Learning Exposure Correction Via Consistency Modeling

Ntumba Elie Nsampi
elientumba@mail.nwpu.edu.cn
Zhongyun Hu
zy_h@mail.nwpu.edu.cn
Qing Wang
qwang@nwpu.edu.cn

School of Computer Science
Northwestern Polytechnical University
Xi'an 710072, China.

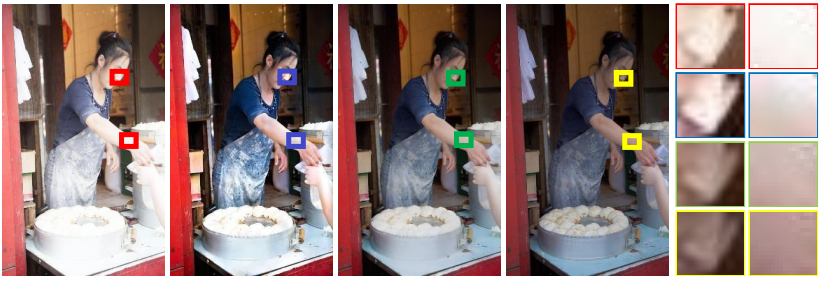
Abstract

Existing works on exposure correction have exclusively focused on either under-exposure or over-exposure. Recent work targeting both under-, and over-exposure achieved state of the art. However, it tends to produce images with inconsistent correction and sometimes color artifacts. In this paper, we propose a novel neural network architecture for exposure correction. The proposed network targets both under-, and over-exposure. We introduce a deep feature matching loss that enables the network to learn exposure-invariant representation in the feature space, which guarantees image exposure consistency. Moreover, we leverage a global attention mechanism to allow long-range interactions between distant pixels for exposure correction. This results in consistently corrected images, free of localized color distortions. Through extensive quantitative and qualitative experiments, we demonstrate that the proposed network outperforms the existing state-of-the-art. Code: https://github.com/elientumba2019/Exposure-Correction-BMVC-2021

1 Introduction

Capturing high-quality images that are neither too dark nor too bright requires a perfect combination of environment lighting and photographic device configuration. Capturing well-exposed images is achievable in studio-like environments and sometimes in outdoor and even indoor environments when lighting conditions are satisfactory. However this is seldom the case, and more often than not, images suffer from either over-exposure or under-exposure. Such is the case for images captured by casual photographers and hobbyists who do not always have professional photographic equipment. Exposure errors can occur due to many factors, including natural ones (Low light environment, bright scenes) and human-caused ones (error in exposure settings of the camera).

Post-capture Exposure correction enables users to enhance the visual quality of images after they have been captured. This is crucial as nowadays most people carry camera-equipped cell phones. Beyond aesthetics, exposure correction is also an important image processing and computer vision problem whose solutions can be applied to images before subjecting them to more high-level vision algorithms, and has been shown to improve their performance [1, 46].



(a) Input (b) MSEC (c) Ours (d) GT (e) Insets

Figure 1: Motivation for our method. Overexposed Input (a). MSEC [1] (b), tends to produce results with inconsistent correction. Our method (c) produces results with consistent correction that are closer to the ground truth (d) as can be observed from insets images (e).

Deep Learning based solutions have achieved state-of-the-art results on multiple computer vision benchmarks [29, 39]. These methods for the most part utilize convolutional neural networks (CNNs) [26] as their main architecture. The majority of existing works on exposure corrections have been limited to solving exclusively for either under- or over-exposure. Recent work by Afifi *et al*. [1], addresses both under, and over-exposure correction in a single framework. Their method [1] however, tends to produce results that suffer from a lack of correction consistency (CC), and sometimes color distortions. We define Correction Consistency as a method's ability to properly correct all pixels in an image, see Figure 1. The lack of CC in [1] occurs more frequently when correcting over-exposed images as opposed to under-exposed images. We hypothesize that the lack of CC in [1] is due to their framework treating distant pixels of similar properties separately without accounting for their similarities in color or illumination. Moreover, [1] does not explicitly address the problem of image exposure consistency (EC), which is to explicitly learn exposure-invariant deep feature representation in the network.

