2018; Xu et al. 2019] use adversarial loss instead of pixel-wise loss functions, but they still do not demonstrate results with high-quality textures. This is potentially because the problem of HDR reconstruction is constrained in the sense that the synthesized content should properly fit the input image using a soft mask. Unfortunately, GANs are known to have difficulty handling these scenarios [Bau et al. 2019]. In contrast, we propose a feature masking strategy and a more constrained VGG-based perceptual loss to effectively train our network and produce results with visually pleasing textures.

3 APPROACH

Our goal is to reconstruct an HDR image from a single LDR image by recovering the missing information in the saturated highlights. We achieve this using a convolutional neural network (CNN) that takes an LDR image as the input and estimates the missing HDR information in the bright regions. We compute the final HDR image by combining the well-exposed content of the input image and the output of the network in the saturated areas. Formally, we reconstruct the final HDR image $\hat{H}$, as follows:

$$\hat{H} = M \odot T^{\gamma} + (1 - M) \odot [\exp(\hat{Y}) - 1], \tag{1}$$

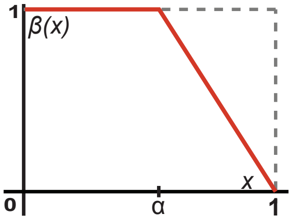

Fig. 2. We use this function to measure how well-exposed a pixel is. The value 1 indicates that the pixel is well-exposed, while 0 is assigned to the pixels that are fully saturated. In our implementation, we set the threshold $\alpha = 0.96$.

where the $\gamma = 2.0$ is used to transform the input image to the linear domain, and $\odot$ denotes element-wise multiplication. Here, T is the input LDR image in the range $[0, 1]$, $\hat{Y}$ is the network output in the logarithmic domain (Section 3.2), and M is a soft mask with values in the range $[0, 1]$ that defines how well-exposed each pixel is. We obtain this mask by applying the function $\beta(\cdot)$ (see Figure 2) to the input image, i.e., $M = \beta(T)$. In the following sections, we discuss our proposed feature masking approach, loss function, as well as the training process.

3.1 Feature Masking

Standard convolutional layers apply the same filter to the entire image to extract a set of features. This is reasonable for a wide range of applications, such as image super-resolution [Dong et al. 2015], style transfer [Gatys et al. 2016], and image colorization [Zhang et al. 2016], where the entire image contains valid information. However, in our problem, the input LDR image contains invalid information in the saturated areas. Since meaningful features cannot be extracted from the saturated contents, naïve application of standard convolution introduces ambiguity during training and leads to visible artifacts (Figure 10).

We address this problem by proposing a feature masking mechanism (Figure 3) that reduces the magnitude of the features generated from the invalid content (saturated areas). We do this by multiplying the feature maps in each layer by a soft mask, as follows:

$$Z_l = X_l \odot M_l, \tag{2}$$

where $X_l \in \mathbb{R}^{H \times W \times C}$ is the feature map of layer l with height H, width W, and C channels. $M_l \in [0, 1]^{H \times W \times C}$ is the mask for layer l and has values in the range $[0, 1]$. The value of one indicates that the features are computed from valid input pixels, while zero is assigned to the features that are computed from invalid pixels. Here, $l = 1$ refers to the input layer and, thus, $X_{l=1}$ is the input LDR image. Similarly, $M_{l=1}$ is the input mask $M = \beta(T)$. Note that, since our masks are soft, weak signals in the saturated areas are not discarded using this strategy. In fact, by suppressing the invalid pixels, these weak signals can propagate through the network more effectively.

Once the features of the current layer l are masked, the features in the next layer X_{l+1} are computed as usual:

$$X_{l+1} = \phi_l(W_l * Z_l + b_l), \tag{3}$$

where W_l and b_l refer to the weight and bias of the current layer, respectively. Moreover, ϕ_l is the activation function and * is the standard convolution operation.

We compute the masks at each layer by applying the convolutional filter to the masks at the previous layer (See Figure 4 for visualization of some of the masks). The basic idea is that since the features are computed by applying a series of convolutions, the same filters can be used to compute the contribution of the valid pixels

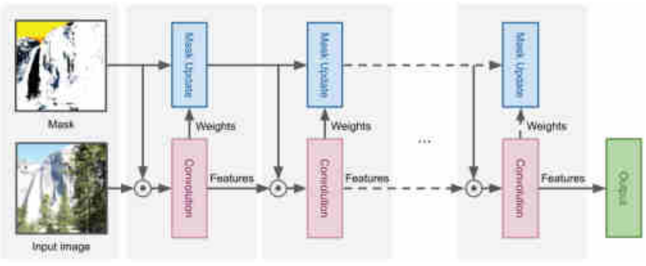

Fig. 3. Illustration of the proposed feature masking mechanism. The features at each layer are multiplied with the corresponding mask before going through the convolution process. The masks at each layer are obtained by updating the masks at the previous layer using Eq. 4.

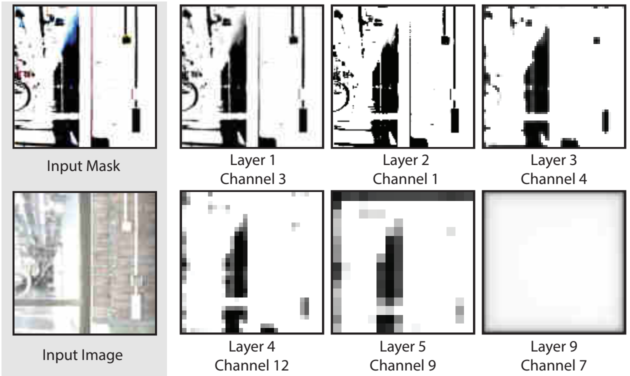

Fig. 4. On the left, we show the input image and the corresponding mask. On the right, we visualize a few masks at different layers of the network. Note that, as we move deeper through the network, the masks become blurrier and more uniform. This is expected since the receptive field of the features become larger in the deeper layers.

in the features. However, since the masks are in the range $[0, 1]$ and measure the percentage of the contributions, the magnitude of the filters is irrelevant. Therefore, we normalize the filter weights before convolving them with the masks as follows:

$$M_{l+1} = \left(\frac{|W_l|}{\|W_l\|_1 + \epsilon} \right) * M_l, \tag{4}$$

where $\|\cdot\|_1$ is the l_1 function and $|\cdot|$ is the absolute operator. Here, $|W_l|$ is a $\mathbb{R}^{H \times W \times C}$ tensor and $\|W_l\|_1$ is a $\mathbb{R}^{1 \times 1 \times C}$ tensor. To perform the division, we replicate the values of $\|W_l\|_1$ to obtain a tensor with the same size as $|W_l|$. The constant ϵ is a small value to avoid division by 0 (10^{-6} in our implementation).

