Reconstructing HDR Images using Non-Learning and Deep Learning Based Multi-exposure Image Synthesis Techniques

Bin (Claire) Zhang, and Megan Zhang

Abstract—High dynamic range (HDR) imaging has gained its popularity in the past two decades. With the enhancement of HDR synthesis algorithms and the improvement in imaging systems hardware, this technology has become increasingly accessible and thus influential. For this project, we propose and examine multiple exposure HDR synthesis techniques, including non-learining based and deep learning based algorithms. We compare the performance of those methods to better understand what are the components that make each of the algorithms effective or less than ideal. For the non-learning based technique, the fusion computations are done in both the RGB and the YCbCr color space. For the deep learning based method, we train a U-Net architecture on the multi-exposure HDR image synthesis task with different training schemes and parameter choices, including optimizer, loss function, and regularization.

Index Terms—Computational Photography, High Dynamic Range Imaging, Deep Learning

1 Introduction

HIGH dynamic range (HDR) imaging has been well integrated in modern technology since the first introduction of its core principle in the late 20-th century. HDR imaging technique attempts to circumvent the physical limitations of modern imaging systems and produce a greater range of luminance which is similar to that experienced by the human visual system. Its ubiquitousness is reflected in its variety of applications, such as autonomous vehicles [1] and consumer photography [2].

Multi-exposure HDR capture is one of the most popular techniques used to combine multiple low dynamic range (LDR) images of the same scene under different exposures and convert them into a single HDR image. There exist multiple viable methods to achieve such results. These approaches can be divided into two main categories, non-learning based and learning based. Non-learning based algorithms often involve finding the appropriate weighting function to better combine the multiple exposures, whereas learning based methods include the use of deep learning model and large datasets to train a neural network that performs well on such task.

In this project, we explore two HDR synthesis techniques, one of which is non-learning based and the other learning based. The non-learning based technique is adapted from the one proposed in [3], where the luminance and chrominance of LDR images are processed separately in the YCbCr color space instead of the RGB color space. We proposed a learning based HDR fusion method using a neural network model with the U-Net [4] architecture.

Analyzing the performance of the two methods gives us more insights into the performance of the two proposed methods. The advantages and shortcomings of both approaches are discussed to give readers a more comprehensive overview of these different HDR fusion techniques.

2 Related Work

2.1 Non-learning based methods

Debevec's algorithm, a non-learning based algorithm presented in [5], fuses multiple LDR images to reconstruct the HDR radiance maps. More specifically, it is achieved by giving higher weight to exposures where the pixel value is closer to the middle of the dynamic range of each exposure. One of the limitations of this approach, however, is that it requires information about the exposure time of the LDR images. This additional requirement puts restrictions on the application of this algorithm, making it less versatile and less flexible.

Other non-learning based methods may have additional steps for pre-processing the LDR images or post-processing the HDR image, but the main objective remains the same - that is, to find the optimal weighting function for the LDR captures. Mertens et al. [6], for instance, proposes a method that select the "good" pixels from the input LDR sequence and merge them into the final result, guided by some computed perceptual quality measure that encodes desirable qualities, such as saturation and constrast.

The HybridHDR method proposed in [3], takes a slightly different approach and also processes the LDR images in the YCbCr color space after image alignment. The algorithm separates luminance (Y) channel from the chrominance (CbCr) channels and uses different method to merge images in these channels. For the luminance channel, it performs fusion in the independent component analysis (ICA) domain using the method proposed in [7]. The color channels are extracted from the fused image using the approach presented in [6].

- *Claire and Megan are within the Department of Electrical Engineering, Stanford University, California, 94305.*
E-mail: zhangbin@stanford.edu, lzhang8@stanford.edu

2.2 Learning based methods

Learning-based multi-exposure HDR image reconstruction usually takes two steps. The first step is to align the LDR images to achieve good LDR pixel correspondance and prevent visible artifact in the reconstruction. Common methods for image alignment include augmenting classical optical flow algorithm [8], using optical flow networks like FlowNet [9], taking advantage of the attention mechanism to achieve alignment in the feature space [10], and more.

The second step is to fuse multiple LDR exposures. This is done in many fashions with the help of learning methods. To avoid hand-crafting features for image fusion and to have robustness to varying input conditions, DeepFuse [11] uses an unsupervised deep learning framework, which uses a no-reference quality metric as loss function and fuses common low level features extracted by the neural network for each image to generate the result. The approach is also powered by a large dataset of multi-exposure image stacks collected by the team, in order to circumvent using ground truth. This approach has the trade-off between learning a robust image representation and expensiveness of gathering and training without supervision on a large dataset. Generative adversarial network (GAN) based approach, which is another compute intensive method, uses the generator to produce a fused image and tries to fool the discriminator, aiming to make the predicted fused image have a similar probability distribution as the ground truth HDR images. [12] as an GAN-based example, even adds in self-attention mechanism to enable long-range dependency learning from the input images.