In this work, we tackle the above challenges and propose a novel neural network architecture for exposure correction. Specifically, we use an encoder to extract features from an input image, and a decoder to recover a well-exposed image. Motivated by the Retinex model [25] we propose a deep feature matching loss that is used to model image exposure consistency. Using the proposed feature matching loss in our pipeline, encourages the network to learn an exposure-invariant feature representation. We also leverage a Global Attention Block (GAB), which we introduce in our learning pipeline. The modeling of long-range interactions between distant pixels via the GAB, enables our network to produce consistently corrected images. To train our network, we leverage a large-scale dataset [1] which comes with various exposure levels and diverse scenes. In summary, we make the following contributions:

- We propose a novel network architecture for exposure correction. Based on the hypothesis that pixels with similar properties should be given equal importance, to allow consistently corrected images, we design our architecture taking into account the long-range interactions between distant pixels.

- We propose a deep feature matching loss on encoder-generated features that enables our network to learn an exposure-invariant feature representation and at the same time enforce image exposure consistency.

- We perform extensive experiments and demonstrate that the proposed network outperforms the state of the art both qualitatively and quantitatively.

2 Related work

Image enhancement. These techniques are designed to enhance the visual quality of images. Statistic based methods [14, 27, 36, 57, 58] work on the image histogram, which is manipulated to obtain a higher quality image. Histogram methods essentially alter an image's contrast. Lower contrast areas in an image are promoted to higher contrast and contrast areas in excess of the desired range are reduced to the desired contrast. Tone curve adjustment techniques estimate which curve best corresponds to a desirable visual quality. Whereas earlier solutions are based on processing a single image [52]. Recent learning solutions [16, 32, 33, 51] leverage large datasets that are used for training. We propose a network architecture for exposure error correction which is different from general image enhancement.

Under-exposure correction. These techniques are also known as Low light image enhancement methods. They aim to promote a well-exposed image from an underexposed image. Retinex based methods [25], assume that images can be formulated as a pixel-wise multiplication of two separate images, namely an illumination map and a reflectance image [12, 17, 38, 46]. The underexposure problem is therefore reformulated as the recovery of both illumination and reflectance images that correspond to a well-exposed image. Extreme low light enhancement methods [3, 4] process raw images and are designed to simulate the Image processing pipeline. Learning-based solutions for underexposure correction make use of large-scale datasets and optimize various frameworks. They can either be supervised [7, 18, 43, 50, 54, 55], or unsupervised [16, 21]. Unlike these works, our work aims at promoting a well-exposed image from an under-exposed or over-exposed image in a single framework.

Over-exposure correction. Over-exposure correction techniques promote an over-exposed image to a well-exposed one. Over-exposed images suffer from a loss of texture and color details which make the over-exposed correction problem extremely challenging, requiring details hallucination. For the most part solutions to over-exposure, correction are cast as HDR reconstruction [37] where the main goal is to both hallucinate details in clipped pixels and recover the scene radiance. On one hand, Multi-image image HDR techniques [11, 23, 47, 48, 49] leverage the abundance of data found in multi-exposure images by fusing them to obtain a correctly exposed HDR image or 8bits image [31]. On the other hand single Image HDR reconstruction techniques [10, 28, 30, 40] are under-constrained and more challenging than their multi-images counterpart. They solely rely on hallucinating details in missing regions. Contrary to these works, Our work does not aim at reconstructing scene radiance, nor hallucinate missing details. Instead, we propose a framework for exposure error correction treating both under- and over-exposure errors.

Transformers in computer vision. Transformers [42] can model long-range dependencies between elements in a sequence. This makes them able to capture global context. Transformers [42] have recently been successfully applied to vision problems as diverse as Image classification [8, 56], image generation [6, 34] , image segmentation [5], video segmentation [9, 44], video action recognition, Object detection [56]. Vision transformers solutions are applied to images by treating a single image as a sequence of words, which is achieved by either breaking an image into patches or by using a backbone to first down-sample the

image then reshape it into a sequence. In this work, we leverage a Global Attention Block (GAB) which we implement as a stacked transformer layers.

3 Method

Given an input image rendered with the incorrect exposure settings, our goal is to produce a well-exposed image from the input image. Our solution is inspired by the retinex formation model [25], defined as follows:

$$I = R * S \tag{1}$$

Where I denotes an image that is formed via a pixel-wise multiplication of R which denotes the reflectance image, and S which is the illumination map. Under the retinex formation model, exposure correction is the recovery of either the reflectance image R or the illumination map S. Unlike previous works [12, 17, 38, 43, 46] that directly estimate R or S , we model equation (1) implicitly in our learning pipeline. Specifically, we constrain the network learning by introducing a deep feature matching loss that is used to model image exposure consistency in the feature space. In other words, we constrain our network to learn an exposure-invariant feature representation such that, given images with the same content but with different exposure, the resulting feature representation is approximately the same. Our learning scheme is thus analogous to learning a retinex model in deep feature space, where the exposure-invariant features represent R, and the illumination S is implicitly recovered by the network during the decoding phase. Figure 2 illustrates our learning pipeline. Note that the proposed method requires multiple exposures (at least two) at training time, and a single image at inference time. In the following, we present in detail our solution and training.