Note that a couple of recent approaches have proposed strategies to overcome similar issues in image inpainting [Liu et al. 2018; Yu et al. 2019]. Specifically, Liu et al. [2018] propose to modify the convolution process to only apply the filter to the pixels with valid information. Unfortunately, this approach is specially designed for cases with binary masks. However, the masks in our application are soft and, thus, this method is not applicable. Yu et al. [2019] propose to multiply the features at each layer with a soft mask, similar to our feature masking strategy. The key difference is that their mask at each layer is learnable, and it is estimated using a small network from the features in the previous layer. Because of the additional parameters and complexity, training this approach on limited HDR images is difficult. Therefore, this approach is not able to produce high-quality HDR images (see Section 5.3).

3.2 Loss Function

The choice of the loss function is critical in each learning system. Our goal is to reconstruct an HDR image by synthesizing plausible textures in the saturated areas. Unfortunately, using only pixel-wise loss functions, as utilized by most previous approaches, the network tends to produce blurry images (Figure 12). Inspired by the recent image inpainting approaches [Han et al. 2019; Liu et al. 2018; Yang et al. 2017], we train our network using a VGG-based perceptual loss function. Specifically, our loss function is a combination of an HDR reconstruction loss L_r and a perceptual loss L_p, as follows:

$$L = \lambda_1 L_r + \lambda_2 L_p \tag{5}$$

where $\lambda_1 = 6.0$ and $\lambda_2 = 1.0$ in our implementation.

Reconstruction Loss: The HDR reconstruction loss is a simple pixel-wise l_1 distance between the output and ground truth images in the saturated areas. Since the HDR images could potentially have large values, we define the loss in the logarithmic domain. Given the estimated HDR image $\hat{Y}$ (in the log domain) and the linear ground truth image H, the reconstruction loss is defined as:

$$L_r = \|(1 - M) \odot (\hat{Y} - \log(H + 1))\|_1. \tag{6}$$

The multiplication by $(1 - M)$ ensures that the loss is computed in the saturated areas.

Perceptual Loss: Our perceptual term is a combination of the VGG and style loss functions as follows:

$$L_p = \lambda_3 L_v + \lambda_4 L_s. \tag{7}$$

In our implementation, we set $\lambda_3 = 1.0$ and $\lambda_4 = 120.0$. The VGG loss function L_v evaluates how well the features of the reconstructed image match with the features extracted from the ground truth. This allows the model to produce textures that are perceptually similar to the ground truth. This loss term is defined as follows:

$$L_v = \sum_l \|\phi_l(\mathcal{T}(\tilde{H})) - \phi_l(\mathcal{T}(H))\|_1 \tag{8}$$

where ϕ_l is the feature map extracted from the l^{th} layer of the VGG network. Moreover, the image $\tilde{H}$ is obtained by combining the information of the ground truth H in the well-exposed regions and the content of the network's output $\hat{Y}$ in the saturated areas using the mask M, as follows:

$$\tilde{H} = M \odot H + (1 - M) \odot \hat{Y}. \tag{9}$$

We use $\tilde{H}$ in our loss functions to ensure that the supervision is only provided in the saturated areas. Finally, $\mathcal{T}(\cdot)$ in Eq. 8 is a function that compresses the range to $[0, 1]$. Specifically, we use the differentiable μ-law range compressor:

$$\mathcal{T}(H) = \frac{\log(1 + \mu H)}{\log(1 + \mu)}, \tag{10}$$

where μ is a parameter defining the amount of compression ($\mu = 500$ in our implementation). This is done to ensure that the input to the VGG network is similar to the ones that it has been trained on.

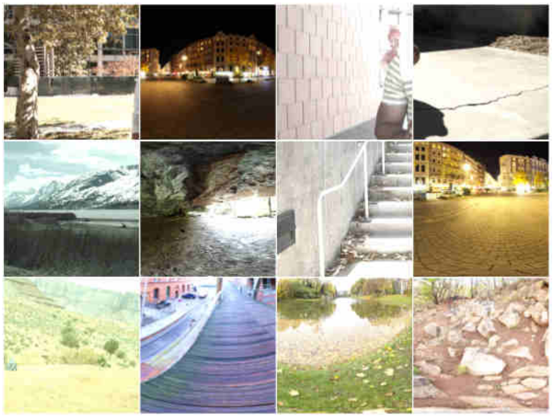

Fig. 5. A few example patches selected by our patch sampling approach. These are challenging examples as the HDR images corresponding to these patches contain complex textures in the saturated areas.

The style loss in Eq. 7 (L_s) captures style and texture by comparing global statistics with a Gram matrix [Gatys et al. 2015] collected over the entire image. Specifically, the style loss is defined as:

$$L_s = \sum_l \|G_l(\mathcal{T}(\tilde{H})) - G_l(\mathcal{T}(H))\|_1, \tag{11}$$

where $G_l(X)$ is the Gram matrix of the features in layer l and is defined as follows:

$$G_l(X) = \frac{1}{K_l}\phi_l(X)^T\phi_l(X). \tag{12}$$

Here, K_l is a normalization factor computed as $C_lH_lW_l$. Note that, the feature ϕ_l is a matrix of shape $(H_lW_l) \times C_l$ and, thus, the Gram matrix has a size of $C_l \times C_l$. In our implementation, we use the VGG-19 [Simonyan and Zisserman 2015] network and extract features from layers pool1, pool2 and pool3.

3.3 Inpainting Pre-training

Training our system is difficult as large-scale HDR image datasets are currently not available. Existing techniques [Eilertsen et al. 2017] overcome this limitation by pre-training their network on simulated HDR images that are created from standard image datasets like the MIT Places [Zhou et al. 2014]. They then fine-tune their network on real HDR images. Unfortunately, our network is not able to learn to synthesize plausible textures with this strategy (see Figure 11), as the saturated areas are typically in the bright and smooth regions.