Our project assumes that the input LDR images are already aligned. Therefore our task of interest lies in the LDR-HDR conversion. We observe that, despite many complex model architecture and training framework choices, all learning-based methods rely on the network to be able to reconstruct an output of the same size as the input, a reconstruction task for which U-Net [4] is a great candidate. We then condense our question to, when the architecture is fixed, what training scheme and evaluation method can help the learning method to better fuse LDR images to HDR result.

2.3 Regularization for Image Quality

Designing or selecting an appropriate regularization method for the objective function is an essential part of many deep learning based algorithms. Total variation (TV) [13] is a widely used constrained optimization algorithm that suppresses noise from images. Subsequently, more approaches have been proposed to overcome some of the limitations found in the TV algorithm. High-order TV (HOTV) [14] and infimal-convolution TV (ICTV) [15] aim to avoid staircase artifacts caused by using the conventional TV regularization.

3 Dataset

We use the LDR-HDR pair dataset originally published in [16] and download the data content from [17]. There are 450 scenes captured by a DSLR camera in the autoexposure bracket mode at -2, 0, and +2 EV and at 1024x1024 resolution. The dataset covers various scenes, from outdoors to indoors, landscapes to buildings, etc. Ground truth HDR images are generated by fusing the three LDR images using the algorithm proposed in [18]. Figure 1 shows an example set of LDR images and the correspnding HDR ground truth.

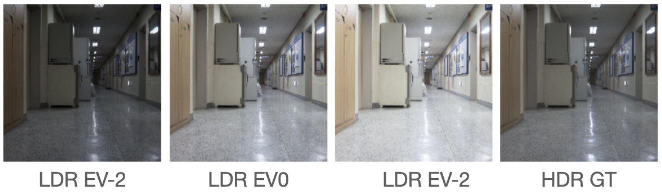

Fig. 1. Example input LDR images and ground truth HDR radiance maps in the dataset.

Training, validation, and test sets are split at the ratio of 7:2:1. Due to compute constraints, the images are resized to 256x256 using nearest neighbor interpolation. Input HDR images to the neural network have their values normalized to [0,1]. The output HDR radiance maps are clipped with a lower-bound of 0 and then go through inverse gamut mapping with $\gamma = 2.2$ to be saved as image files for visualization.

4 Proposed Methods

4.1 Non-Learning Based methods

Inspired by the HybridHDR algorithm proposed in [3] and also by the fact that human visual system is more sensitive to luminance/brightness and is less attentive to details and errors in the color channels, we propose a non-learning based method, incorporated with the Debevec's method, where processing is performed differently to the luminance and chrominance channels in the YCbCr color space (see Figure 2).

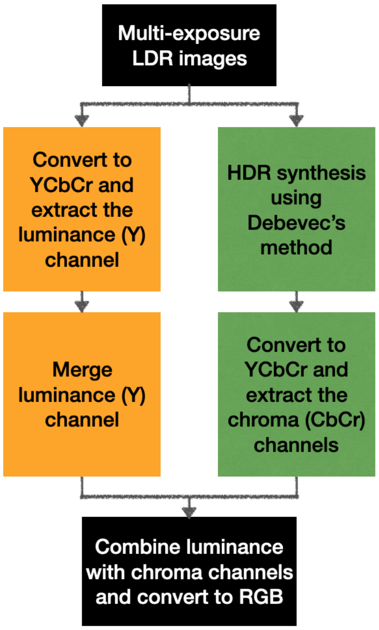
Multi-exposure LDR images
Convert to YCbCr and extract the luminance (Y) channel
HDR synthesis using Debevec's method
Merge luminance (Y) channel
Convert to YCbCr and extract the chroma (CbCr) channels
Combine luminance with chroma channels and convert to RGB

Fig. 2. Flowchart for the proposed non-learning based model

To better compare the results with those from the deep learning based method, we also use three LDR captures as the input data, denoted as $I_{\text{LDR},i}, i \in \{0, 1, 2\}$. The exposure time t_{exp} of each capture is also known. An example of the input data is shown in Figure 3.