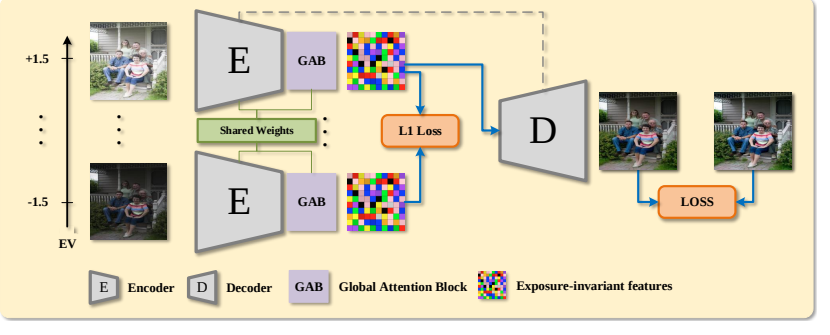


Figure 2: Training pipeline of our method. Given the images under different exposures, the shared-weights encoders are used to obtain an exposure-invariant feature representation, which is then fed to the decoder to recover a well-exposed image. A deep feature matching loss is used to enforce image exposure consistency across images.

3.1 Consistency modeling

Image Exposure Consistency. Exposure-consistency modeling guarantees that images of similar content but different exposure, should result in the same well-exposed image. Exposure consistency is achieved via the learning of exposure-invariant deep feature representation. Specifically, the input images are fed to the shared-weights encoders for feature

extraction. Our encoders are composed of a series of Residual Dense Blocks (RDB) followed by pooling layers. For images that have the same content but different exposure, the deep features produced by the encoder may vary due to change in illumination. Inspired by the retinex model [25], we introduce a deep feature matching loss that constrains our network to learn an exposure-invariant feature representation. For a given image I_k with exposure k, the encoder ϕ_{enc} encodes I_k into its feature representation e. Our feature loss is defined as:

$$L_{feat} = \frac{1}{M}\sum_{j=1}^{M} \|e - e_j\|_1 \tag{2}$$

where L_{feat} denotes the total loss, M denotes the number of images, and e_j denotes the feature representation of an image I_j, that has the same content as I_k but differs in exposure.

Image Correction Consistency. To produce consistently corrected images, our network needs to adjust global image properties (e.g., color distribution, average brightness) across multiple distant regions in the image. As shown in Figure 6, for a given image and query pixel our network needs to attend to distant pixels to complement each other. To model such long-range interactions between distant pixels, we employ a Global Attention Block (GAB), which we implement as a series of stacked transformer [42] layers. The input to the GAB is the deep features from the encoder. These features are collapsed along spatial dimensions, to produce a $N \times S$ feature map, where $N = \left(\frac{H}{16} \times \frac{W}{16}\right)$, and S is the dimension of the embedding. To retain positional information, a Fixed Positional Encoding [42] scheme is used, where position embeddings are added to each token as follows :

$$z_0 = \left[x_p^1, x_p^2; ...; x_p^N\right] + x_{pos} \tag{3}$$

In equation (3) $x_p^i \in R^S$ are tokens, and $x_{pos} \in R^{N \times S}$ denotes the position embeddings. The positional encoding used in this work is represented using sine and cosine functions of different frequencies [42]. We also experimented with learned positional encoding [13] and did not observe any significant improvement in terms of performance. In our ablation studies, we empirically demonstrate that the use of a positional encoding scheme results in increased performance as opposed to not using one. Each transformer layer in the GAB is composed of L layers of Multi-head Self-Attention, and Multi-Layer Perceptron blocks. The output of a transformer layer is computed as:

$$z_l^{'} = MSA\left(LN\left(z_{l-1}\right)\right) + z_{l-1} \tag{4}$$

$$z_l = MLP\left(LN\left(z_l^{'}\right)\right) + z_l^{'} \tag{5}$$

$$y = LN\left(z_l\right) \tag{6}$$

where MSA, MLP, and LN denote Multi-Head Attention, Multi-Layer Perceptron, and Layer Normalization blocks. For a given feature token x, each head h in an MSA attends to distant pixels via a self-attention mechanism defined as:

$$\alpha_{i,j}^{(h)} = softmax_j\left(\frac{\langle W_{h,q}^T x^i, W_{h,k} x^j \rangle}{\sqrt{k}}\right) \tag{7}$$

$$y_j = \sum_{h=1}^{H} W_{c,h}^T \sum_{j=1}^{n} \alpha_{i,j}^{(h)} W_{h,v}^T x^i \tag{8}$$