To address this problem, we propose to pre-train our network on image inpainting tasks. Intuitively, during inpainting, our network leverages a large number of training data to learn an appropriate internal representation that is capable of synthesizing visually pleasing textures. In the HDR fine-tuning stage, the network adapts the learned representation to the HDR domain to be able to synthesize HDR textures. We follow Liu et al.'s approach [2018] and use their loss function and mask generation strategy during pre-training. Note that we still use our feature masking mechanism for pre-training, but the input masks are binary. We fine-tune the network on real HDR images using the loss function, discussed in Section 3.2.

One major problem is that the majority of the bright areas in the HDR examples are smooth and textureless. Therefore, during fine-tuning, the network adapts to these types of patches and, as a result, has difficulty producing textured results (see Figure 11). In the next section, we discuss our strategy to select textured and challenging patches.

Algorithm 1 Patch Sampling

```
procedure PatchMetric($H$, $M$)
    $H$: HDR image, $M$: Mask
    $\sigma_c$ = 100.0                              ▷ Bilateral filter color sigma
    $\sigma_s$ = 10.0                               ▷ Bilateral filter space sigma
    $I$ = RgbToGray($H$)
    L = log($I$ + 1)
    B = bilateralFilter($L, \sigma_c, \sigma_s$)    ▷ Base Layer
    D = L - B                                       ▷ Detail Layer
    $G_x$ = getGradX($D$)
    $G_y$ = getGradY($D$)
    G = abs($G_x$) + abs($G_y$)
    return mean($G \odot (1 - M)$)
```

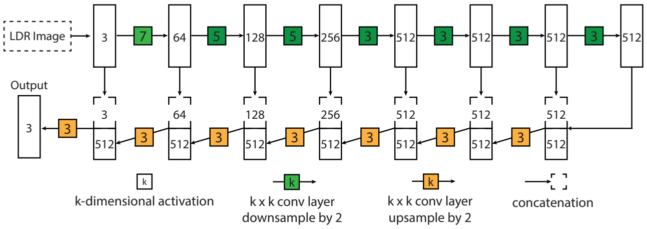

Fig. 6. The proposed network architecture. The model takes as input the RGB LDR image and outputs an HDR image. We use a feature masking mechanism in all the convolutional layers.

3.4 Patch Sampling

Our goal is to select the patches that contain texture in the saturated areas. We perform this by first computing a score for each patch and then choosing the patches with a high score. The main challenge here is finding a good metric that properly detects the textured patches. One way to do this is to compute the average of the gradient magnitude in the saturated regions. However, since our images are in HDR and can have large values, this approach can detect a smooth region with bright highlights as textured.

To avoid this issue, we propose to first decompose the HDR image into base and detail layers using a bilateral filter [Durand and Dorsey 2002]. We use the average of the gradients (Sobel operator) of the detail layer in the saturated areas as our metric to detect the textured patches. We consider all the patches with a mean gradient above a certain threshold (0.85 in our implementation) as textured, and the rest are classified as smooth. Since the detail layer only contains variations around the base layer, this metric can effectively measure the amount of textures in an HDR patch. Figure 5 shows example of patches selected using this metric. As shown in Figure 11, this simple patch sampling approach is essential for synthesizing HDR images with sharp and artifact-free details in the saturated areas. The summary of our patch selection strategy is listed in Algorithm 1.

4 IMPLEMENTATION

Architecture. We use a network with U-Net architecture [Ronneberger et al. 2015], as shown in Figure 6. We use the feature masking strategy in all the convolutional layers and up-sample the