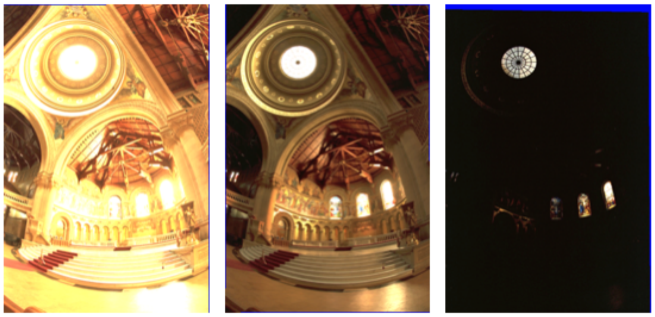

Fig. 3. Multi-exposure LDR inputs for the Debevec's and the proposed method

4.1.1 Luminance Channel Fusion

After converting the original input to the YCbCr color space, we isolate the luminance channels of the three images, $I^{\text{Y}}_{\text{LDR},i}, i \in \{0, 1, 2\}$. The merged luminance channel for the HDR image is calculated as $I^{\text{Y}}_{\text{HDR}} = \sum_{k=0}^{2} w_k I^{\text{Y}}_{\text{LDR},k}$ and the weight is chosen to have the symmetric triangular distribution, $w = \{1/4, 1/2, 1/4\}$. An example of fused luminance channel is shown in Figure 4.

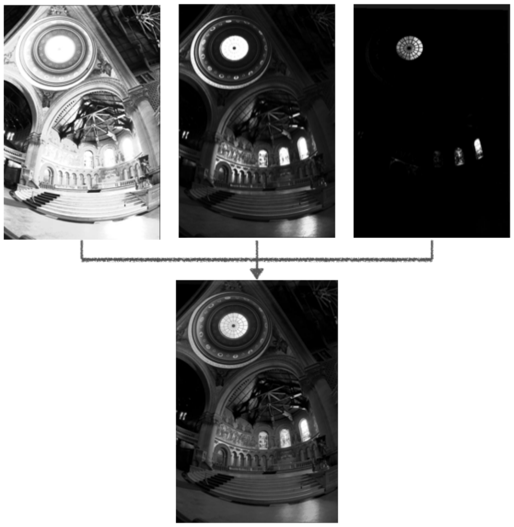

Fig. 4. Example of Luminance Channel Fusion

4.1.2 Chrominance Channels Fusion

To select the best HDR image color for each pixel, we adop the Debevec's method [5] to first merge the LDR images in the RGB color space. First, the LDR exposures are linearized by applying the inverse gamma function: $I_{\text{lin},i} = (I_{\text{LDR},i})^{\gamma}$, where $\gamma = 2.2$. At each pixel coordinate $[m, n]$, the weight is calculated as

$$w_{mn} = \exp\left(-4\frac{(I_{\text{lin},mn} - 0.5)^2}{0.5^2}\right) \tag{1}$$

and the HDR pixel value is computed as

$$I_{\text{HDR}} = \exp\left(\frac{\sum_k w_k(\log(I_{\text{lin},k}) - \log(t_{\text{exp},k}))}{\sum_k w_k}\right). \tag{2}$$

Next, the color channels, Cb and Cr, of the fused image are extracted after RGB-to-YCbCr conversion. The example chrominance channels are shown in grayscale in Figure 5.

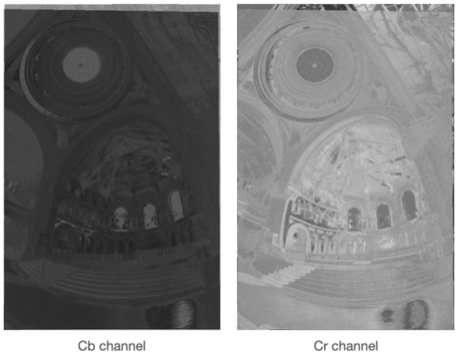

Fig. 5. Example of Chrominance Channels Extraction

4.1.3 HDR Image Reconstruction

Lastly, the merged luminance channel from the left branch in Figure 2 and the extracted color channels from the right branch are combined to form a YCbCr image. The final result is then converted back to RGB, as presented in Figure 6.

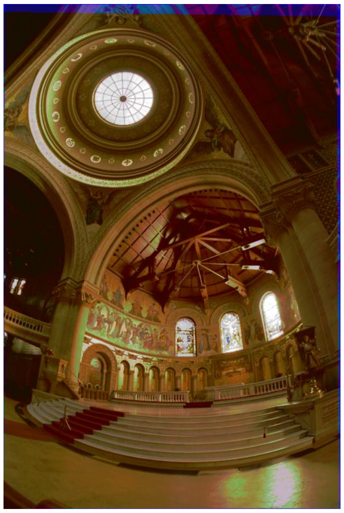

Fig. 6. Example of Final Reconstructed HDR Image

4.2 Deep Learning Based Method

For the learning based method, we choose U-Net [4] as our architecture for its encode-decoder structure, which is a great candidate for reconstruction task. The encoder consists of several 2D convolution and max pooling blocks to condense the input images to representation space. The decoder symmetrically expands the representation with 2D upsampling and convolution blocks, while appending the feature map of each encoder step to the corresponding decoder step, to reconstruct the output image.