Where W_q, W_k, W_v and W_c are learned weight matrices. $\alpha_{i,j}^{(h)}$ denotes attention weight, $\sqrt{k}$ is a scaling factor and y_j is the self-attention output feature. A total of 6 transformer layers are used in the GAB, each having 8 attention heads and an internal representation of size 512.

To produce a well-exposed image, the updated feature maps from the GAB are first reshaped back to their original shape, and then subsequently fed to the decoder. The decoder is comprised of up-sampling blocks followed by convolutional layers. To recover the high-frequency details that were lost during the encoding phase, our network uses skip connections between the encoder and the decoder layers. The decoder progressively upsamples feature maps, until they reach the same resolution as that of the input image, after which a well-exposed image is produced. See the supplementary materials for details of the network architecture.

3.2 Loss function

To optimize our network's parameters, we train our model with a content loss and a perceptual loss [22]. While the content loss aims at minimizing the differences between the output image and the ground truth in image space, the perceptual loss aims at doing the same in feature space. The content loss is defined as:

$$L_{content}(I, I_i^*) = \frac{1}{n}\sum_i z_i$$
$$z_i = \begin{cases} 0.5(I - I_i^*)^2 \div \beta, & \text{if } |I_i - I_i^*| \leq \beta \\ |I_i - I_i^*| - 0.5 * \beta, & \text{otherwise} \end{cases} \tag{9}$$

Where I, and I^* denote the input image and the corresponding ground truth, $n = H \times W \times C$ denotes the total number of pixels and β is a scaling factor. We set β to a default value of 1. Equation (9) becomes the Hubert loss [19] function when β is omitted.
The Perceptual loss is defined as :

$$L_{perceptual}(I, I_i^*) = \frac{1}{n}\sum_i \sum_l \|\phi_l(I_i) - \phi_l(I_i^*)\|_1 \tag{10}$$

where $\phi_l(\cdot)$ denotes feature activation at the l_{th} layer of a pre-trained VGG-19 [41] network. The final loss is the combination of the losses in equations (2), (9) and (10) defined as :

$$L_{final} = L_{content} + \lambda_1 * L_{perceptual} + \lambda_2 * L_{feat} \tag{11}$$

where λ_1 and λ_2 are scalar weights to balance the overall loss. We use a value of 1 for λ_1 and 0.1 for λ_2.

4 Experiment and results

Dataset: We train our network on the exposure correction dataset of Afifi *et al*. [1]. It comes with realistically rendered over-exposed and under-exposed images, as well as their corresponding well-exposed ground truths. The dataset is rendered from the MIT-Adobe FiveK dataset [2] , corrected by 5 experts. Each image comes in five different exposures (EV: -1.5, -1.0, 0, +1.0, +1.5). A total of 17,675 images are available for training, 750 images for validation, and 5,905 images for testing.

Training details: Our network is implemented in PyTorch[35] on an NVIDIA GTX 1080Ti. We end-to-end optimize our network's parameters using Adam [24] with beta values 0.9 and 0.999, and a learning rate of 10^{-4}. Five exposures are used during training and a single image can be used at inference time. We start by training the network on 128x128 randomly selected patches. Once the training curve plateaus, which is when we observe no improvement on the overall loss for at least ten epochs, we increase the resolution of the patches (256x256, 384x384, 512x512) and continue training until we reach a final resolution of 768x768. We repeat the same process at each resolution until we reach the final resolution. In total the network is trained for 300 epochs using mini-batches of size 32.

4.1 Quantitative results

Our method is quantitatively evaluated on the test set of [1]. We use the Peak Signal to Noise Ratio (PSNR) and the Structural Similarity Index Metric (SSIM) [45]. We compare our method against previous learning, and non learning method [1, 7, 10, 12, 14, 16, 17, 18, 20, 43, 46, 53, 57]. We average results obtained on each expert test set, and report them in Table 1. Our methods outperform both methods that deal exclusively with either under, or over-exposure. It also outperforms the method of Afifi *et al*. [1], which deals with both under- and over-exposure correction. Detailed quantitative results on each expert set are reported in the supplementary material.