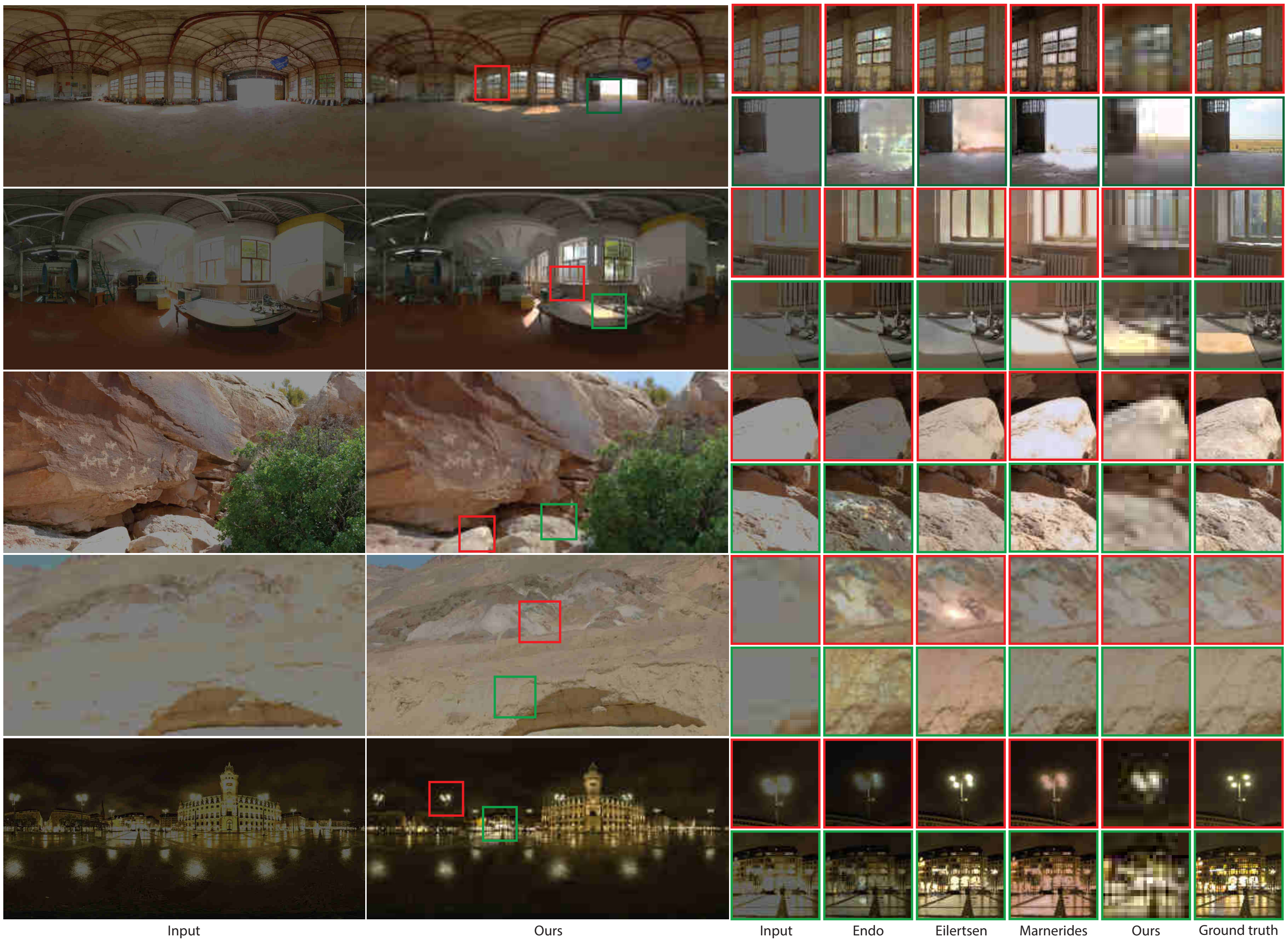

Fig. 7. We compare our method against state-of-the-art approaches of Endo et al. [2017], Eilertsen et al. [2017], and Marnerides et al. [2018] on a diverse set of synthetic scenes. Our method is able to synthesize textures in the saturated areas better than the other approaches (rows one to four), while producing results with similar or better quality in the bright highlights (fifth row).

features in the decoder using nearest neighbor. All the encoder layers use Leaky ReLU activation function [Maas et al. 2013]. On the other hand, we use ReLU [Nair and Hinton 2010] in all the decoder layers, with the exception of the last one, which has a linear activation function. We use skip connections between all the encoder layers and their corresponding decoder layers.

Dataset. We use different datasets for each training step. For the image inpainting step, we use the MIT Places [Zhou et al. 2014] dataset with the original train, test, and validation splits. We choose Places for this step because it contains a large number of scenes ($\sim 2.5M$ images) with diverse textures. We use the method of Liu et al. [2018] to generate masks of random streaks and holes of arbitrary shapes and sizes. On the other hand, for the HDR fine-tuning step, we collect approximately 2,000 HDR images from 735 HDR images and 34 HDR videos. From each HDR image, we extract 250 random patches of size 512×512 and generate the input LDR patches following the approach by Eilertsen et al. [2017]. We then select a subset of these patches using our patch selection strategy. We also discard patches with no saturated content, since they do not provide any source of learning to the network. Our final training dataset is a set of 100K input and corresponding ground truth patches.

Training. We initialize our network using the Xavier approach [Glorot and Bengio 2010] and train it on image inpainting task until convergence. We then fine-tune the network on HDR reconstruction. We train the network with a learning rate of 2×10^{-4} in both stages. However, during the second stage, we reduce the learning rate by a factor of 2.0 when the optimization plateaus. The training process is performed until convergence. Both inpainting and HDR fine-tuning stages are optimized using Adam [Kingma and Ba 2015] with the default parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$ and mini-batch size of 4. The entire training takes approximately 11 days on a machine with an Intel Core i7, 16GB of memory, and an Nvidia GTX 1080 Ti GPU.

5 RESULTS

We implement our network in PyTorch [Paszke et al. 2019], but write the data pre-processing, data augmentation, and patch sampling

Table 1. Numerical comparison in terms of mean square error (MSE) and HDR-VDP-2 [Mantiuk et al. 2011] against existing learning-based single image HDR reconstruction approaches.

Method	MSE	HDR-VDP-2
Endo et al. [2017]	0.0390	55.67
Eilertsen et al. [2017]	0.0387	59.11
Marnerides et al. [2018]	0.0474	54.31
Ours	**0.0356**	**63.18**

code in C++. We implement the feature masking mechanism using the existing standard convolutional layer in PyTorch. We compare our approach against three existing learning-based single image HDR reconstruction approaches of Endo et al. [2017], Eilertsen et al. [2017], and Marnerides et al. [2018]. We use the source code provided by the authors to generate the results for all the other approaches.

5.1 Synthetic Images

We begin by quantitatively comparing our approach against the other methods in terms of mean squared error (MSE) and HDR-VDP-2 [Mantiuk et al. 2011] in Table 1. The errors are computed on a test set of 75 randomly selected HDR images, with resolutions ranging from 1024 × 768 to 2084 × 2844. We generate the input LDR images using various camera curves and exposures, similar to the approach by Eilertsen et al. [2017]. We compute the MSE values on the gamma corrected images and HDR-VDP-2 scores are obtained on the linear HDR images. As seen, our method produces significantly better results, which demonstrate the ability of our network to accurately recover the full range of luminance.

Next, we compare our approach against the other methods on five challenging scenes in Figure 7. Overall other approaches are not able to synthesize texture and produce results with blurriness, discoloration, and checkerboard artifacts. However, our approach can effectively utilize the information in the non-saturated color channels and the contextual information to synthesize visually pleasing textures. It is worth noting that although our approach has been trained using a perceptual loss, it can still properly recover the bright highlights. For example, our results in Figure 7 (fifth row) are similar to Eilertsen et al. [2017] and better than Endo et al. [2017] and Marnerides et al. [2018].

We also demonstrate that our approach can consistently generate high-quality results on images with different amount of saturated areas in Figure 8. As can be seen, the results of all the other approaches degrade quickly by increasing the percentage of the saturated pixels in the input LDR image. On the other hand, our approach is able to produce high-quality results with sharp details and bright highlights in all the cases.

5.2 Real Images

We show the generality of our approach by producing results on a set of real images, captured with standard cameras, in Figure 9. Specifically, the top three images are from Google HDR+ dataset [Hasinoff et al. 2016], captured with a variety of smartphones, such as Nexus 5/6/5X/6P, Pixel, and Pixel XL. The image in the last row is captured by a Canon 5D Mark IV camera. All the other approaches are not able to properly reconstruct the saturated regions, producing results with discoloration and blurriness, as indicated by the arrows. On the other hand, our method is able to properly increase the dynamic range by synthesizing realistic textures.