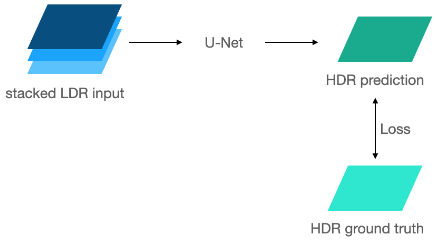

Fig. 7. Flowchart of Learning-based Model.

The pipeline of our learning method is shown in Figure 7. Concretely, our input to U-Net is of shape *batch size*x9x256x256. We have three LDR images for each scene, which have negative, zero, and positive exposures respectively. The LDR images are stacked vertically in that order to create a 9-channel block to feed into U-Net. The 2D convolution layer in our U-Net has kernel size of 3x3, stride of 1, padding of 1, and no bias term, and is followed by batch normalization and ReLU activation. The 2D upconvolution procedure uses bilateral upsampling. The convolution parameters are chosen so that the feature maps from the encoder has shapes matching those of the corresponding decoder steps. The output of our U-Net has shape *batch size*x3x256x256, where 3 stands for the RGB channels as the U-Net predicts the radiance map for each of the color channels. Loss is calculated between the U-Net prediction and the ground truth HDR radiance maps. More specific U-Net parameters are detailed in Figure 8.

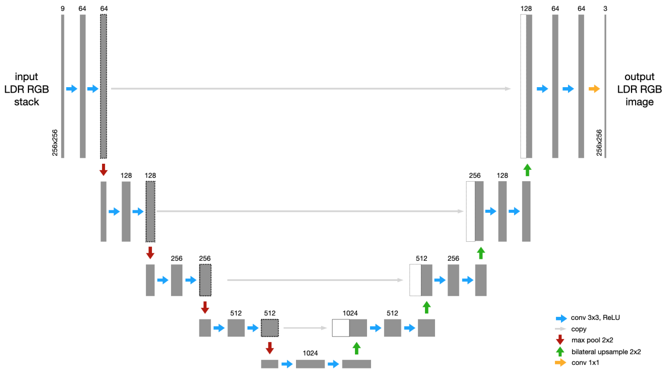

Fig. 8. U-Net architecture and the specific parameters applied in our project.

The U-Net architecture, dataloader, and training, validation, and test pipelines are written in PyTorch. For the training process, we use batch size of 16, learning rate of 5E-5, momentum of 0.9 for the SGD optimizer, and 150 training epochs. Other parameters for the optimizer and neural network layers are set to be PyTorch's default values.

We study the learning-based method with U-Net from three aspects, combination of optimizer and loss function, image quality regularization, and loss function variation. We aim to see how changes in each of those aspects impact the learning result. For model evaluation, we look at both qualitative results and quantitative metrics, including PSNR value to evaluate the signal quality in the reconstruction and SSIM value to perceptually measure structural similarity and image reconstruction quality.

4.2.1 Optimizer and Loss Function

We experiment with two optimizers, SGD [19] and Adam [20], and two loss functions, L2 norm and L1 norm, to see which combination works the best for our LDR-HDR conversion task. Different from SGD, Adam utilizes both decaying average of gradient and squared gradient. It is claimed that Adam converges faster to global minimum while SGD generalizes better [21]. Generally speaking, L2 norm promotes spreading out weights to smaller values, while L1 norm tends to zero out insignificant weight components and promote sparsity.

4.2.2 Image Quality Regularization

Besides directly reducing the difference between model prediction and ground truth, adding constraints on inherent qualities of the predicted image themselves may help guide the model to learn specific traits for a particular type of image. One trait we want in the model output is low amount of blurring and noise. Therefore, we add TV regularization term multiplied by different weights to obverse whether it can promote sparse gradients in the outcomes. For the sake of simplicity, we use anisotropic TV which adds the L1 norm of both horizontal and vertical gradients.

$$\text{TV}_{\text{anisotropic}}(I) = ||D_x I||_1 + ||D_y I||_1 \tag{3}$$

where I is the vectorized image, and $D_x I$ and $D_y I$ are gradients in the horizontal and vertical directions, respectively. The regularizer term then can be expressed as $\lambda\text{TV}_{\text{anisotropic}}(I)$, where $\lambda \in \mathcal{R}$ adjusts the contribution of the TV regularization term.

4.2.3 Loss Function Variation

Setting the suitable loss function helps the model to not only reduces the general pixel-wise difference between model prediction and ground truth, but also reduces the difference in particular expected image features. Cosine similarity is used in the loss term in [22] to promote color correctness in the RGB channels. It measures how close the directions of pixel-wise color vectors in RGB space are between the prediction and ground truth. The objective function of the i-th image of the dataset, including the loss contribution resulting from cosine similarity, is

$$L_i = L_{1 \text{ or } 2} + \lambda \left(1 - \frac{1}{K} \sum_{j=1}^{K} \frac{\hat{I_i^j} \cdot I_i^j}{||\hat{I_i^j}||_2 \cdot ||I_i^j||_2} \right) \tag{4}$$

where $\hat{I}$ is the predicted HDR image and I is the ground truth. I_i^j denotes the j-th pixel vector of ground truth image I_i, and K denotes the total number of pixels of image I_i.