Methods	Underexposed		Overexposed		Under/Over exposure	
	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑
HE [14] *	16.576	0.679	16.582	0.683	16.580	0.682
CLAHE [57] *	16.350	0.621	14.808	0.589	15.425	0.602
WVM [12] *	18.615	0.735	13.503	0.657	15.548	0.688
LIME [16] *	14.643	0.671	10.487	0.582	12.150	0.618
HDR-CNN w/RHT [10]	13.589	0.420	13.842	0.486	13.741	0.460
HDR-CNN w/PS [10]	18.467	0.698	16.076	0.680	17.032	0.687
DPED(iPhone) [20]	19.858	0.685	13.883	0.591	16.274	0.629
DPED(BlackBerry) [20]	20.059	0.685	16.444	0.662	17.890	0.671
DPED(Sony) [20]	18.263	0.652	17.627	0.692	17.881	0.676
DPE(HDR) [1]	17.403	0.673	15.408	0.589	16.206	0.623
DPE(S-5K) [1]	18.495	0.677	15.453	0.640	16.670	0.655
DPE(U-5K) [1]	19.720	0.702	16.035	0.661	17.510	0.677
HQEC [53] *+	16.905	0.706	12.875	0.638	14.487	0.666
RetinexNet[46] +	12.494	0.619	11.059	0.600	11.633	0.607
Deep UPE[43] +	19.106	0.741	11.008	0.573	14.247	0.640
Zero-DCE[18] +	14.964	0.593	11.020	0.519	12.597	0.549
MSEC[1]	19.646	0.737	19.198	0.728	19.377	0.731
Ours	**21.126**	**0.839**	**21.881**	**0.866**	**21.579**	**0.855**

Table 1: Quantitative comparison on the test set of [1]. Methods are compared based on exposure. * denotes non learning-based methods. S and U stand for Supervised and Unsupervised. + denotes under-exposure correction methods. Our Method achieves higher PSNR and SSIM.

4.2 Qualitative results

We select three methods for both under-, and over-exposure correction [1, 20, 57] and compare them against our method. Figure 3 and Figure 4 show visual comparisons between the selected methods and ours. In Figure 3 we perform comparisons in terms of exposure consistency. The well-exposed images obtained from correcting two images of different exposure

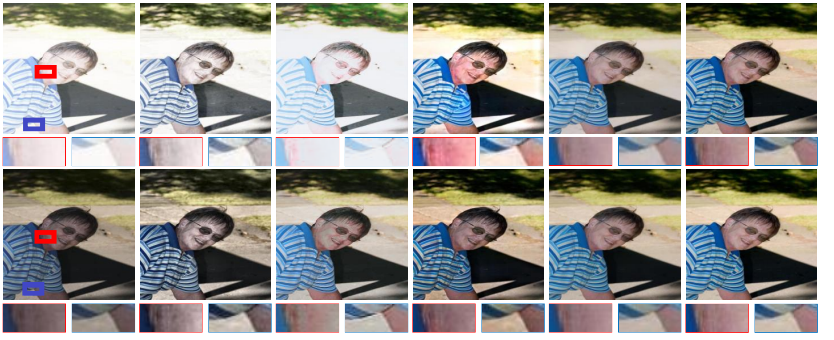


Figure 3: Qualitative comparison on the test set of [1] in terms of Exposure Consistency. Given two images with the same content but different exposures, our method tends to generate images with consistent exposure.

but the same content, should be as close as possible to each other and to the ground truth. Note that our method achieves significantly better results compared to other methods.

In Figure 4 We show results on under-, and over-exposure correction. CLAHE [14] fails to accurately recover colors for both under-, and over-exposure. DPED [20] produces images that are darker and contain artifacts for under-exposure correction. MSEC [1] fails to recover colors and textures details in some regions when correcting under-exposed images, while producing distorted colors when correcting over-exposed images.

Figure 5 shows results obtained from correcting images captured in the wild using a Nikon D90 camera. Note that the cost associated with collecting such data at scale is enormous. We therefore only include these two images as a proof of concept, and reserve the collection of a larger dataset for future work. Images captured with EV:0 are included for reference only, and should not be taken as ground truths. The results produced by our method are on par with those of Afifi *et al*. [1], whose method [1] tends to produce images with color artifacts as shown in insets. Both our method and [1] tend to produce images with deeper colors, which is an inherent characteristic of the dataset on which both models were trained. Our method produces results that are consistently corrected and free of artifacts. Additional qualitative results are presented in the supplementary materials.