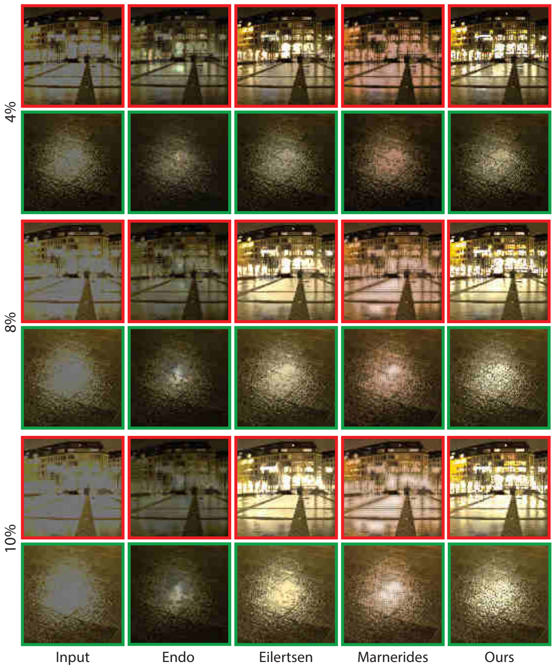

Fig. 8. We compare the performance of the proposed method against previous methods for various amounts of saturated areas. The numbers indicate the percentage of the total number of pixels that are saturated in the input. Although our method slightly degrades as the saturation increases, we consistently present better results than the previous methods.

Table 2. We evaluate the effectiveness of our masking and pre-training strategies by comparing against other alternatives in terms of MSE and HDR-VDP-2 [Mantiuk et al. 2011]. Here, SConv, GConv, IMask, and FMask refer to standard convolution, gated convolution [Yu et al. 2019], only masking the input image, and our full feature masking approach, respectively. Moreover, Inp. pre-training and HDR pre-training correspond to our proposed pre-training on inpainting and HDR reconstruction tasks, respectively.

Method (Masking + Pre-training)	MSE	HDR-VDP-2
SConv + HDR pre-training	0.0402	58.43
SConv + Inp. pre-training	0.0374	60.03
GConv + HDR pre-training	0.0398	53.32
GConv + Inp. pre-training	0.1017	43.13
IMask + HDR pre-training	0.0398	58.39
IMask + Inp. pre-training	0.0369	61.27
FMask + HDR pre-training	0.0393	58.81
FMask + Inp. pre-training (Ours)	**0.0356**	**63.18**

5.3 Ablation Studies

Inpainting Pre-training. We begin studying the effect of the proposed inpainting pre-training step by comparing it against the commonly-used synthetic HDR pre-training in Table 2 and Figure 11. As seen, our pre-training ("FMask + Inp. pre-training (Ours)") performs better than HDR pre-training ("FMask + HDR pre-training") both numerically and visually. Specifically, as shown in Figure 11, our network using inpainting pre-training is able to learn better features and synthesizes sharp textures in the saturated areas.

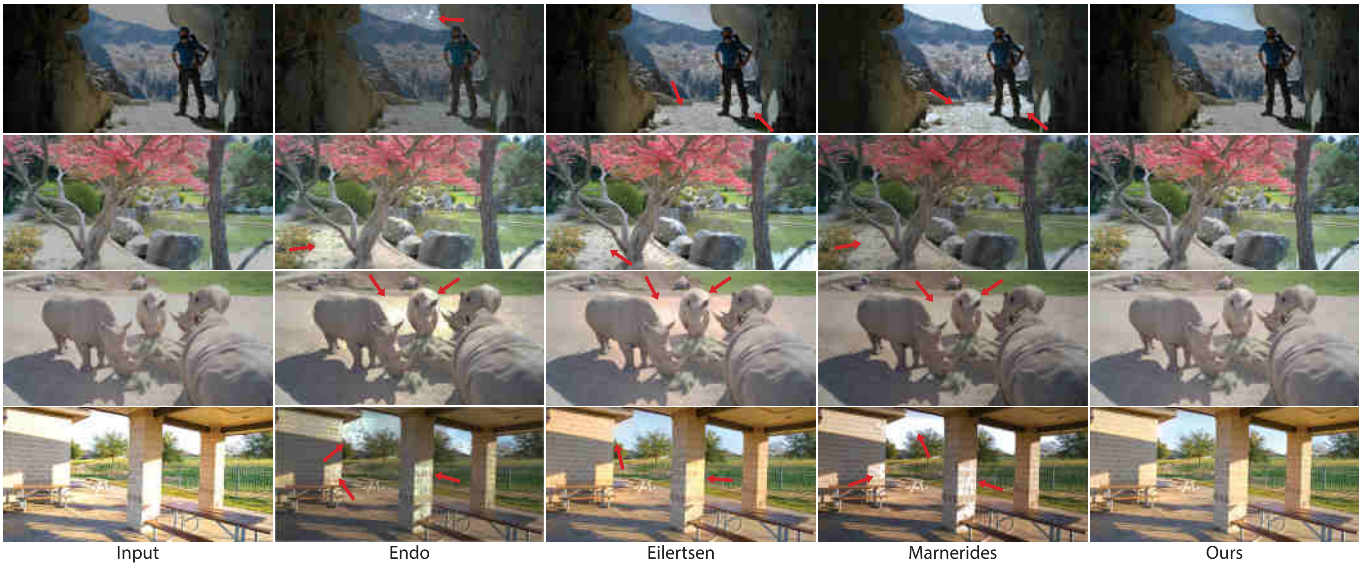

Fig. 9. Comparison against state-of-the-art approaches on images captured by standard cameras. Zoom in to the electronic version to see the differences.