We add cosine similarity to L2 loss, which in section 5 we will report to have a higher average test PSNR value than L1, and weight the similarity term by different values to examine how it affects the model's learning.

5 Experimental Results

5.1 Non-Learning Based Method

Note that the fundamental difference between the proposed non-learning based method and the Debevec's method is that it involves some level of image processing in the YCbCr space. Therefore, by comparing the performance of proposed algorithm with Debevec's, we can get insights into how effective it is to separately process the isolated luminance and chrominance channels.

Due to the non-learning based nature of this proposed method, there is no ground truth HDR image for the set of input exposures we use in Figure 3. Therefore, the comparison of final results are performed in a qualitative fashion.

The results of merging the set of LDR exposures in Figure 3 using the two non-learning methods are shown in Figure 9. The HDR image reconstructed using the proposed result (left) shows more details in areas where details are only preserved in over- or under-exposed LDR captures, such as the frame of the round window on the top-left corner. Overall, it recovers most of the details of the original scene. The Debevec's method, on the other hand, is not as effective in recovering such details. It does, however, show a brighter and more vibrant scene in general.

It is important to acknowledge the qualitative assessment and judgement of an HDR image can be highly subjective. However, the proposed approach is shown to be effective in a sense that more details can be observed after the image reconstruction.

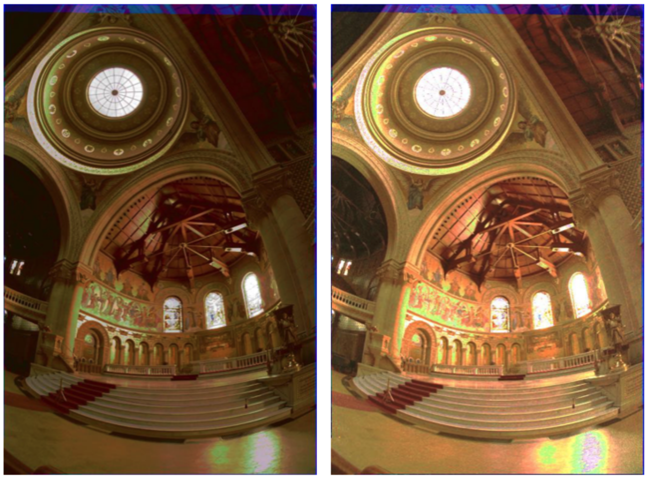

Fig. 9. HDR image synthesized using the proposed non-learning (left) and the Debevec's (right) algorithms.

5.2 Deep Learning Based Method

5.2.1 Optimizer and Loss

From the quantitative results summarized in Table 1, we see that Adam optimizer performs significantly better regardless of which loss function is used. There is a 6-8 dB lift in PSNR value and 0.4-0.6 increase in SSIM value when the optimizer is switched from SGD to Adam. When the optimizer is fixed to Adam, using L2 loss results in a slightly higher average test PSNR score, while using L1 loss achieves a good PSNR score and also higher SSIM value.

TABLE 1
Average test PSNR (dB) and SSIM for different combinations of optimizer and loss function.

	PSNR (dB)	SSIM
SGD+L2	14.91	0.2865
SGD+L1	16.90	0.5242
Adam+L2	**22.75**	0.8260
Adam+L1	22.69	0.9382

From Figure 10, we see that using SGD as the optimizer generally results in noisier images and more color artifacts. A potential explanation for this observation is that SGD may have led the training process to stop at a local minimum, which is avoided when Adam optimizer is used and achieves higher PSNR and SSIM scores. When SGD is set as the optimizer, using L1 loss in training results in the images with clearer structures and fewer color artifacts. This may likely be due to the utility of L1 loss promoting sparse weights instead of diffusing the weight values and leading to more spread-out pixel-wise differences. For the two examples given in Figure 10, the combination of Adam+L1 results in better HDR images both quantitatively and qualitatively. There are more visible color shifts in the Adam+L2 results than in the Adam+L1 results. However, this observation may not generalize to all test images or scenes in other datasets.

As Table 1 suggests that Adam+L2 has the highest average test PSNR score, we use Adam+L2 as the basis for the following two sets of experiments discussed in the next sections.

5.2.2 Image Quality Regularization

The motivation for using the TV regularizer is to model natural images by piecewise constant approximations and promote sharp edges between areas of relatively constant intensity. As shown in Table 2, adding the TV regularizer does not help with training, resulting a much less than ideal PSNR.