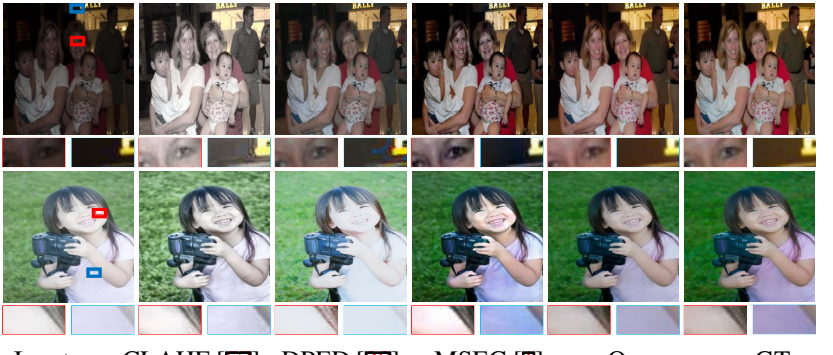


Figure 4: Qualitative comparisons on under- (Row1), and over-exposure (Row2) correction on the test set of [1].

Figure 5: Qualitative comparisons on real images captured in the wild using a Nikon D90 camera. *Top row*: EV:-1.5, *bottom row:*EV:+1.5.

Model	L_p	RDB	GAB	PE	L_{feat}	PSNR ↑	SSIM ↑
Baseline UNET	×	×	×	×	×	18.985	0.783
Model1	✓	×	×	×	×	18.724	0.800
Model2	✓	✓	×	×	×	20.897	0.821
Model3	✓	✓	✓	×	×	21.871	0.841
Model4	✓	✓	✓	✓	×	22.317	0.843
Ours	✓	✓	✓	✓	✓	**22.816**	**0.855**

Table 2: Ablation studies on the validation set of [1]. RDB denotes Redidual Dense Blocks, GAB denotes the Global Attention block. PE denotes Positional Encoding. L_p denotes the perceptual loss and L_{feat} denotes our deep feature matching loss.Our full model achieves higher PSNR and SSIM.

4.3 Ablation studies

Loss ablation. To evaluate the contribution of each loss term, we train different models using different loss combinations and compare them against our full model. Quantitative results on the validation set are reported in Table 2.

Model ablation. We build a baseline model (Baseline UNET) which is a simple encoder-decoder with skip connections, to illustrate the contribution of all components in our framework. In the baseline model, we replace all components with standard convolutional layers. To evaluate the importance of the Global Attention Block (GAB) in our network, we train a model (Model2) in which the GAB is removed. For a fair comparison against our full model, we replace the GAB with six convolutional layers with relu activation function. In Table 2 Our full model with GAB achieves superior performance. We also perform an ablation on the impact of Positional Encoding (PE), by training a model without any PE scheme (Model3). Our full model with a Fixed Positional Encoding [47] achieves higher PSNR and SSIM compared to its counterpart, as shown in Table 2. In Figure 6 we show the self-attention weights from the GAB. Given an over-exposed image and a query pixel, the GAB can attend to all pixels in the image and gives more importance to pixels that are similar to the query pixel in terms of color or illumination.

Deep features visualization. Exposure consistency modeling implies that given images sharing the same content but having different exposures, the features extracted from these images should be as close as possible to each other. In Figure 7 we show the deep features learned by our network with EC modeling and those of our variant (Model4) without EC modeling. These features are extracted from our network's bottleneck, and averaged along the channel dimension for visualization. The features learned by our model (with EC) are closer to each other, as can be observed from error maps. Note that error maps are calculated

by Error $(x,y) = |x-y|$.

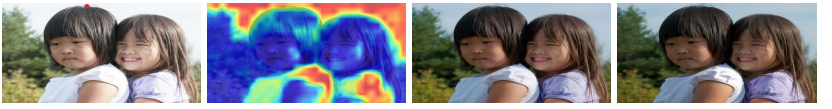


Figure 6: Visualization of self-attention. Given a query pixel, depicted by the red dot in (a), the GAB is able to attend to all pixels within the image and attend more to pixels that are similar to the query pixel in terms of color or illumination (b).