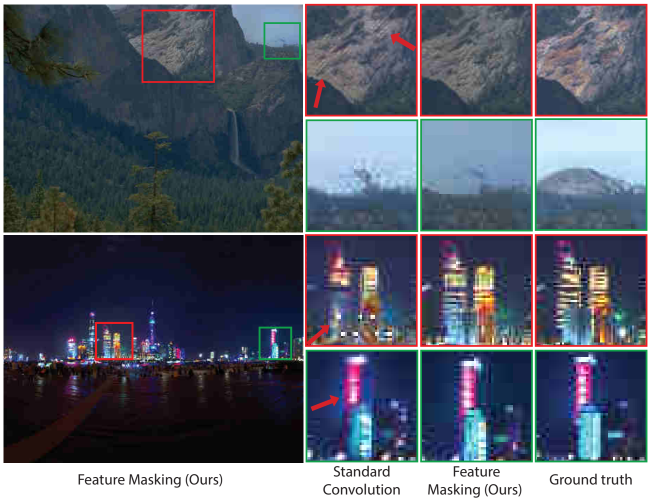

Fig. 10. In regions with both saturated and well-exposed content (boundaries of sky and mountain and bright building lights), the response of the invalid saturated areas in standard convolution dominates the feature maps. Therefore, the network cannot properly utilize the content of the valid regions, introducing high frequency checkerboard artifacts (top row) and blurriness and halo (bottom row). Our approach suppresses the features from the saturated content and allows the network to synthesize the image using the well-exposed information.

Feature Masking. Here, we compare our feature masking strategy against several other approaches in Table 2. Specifically, we compare our method against standard convolution (SConv), gated convolution [Yu et al. 2019] (GConv), and the simpler version of our masking strategy where the mask is only applied to the input (IMask). For completeness, we include the result of each method with both inpainting and HDR pre-training. As seen, our masking strategy is considerably better than the other methods. It is worth noting that unlike other methods, the performance of gated convolution with inpainting pre-training is worse than HDR pre-training. This is mainly because gated convolution estimates the masks at each layer using a separate set of networks which become unstable after transitioning from inpainting pre-training to HDR fine-tuning.

We also visually compare our feature masking method against standard convolution in Figure 10. Standard convolution produces results with checkerboard artifacts (top) and halo and blurriness (bottom), while our network with feature masking produces considerably better results. Moreover, we visually compare our approach against other masking strategies in Figure 11. Note that, for each masking strategy, we only show the combination of masking and pre-training that produces the best numerical results in Table 2, i.e., gated convolution (GConv) with HDR pre-training and input masking (IMask) with inpainting pre-training. Gated convolution is not able to produce high frequency textures in the saturated areas. Input masking performs reasonably well, but still introduces noticeable artifacts. Our feature masking method, however, is able to synthesize visually pleasing textures.

Patch Sampling. We show our result without patch sampling (Section 3.4) to demonstrate its effectiveness in Figure 11. As seen, by training on the textured patches (ours), the network is able to synthesize textures with more details and fewer objectionable artifacts.

Loss Function. Finally, we compare the proposed perceptual loss function against a simple pixel-wise (l_1) loss. As seen in Figure 12, using only the pixel-wise loss function our network tends to produce blurry images, while the network trained using the proposed perceptual loss function can produce visually realistic textures in the saturated regions.

6 LIMITATIONS AND FUTURE WORK

Single image HDR reconstruction is a notoriously challenging problem. Although our method can recover the luminance and hallucinate textures, it is not always able to reconstruct all the details. One of such cases is shown in Figure 13 (top), where our approach fails to reconstruct the wrinkles on the curtain. Nevertheless, our result is still better than the other approaches as they overestimate the

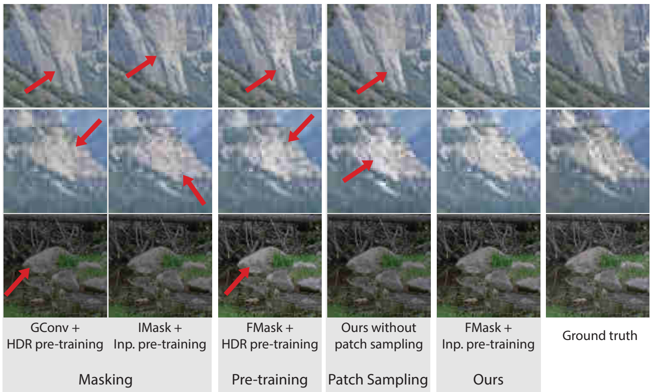

Fig. 11. From left to right, we compare our method against two other masking strategies as well as a pre-training method, and evaluate the effect of patch sampling. Here, GConv, IMask, and FMask refer to gated convolution [Yu et al. 2019], only masking the input image, and our full feature masking method, respectively. Moreover, Inp. pre-training refers to our proposed pre-training on inpainting task.

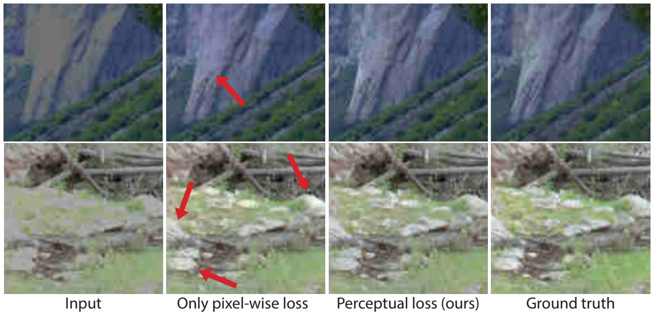

Fig. 12. We compare the results of our network trained with only a pixel-wise loss (l_1) and the proposed perceptual loss. Using the perceptual loss function, our network can synthesize visually realistic textures, while the network trained with only a pixel-wise loss produces blurry results.

brightness of the window and produce blurry results. Moreover, as shown in Figure 13 (middle), when the input lacks sufficient information about the underlying texture, our method could potentially introduce patterns that do not exist in the ground truth image. Despite that, our result is still comparable to or better than the other approaches. Additionally, in some cases, our method reconstructs the saturated areas with an incorrect color, as shown in Figure 13 (bottom). It is worth noting that the network reconstruct the building in blue since trees and skies are usually next to each other in the training data. As seen, other approaches also reconstruct parts of the building in blue color.

Although our network can be used to reconstruct an HDR video from an LDR video, our result is not temporally stable. This is mainly because we synthesize the content of every frame independently. In the future, it would be interesting to address this problem through temporal regularization [Eilertsen et al. 2019]. Moreover, we would like to experiment with the architecture of the networks to increase the efficiency of our approach and reduce the memory footprint.