TABLE 2
Average test PSNR (dB) and SSIM for TV regularizer with different λ

TV weight	PSNR (dB)	SSIM
λ =0.0001	10.57	0.3729
λ =0.001	10.87	0.3585
λ =0.01	10.82	0.3759
λ =1	9.323	0.07885

One possible reason for this is that the calculated anisotropic TV term in equation (4) is proven to be much larger (by three or four orders of magnitude) than the L1 or L2 loss, which potentially can cause the main learning objective of the neural network to shift to image denoising, instead of recovering the HDR image. This can be partially reflected by the relatively consistent trend observed in Table 2 - the SSIM increases as we decrease the weight of the TV regularization term.

Since TV prior is a natural image prior, it is based on the assumption that our training dataset predominantly consists

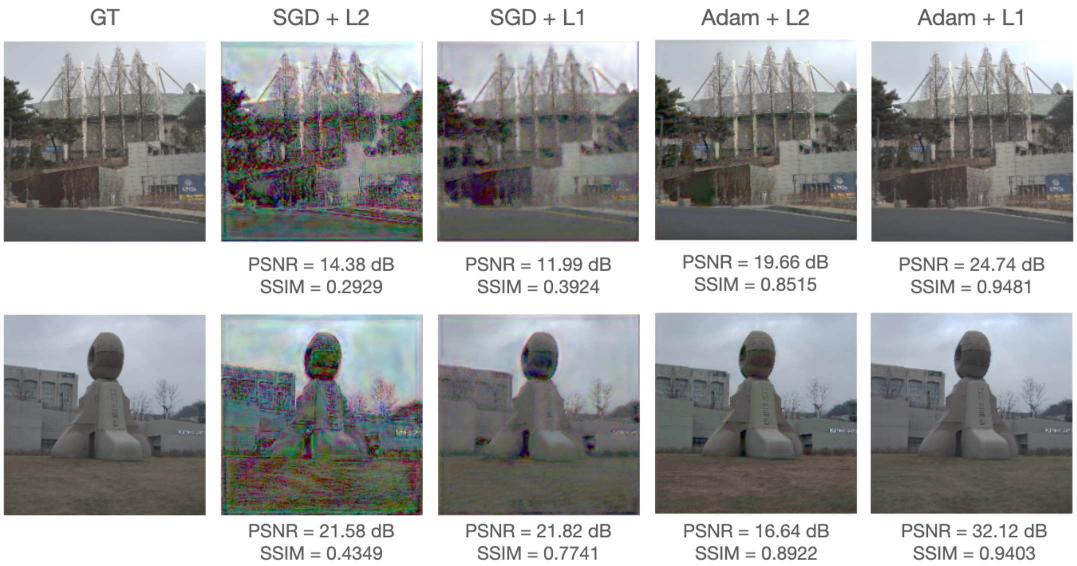

Fig. 10. HDR images generated with models that are trained with different combinations of optimizers and loss functions.

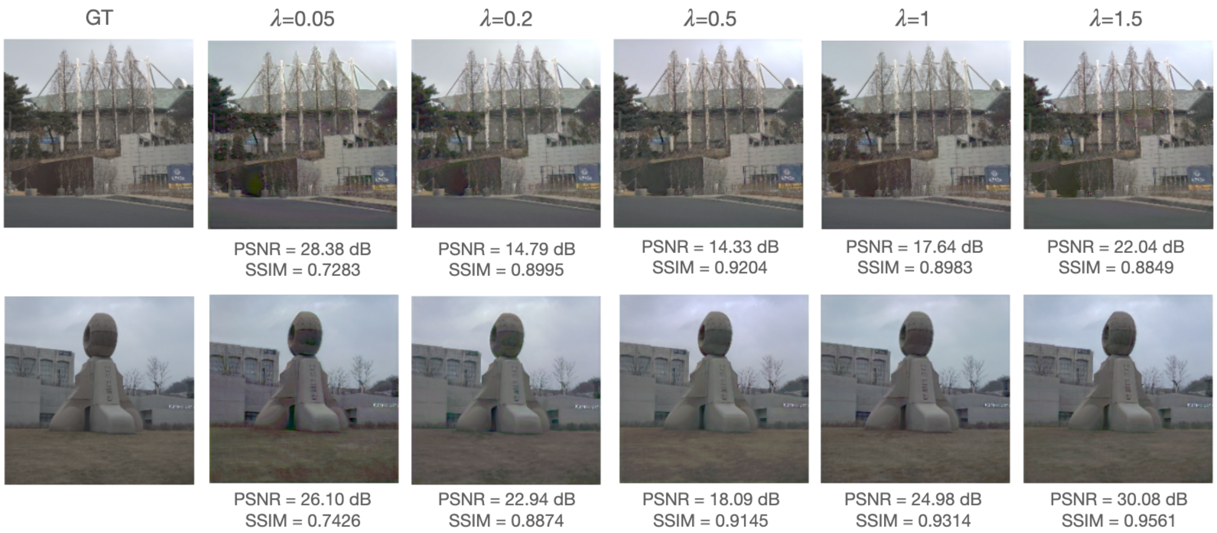

Fig. 11. HDR images generated with models that are trained with differently weighted L2+cosine similarity loss.

of natural images. However, this is not the case. The LDR-HDR pair dataset used in this project has a variety of images, including indoor scenes, architecture, and landscapes. This may also explain why TV regularizer does not work well for this model.