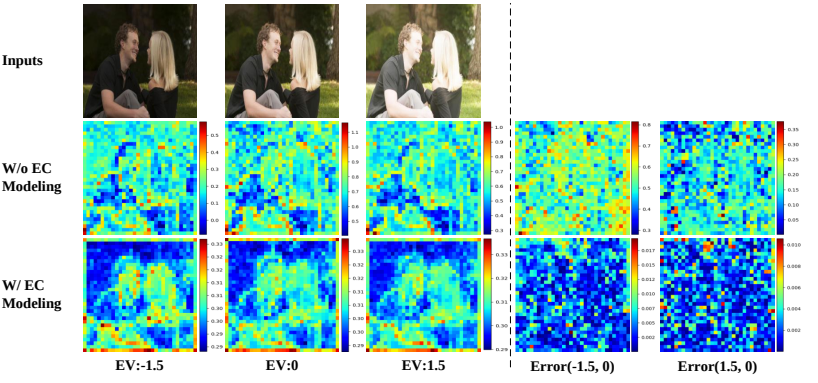


Figure 7: Visualization of deep features. *Top row*: input images sharing the same content but having different exposures. *middle row*: features learned without EC modeling (Model4). *bottom row*: features learned with EC modeling. The deep features learned by our model (w/ EC) are closer to each other, as shown in error maps (column 4, 5).

5 Conclusion

In this paper, we have presented a new network architecture for exposure correction, that addresses both under-, and over-exposure. Exposure consistency is modeled by constraining the network to lean an exposure-invariant feature representation. Moreover, we leverage a Global Attention Block (GAB) to model the long-range interaction between distant pixels. This design choice enables the proposed network to generate images that are consistently corrected and free of artifacts.

Limitations. We observe that our method fails when a given image is extremely under- or over-exposed with saturated pixels, and/or missing semantic information (see supplementary materials for visual examples). This limitation is primarily due to our method's inability to hallucinate non-existing content. A possible extension could be the inclusion of adversarial learning [15] within our approach, as it has proven effective in hallucinating plausible contents in images.

Acknowledgements. This work was supported by NSFC under Grant 62031023.

References

[1] Mahmoud Afifi, Konstantinos G Derpanis, Björn Ommer, and Michael S Brown. Learning multi-scale photo exposure correction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021.

[2] Vladimir Bychkovsky, Sylvain Paris, Eric Chan, and Frédo Durand. Learning photographic global tonal adjustment with a database of input / output image pairs. In *The Twenty-Fourth IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[3] Chen Chen, Qifeng Chen, J. Xu, and V. Koltun. Learning to see in the dark. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3291–3300, 2018.

[4] Chen Chen, Qifeng Chen, M. Do, and V. Koltun. Seeing motion in the dark. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3184–3193, 2019.

[5] Jieneng Chen, Yongyi Lu, Qihang Yu, Xiangde Luo, E. Adeli, Yan Wang, Le Lu, A. Yuille, and Yuyin Zhou. Transunet: Transformers make strong encoders for medical image segmentation. *ArXiv*, abs/2102.04306, 2021.

[6] Mark Chen, Alec Radford, Jeff Wu, Heewoo Jun, Prafulla Dhariwal, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *ICML*, 2020.

[7] Y. Chen, Yu-Ching Wang, Man-Hsin Kao, and Yung-Yu Chuang. Deep photo enhancer: Unpaired learning for image enhancement from photographs with gans. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6306–6314, 2018.

[8] A. Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, M. Dehghani, Matthias Minderer, G. Heigold, S. Gelly, Jakob Uszkoreit, and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ArXiv*, abs/2010.11929, 2020.

[9] Brendan Duke, A. Ahmed, C. Wolf, P. Aarabi, and Graham W. Taylor. Sstvos: Sparse spatiotemporal transformers for video object segmentation. *ArXiv*, abs/2101.08833, 2021.

[10] G. Eilertsen, Joel Kronander, G. Denes, R. Mantiuk, and J. Unger. Hdr image reconstruction from a single exposure using deep cnns. *ACM Transactions on Graphics (TOG)*, 36:1 – 15, 2017.

[11] Y. Endo, Y. Kanamori, and J. Mitani. Deep reverse tone mapping. *ACM Transactions on Graphics (TOG)*, 36:1 – 10, 2017.

[12] Xueyang Fu, Delu Zeng, Y. Huang, X. Zhang, and Xinghao Ding. A weighted variational model for simultaneous reflectance and illumination estimation. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2782–2790, 2016.

[13] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann Dauphin. Convolutional sequence to sequence learning. In *ICML*, 2017.

[14] R. González and R. Woods. Digital image processing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-3:242–243, 1981.

[15] I. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014.

[16] C. Guo, Chongyi Li, J. Guo, Chen Change Loy, J. Hou, S. Kwong, and Runmin Cong. Zero-reference deep curve estimation for low-light image enhancement. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1777–1786, 2020.