7 CONCLUSION

We present a novel learning-based system for single image HDR reconstruction using a convolutional neural network. To alleviate the artifacts caused by conditioning the convolutional layer on the saturated pixels, we propose a feature masking mechanism with an automatic mask updating process. We show that this strategy reduces halo and checkerboard artifacts caused by standard convolutions. Moreover, we propose a perceptual loss function that is designed specifically for the HDR reconstruction application. By minimizing this loss function during training, the network is able to synthesize visually realistic textures in the saturated areas. We further propose to train the system in two stages where we pre-train the network on inpainting before fine-tuning it on HDR generation. To encourage the network to synthesize textures, we propose a sampling strategy to select challenging patches in the HDR examples. Our model can robustly handle saturated areas and can reconstruct high-frequency details in a realistic manner. We show quantitatively and qualitatively that our method outperforms previous methods on both synthetic and real-world images.

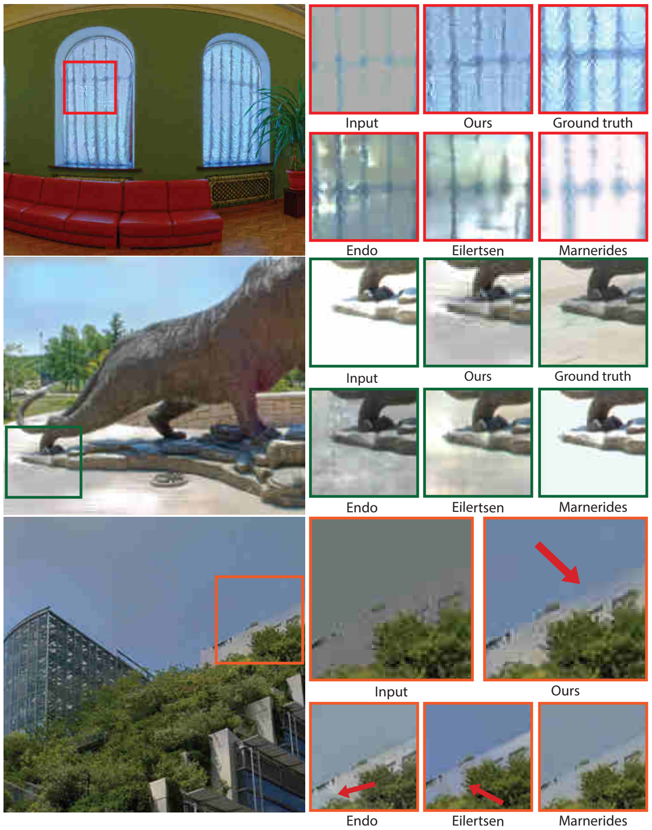

Fig. 13. Failure cases of our approach. From top to bottom, our method fails to reconstruct the wrinkles on the curtain, introduces textures that are not in the ground truth, and incorrectly reconstructs the building with sky color. Note that, the top two examples are synthetic, but the bottom one is real for which we do not have access to the ground truth image.

ACKNOWLEDGMENTS

We thank the reviewers for their constructive comments. M. Santos is funded by the Brazilian agency CNPQ grant 161268/2018-8. T. Ren is partially supported by FACEPE grant APQ-0192- 1.03/14. N. Kalantari is in part funded by a TAMU T3 grant 246451.

REFERENCES

Francesco Banterle, Patrick Ledda, Kurt Debattista, and Alan Chalmers. 2006. Inverse tone mapping. In *Proceedings of the 4th International Conference on Computer Graphics and Interactive Techniques in Australasia and Southeast Asia*. ACM, 349–356.

David Bau, Hendrik Strobelt, William Peebles, Jonas Wulff, Bolei Zhou, Jun-Yan Zhu, and Antonio Torralba. 2019. Semantic Photo Manipulation with a Generative Image Prior. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH)* 38, 4 (2019).

Cambodge Bist, Rémi Cozot, Gérard Madec, and Xavier Ducloux. 2017. Tone expansion using lighting style aesthetics. *Computers & Graphics* 62 (2017), 77–86.

Paul Debevec. 2005. A median cut algorithm for light probe sampling. In *ACM SIGGRAPH 2005 Posters*. ACM, 66.

PE Debevec and J Malik. 1997. Recovering high dynamic range images. In *Proceeding of the SPIE: Image Sensors*, Vol. 3965. 392–401.

Piotr Didyk, Rafal Mantiuk, Matthias Hein, and Hans-Peter Seidel. 2008. Enhancement of bright video features for HDR displays. In *Computer Graphics Forum*, Vol. 27. Wiley Online Library, 1265–1274.

Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. 2015. Image super-resolution using deep convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38, 2 (2015), 295–307.

Frédo Durand and Julie Dorsey. 2002. Fast bilateral filtering for the display of high-dynamic-range images. In *Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques*. 257–266.

Gabriel Eilertsen, Joel Kronander, Gyorgy Denes, Rafał K Mantiuk, and Jonas Unger. 2017. HDR image reconstruction from a single exposure using deep CNNs. *ACM Transactions on Graphics (TOG)* 36, 6 (2017), 178.

Gabriel Eilertsen, RafałMantiuk, and Jonas Unger. 2019. Single-frame Regularization for Temporally Stable CNNs. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Yuki Endo, Yoshihiro Kanamori, and Jun Mitani. 2017. Deep reverse tone mapping. *ACM Transactions on Graphics (TOG)* 36, 6 (2017), 177–1.

Leon A Gatys, Alexander S Ecker, and Matthias Bethge. 2015. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576* (2015).

Leon A Gatys, Alexander S Ecker, and Matthias Bethge. 2016. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2414–2423.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS)*. 249–256.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems (NeurIPS)*. 2672–2680.

Xintong Han, Zuxuan Wu, Weilin Huang, Matthew R Scott, and Larry S Davis. 2019. FiNet: Compatible and Diverse Fashion Image Inpainting. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 4481–4491.

Samuel W Hasinoff, Dillon Sharlet, Ryan Geiss, Andrew Adams, Jonathan T Barron, Florian Kainz, Jiawen Chen, and Marc Levoy. 2016. Burst photography for high dynamic range and low-light imaging on mobile cameras. *ACM Transactions on Graphics (TOG)* 35, 6 (2016), 192.

Geoffrey E Hinton and Ruslan R Salakhutdinov. 2006. Reducing the dimensionality of data with neural networks. *Science* 313, 5786 (2006), 504–507.