5.3 Loss Function Variation

Comparing the results in Table 3 and the Adam+L2 result in Table 1, we find that adding cosine similarity term in the loss function continues to improve the average test PSNR and SSIM scores when the cosine similarity term is weighted by certain λ values. The highest PSNR score is achieved when λ is 1. The second highest PSNR score is achived when λ is 0.5, which also leads to the highest SSIM score. We do not observe a monotonic relationship between λ values and the test results. But it may be a reasonable conclusion to say that λ values in the middle region of around 0.5-1 will add benefit to simply using L2 loss.

TABLE 3
Average test PSNR (dB) and SSIM for cosine similarity with different λ

Cosine Sim weight	PSNR (dB)	SSIM
λ =0.05	22.56	0.8288
λ =0.2	22.40	0.8875
λ =0.5	23.42	0.8998
λ =1	**23.65**	0.8546
λ =1.5	22.08	0.8867

Looking at the qualitative results shown in Figure 11, we see that the trend demonstrated in the quantitative results in Table 1 do not completely translate to the results of two randomly sampled individual test images. In general, we do observe that adding cosine similarity term to L2 loss allows the model to output HDR images with sound structure and correct colors that are close to the ground truth. But in the specific examples given in Figure 11, neither λ=1 or λ=0.5 results in the highest PSNR score, but rather λ=0.05 does.

With the findings in quantitative and qualitative results combined, it is hard to conclude which λ value is the best. The weighting of the cosine similarity term may be a hyperparameter to be tuned for specific tasks or specific images.

5.4 Non-Learning Based vs. Learning Based Methods

To get a better idea of how well each of the methods performs on the multi-exposure HDR synthesis task, we use the three LDR exposures in Figure 3 and generate the output HDR images (see Figure 12 using all three methods). Here, deep learning based method refers to the U-Net trained with cosine similarity term weighed by λ of 1 in the loss function, which is shown in Table 3 to have the highest average test PSNR. The selected images focus on portions of the image where the dynamic range is high, so that the effect of each method can be more clearly reflected.

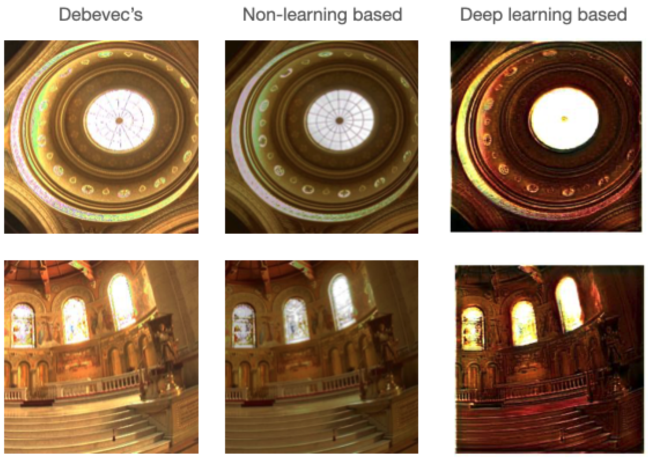

Fig. 12. HDR image synthesized using Debevec's (left), proposed non-learning based (middle) and deep learning based (right) algorithms.

Overall, the proposed non-learning based method shows the highest dynamic range, as the frames of the window (first row) and the window paintings (second row) can be clearly seen from the middle images. The level of details observed in the images produced by the Debevec's method is slightly lower, but the tone of the images is brighter and less dull. In the third column, images predicted by our deep learning based method have richer but a little overly saturated colors, and still have portions of the images over-exposed, such as the windows. One potential explanation for our deep learning method's less ideal qualitative outcome can be that the test images here are out of the distribution of our training data, which are not as rich in colors, have cooler colors, have less extreme difference in dynamic range, and show fewer fine architectural details. Such comparison exposes one drawback of deep learning approach in the LDR-HDR conversion task, that is, the choice of training dataset can limit the ability of the model to generalize well to other datasets or test scenes.

6 Discussion and Future Work

Both of the proposed methods, non-learning and learning based, have their own strengths and weaknesses. By analyzing the results qualitatively, we see that the non-learning based method shows consistently good performance but its algorithm requires more information, the exposure time, in this case, to calculate a better weight function. The deep learning based algorithm, on the other hand, is more versatile in the sense that it does not need extra meta data for the images, but can produce less than ideal results if the test images are very different from the training data. Therefore, it is worth trying out other methods, such as data augmentation, to help create a more robust and domain agnostic learning based model for multi-exposure HDR synthesis task.

More work can also be put into investigating why TV regularizer does not work well for this task. It would also be interesting to further investigate whether other regularization terms on the image quality itself can help enhance HDR image result quality in deep learning based methods.

Acknowledgments

We would like to thank Cindy Nguyen for her suggestions on our project. We also would like to thank EE367 teaching staff for this great course and quarter.

References

[1] N. Paul and C. Chung, "Application of hdr algorithms to solve direct sunlight problems when autonomous vehicles using machine vision systems are driving into sun," *Computers in Industry*, vol. 98, pp. 192–196, 2018.

[2] M. D. Fairchild, "The hdr photographic survey," in *Color and imaging conference*, vol. 2007, no. 1. Society for Imaging Science and Technology, 2007, pp. 233–238.

[3] I. Merianos and N. Mitianoudis, "A hybrid multiple exposure image fusion approach for hdr image synthesis," in *2016 IEEE International Conference on Imaging Systems and Techniques (IST)*, 2016, pp. 222–226.

[4] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," *CoRR*, vol. abs/1505.04597, 2015. [Online]. Available: http://arxiv.org/abs/1505.04597

[5] P. E. Debevec and J. Malik, "Recovering high dynamic range radiance maps from photographs," in *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '97. USA: ACM Press/Addison-Wesley Publishing Co., 1997, p. 369–378. [Online]. Available: https://doi.org/10.1145/258734.258884

[6] T. Mertens, J. Kautz, and F. Van Reeth, "Exposure fusion," in *15th Pacific Conference on Computer Graphics and Applications (PG'07)*, 2007, pp. 382–390.

[7] N. Mitianoudis and T. Stathaki, "Optimal contrast correction for ica-based fusion of multimodal images," *IEEE Sensors Journal*, vol. 8, no. 12, pp. 2016–2026, 2008.

[8] C. Liu, "Exploring new representations and applications for motion analysis," 2009.

[9] A. Dosovitskiy, P. Fischer, E. Ilg, P. Häusser, C. Hazirbas, V. Golkov, P. van der Smagt, D. Cremers, and T. Brox, "Flownet: Learning optical flow with convolutional networks," *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 2758–2766, 2015.

[10] Q. Yan, D. Gong, Q. Shi, A. van den Hengel, C. Shen, I. D. Reid, and Y. Zhang, "Attention-guided network for ghost-free high dynamic range imaging," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1751–1760, 2019.

[11] K. R. Prabhakar, V. S. Srikar, and R. V. Babu, "Deepfuse: A deep unsupervised approach for exposure fusion with extreme exposure image pairs," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 4724–4732.

[12] H. Xu, J. Ma, and X.-P. Zhang, "Mef-gan: Multi-exposure image fusion via generative adversarial networks," *IEEE Transactions on Image Processing*, vol. 29, pp. 7203–7216, 2020.

[13] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," 1992.

[14] T. Chan, A. Marquina, and P. Mulet, "High-order total variation-based image restoration," *SIAM Journal on Scientific Computing*, vol. 22, no. 2, pp. 503–516, 2000.

[15] A. Chambolle and P.-L. Lions, "Image recovery via total variation minimization and related problems," *Numerische Mathematik*, vol. 76, no. 2, pp. 167–188, 1997.

[16] H. Jang, K. Bang, J. Jang, and D. Hwang, "Dynamic range expansion using cumulative histogram learning for high dynamic range image generation," *IEEE Access*, vol. 8, pp. 38 554–38 567, 2020.

[17] E. Pan and A. Vento, "Metahdr: Model-agnostic meta-learning for hdr image reconstruction," March 2021.

[18] P. E. Debevec and J. Malik, "Recovering high dynamic range radiance maps from photographs," in *ACM SIGGRAPH 2008 Classes*, ser. SIGGRAPH '08. New York, NY, USA: Association for Computing Machinery, 2008. [Online]. Available: https://doi.org/10.1145/1401132.1401174

[19] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, ser. ICML'13. JMLR.org, 2013, p. III–1139–III–1147.

[20] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2015.

[21] P. Zhou, J. Feng, C. Ma, C. Xiong, S. C. H. Hoi, and E. Weinan, "Towards theoretically understanding why sgd generalizes better than adam in deep learning," *ArXiv*, vol. abs/2010.05627, 2020.

[22] 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," *Computer Graphics Forum*, vol. 37, 2018.