Jun Hu, Orazio Gallo, Kari Pulli, and Xiaobai Sun. 2013. HDR deghosting: How to deal with saturation?. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 1163–1170.

Nima Khademi Kalantari and Ravi Ramamoorthi. 2017. Deep high dynamic range imaging of dynamic scenes. *ACM Transactions on Graphics (TOG)* 36, 4 (2017), 144–1.

Sing Bing Kang, Matthew Uyttendaele, Simon Winder, and Richard Szeliski. 2003. High dynamic range video. In *ACM Transactions on Graphics (TOG)*, Vol. 22. ACM, 319–325.

Soo Ye Kim, Jihyong Oh, and Munchurl Kim. 2019. Jsi-gan: Gan-based joint super-resolution and inverse tone-mapping with pixel-wise task-specific filters for UHD HDR video. *arXiv preprint arXiv:1909.04391* (2019).

Diederick P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*.

Rafael P Kovaleski and Manuel M Oliveira. 2014. High-quality reverse tone mapping for a wide range of exposures. In *2014 27th SIBGRAPI Conference on Graphics, Patterns and Images*. IEEE, 49–56.

Hayden Landis. 2002. Production-ready global illumination. *SIGGRAPH Course Notes* 16, 2002 (2002), 11.

Siyeong Lee, Gwon Hwan An, and Suk-Ju Kang. 2018a. Deep chain hdri: Reconstructing a high dynamic range image from a single low dynamic range image. *IEEE Access* 6 (2018), 49913–49924.

Siyeong Lee, Gwon Hwan An, and Suk-Ju Kang. 2018b. Deep Recursive HDRI: Inverse Tone Mapping using Generative Adversarial Networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 596–611.

Guilin Liu, Fitsum A Reda, Kevin J Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. 2018. Image inpainting for irregular holes using partial convolutions. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 85–100.

Gonzalo Luzardo, Jan Aelterman, Hiep Luong, Wilfried Philips, Daniel Ochoa, and Sven Rousseaux. 2018. Fully-Automatic Inverse Tone Mapping Preserving the Content Creator's Artistic Intentions. In *2018 Picture Coding Symposium (PCS)*. IEEE, 199–203.

Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. 2013. Rectifier nonlinearities improve neural network acoustic models. In *Proceedings of International Conference on Machine Learning (ICML)*, Vol. 30. 3.

Rafat Mantiuk, Kil Joong Kim, Allan G Rempel, and Wolfgang Heidrich. 2011. HDR-VDP-2: A calibrated visual metric for visibility and quality predictions in all luminance conditions. *ACM Transactions on Graphics (TOG)* 30, 4 (2011), 40.

Demetris Marnerides, Thomas Bashford-Rogers, Jonathan Hatchett, and Kurt Debattista. 2018. ExpandNet: A deep convolutional neural network for high dynamic range expansion from low dynamic range content. In *Computer Graphics Forum*, Vol. 37. Wiley Online Library, 37–49.

Morgan McGuire, Wojciech Matusik, Hanspeter Pfister, Billy Chen, John F Hughes, and Shree K Nayar. 2007. Optical splitting trees for high-precision monocular imaging. *IEEE Computer Graphics and Applications* 27, 2 (2007), 32–42.

Vinod Nair and Geoffrey E Hinton. 2010. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML)*. 807–814.

Shiyu Ning, Hongteng Xu, Li Song, Rong Xie, and Wenjun Zhang. 2018. Learning an inverse tone mapping network with a generative adversarial regularizer. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 1383–1387.

Tae-Hyun Oh, Joon-Young Lee, Yu-Wing Tai, and In So Kweon. 2014. Robust high dynamic range imaging by rank minimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37, 6 (2014), 1219–1232.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems (NeurIPS)*. 8024–8035.

Allan G Rempel, Matthew Trentacoste, Helge Seetzen, H David Young, Wolfgang Heidrich, Lorne Whitehead, and Greg Ward. 2007. Ldr2hdr: on-the-fly reverse tone mapping of legacy video and photographs. In *ACM Transactions on Graphics (TOG)*, Vol. 26. ACM, 39.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-NET: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-assisted Intervention*. Springer, 234–241.

Pradeep Sen, Nima Khademi Kalantari, Maziar Yaesoubi, Soheil Darabi, Dan B Goldman, and Eli Shechtman. 2012. Robust patch-based HDR reconstruction of dynamic scenes. *ACM Transactions on Graphics (TOG)* 31, 6 (2012), 203–1.

Karen Simonyan and Andrew Zisserman. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *International Conference on Learning Representations (ICLR)*.

Michael D Tocci, Chris Kiser, Nora Tocci, and Pradeep Sen. 2011. A versatile HDR video production system. In *ACM Transactions on Graphics (TOG)*, Vol. 30. ACM, 41.

Lvdi Wang, Li-Yi Wei, Kun Zhou, Baining Guo, and Heung-Yeung Shum. 2007. High dynamic range image hallucination. In *Proceedings of the 18th Eurographics Conference on Rendering Techniques*. Eurographics Association, 321–326.

Shangzhe Wu, Jiarui Xu, Yu-Wing Tai, and Chi-Keung Tang. 2018. Deep high dynamic range imaging with large foreground motions. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 117–132.

Yucheng Xu, Shiyu Ning, Rong Xie, and Li Song. 2019. Gan Based Multi-Exposure Inverse Tone Mapping. In *2019 IEEE International Conference on Image Processing (ICIP)*. IEEE, 1–5.

Chao Yang, Xin Lu, Zhe Lin, Eli Shechtman, Oliver Wang, and Hao Li. 2017. High-resolution image inpainting using multi-scale neural patch synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 6721–6729.

Xin Yang, Ke Xu, Yibing Song, Qiang Zhang, Xiaopeng Wei, and Rynson WH Lau. 2018. Image correction via deep reciprocating HDR transformation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 1798–1807.

Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. 2019. Free-form image inpainting with gated convolution. In *Proceedings of the IEEE International Conference on Computer Vision*. 4471–4480.

Richard Zhang, Phillip Isola, and Alexei A Efros. 2016. Colorful image colorization. In *European Conference on Computer Vision (ECCV)*. Springer, 649–666.

Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. 2014. Learning deep features for scene recognition using places database. In *Advances in Neural Information Processing Systems (NeurIPS)*. 487–495.