[17] Xiaojie Guo. Lime: A method for low-light image enhancement. *Proceedings of the 24th ACM international conference on Multimedia*, 2016.

[18] Xiaojie Guo, Y. Li, and Haibin Ling. Lime: Low-light image enhancement via illumination map estimation. *IEEE Transactions on Image Processing*, 26:982–993, 2017.

[19] P. Huber. Robust estimation of a location parameter. *Annals of Mathematical Statistics*, 35:492–518, 1964.

[20] A. Ignatov, Nikolay Kobyshev, Kenneth Vanhoey, R. Timofte, and L. Gool. Dslr-quality photos on mobile devices with deep convolutional networks. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3297–3305, 2017.

[21] Yifan Jiang, Xinyu Gong, Ding Liu, Y. Cheng, Chen Fang, X. Shen, Jianchao Yang, Pan Zhou, and Zhangyang Wang. Enlightengan: Deep light enhancement without paired supervision. *IEEE Transactions on Image Processing*, 30:2340–2349, 2021.

[22] J. Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. *ArXiv*, abs/1603.08155, 2016.

[23] N. Kalantari and R. Ramamoorthi. Deep high dynamic range imaging of dynamic scenes. *ACM Transactions on Graphics (TOG)*, 36:1 – 12, 2017.

[24] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2015.

[25] E. Land. The retinex theory of color vision. *Scientific American*, 237 6:108–28, 1977.

[26] Y. LeCun, P. Haffner, L. Bottou, and Yoshua Bengio. Object recognition with gradient-based learning. In *Shape, Contour and Grouping in Computer Vision*, 1999.

[27] Chulwoo Lee, C. Lee, and C. Kim. Contrast enhancement based on layered difference representation of 2d histograms. *IEEE Transactions on Image Processing*, 22:5372–5384, 2013.

[28] Siyeong Lee, Gwon Hwan An, and Suk-Ju Kang. Deep recursive hdri: Inverse tone mapping using generative adversarial networks. In *ECCV*, 2018.

[29] Tsung-Yi Lin, M. Maire, Serge J. Belongie, James Hays, P. Perona, D. Ramanan, Piotr Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.

[30] Y. Liu, Wei-Sheng Lai, Yu-Sheng Chen, Yi-Lung Kao, Ming-Hsuan Yang, Yung-Yu Chuang, and J. Huang. Single-image hdr reconstruction by learning to reverse the camera pipeline. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1648–1657, 2020.

[31] T. Mertens, J. Kautz, and F. V. Reeth. Exposure fusion: A simple and practical alternative to high dynamic range photography. *Computer Graphics Forum*, 28, 2009.

[32] S. Moran, Pierre Marza, Steven G. McDonagh, Sarah Parisot, and G. Slabaugh. Deeplpf: Deep local parametric filters for image enhancement. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12823–12832, 2020.

[33] Jongchan Park, Joon-Young Lee, Donggeun Yoo, and In-So Kweon. Distort-and-recover: Color enhancement using deep reinforcement learning. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5928–5936, 2018.

[34] Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam M. Shazeer, Alexander Ku, and Dustin Tran. Image transformer. *ArXiv*, abs/1802.05751, 2018.

[35] Adam Paszke, S. Gross, Francisco Massa, A. Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Z. Lin, N. Gimelshein, L. Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019.

[36] S. Pizer, E. P. Amburn, J. D. Austin, R. Cromartie, A. Geselowitz, T. Greer, B. H. Romeny, and J. B. Zimmerman. Adaptive histogram equalization and its variations. *Graphical Models graphical Models and Image Processing computer Vision, Graphics, and Image Processing*, 39:355–368, 1987.

[37] E. Reinhard, G. Ward, S. Pattanaik, P. Debevec, W. Heidrich, and K. Myszkowski. High dynamic range imaging: Acquisition, display, and image-based lighting. 2010.

[38] Xutong Ren, W. Yang, W. Cheng, and Jiaying Liu. Lr3m: Robust low-light enhancement via low-rank regularized retinex model. *IEEE Transactions on Image Processing*, 29:5862–5876, 2020.

[39] Olga Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Zhiheng Huang, A. Karpathy, A. Khosla, Michael S. Bernstein, A. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115: 211–252, 2015.

[40] Marcel Santana Santos, Ing Ren Tsang, and N. Kalantari. Single image hdr reconstruction using a cnn with masked features and perceptual loss. *ACM Transactions on Graphics (TOG)*, 39:80:1 – 80:10, 2020.

[41] K. Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2015.

[42] Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *ArXiv*, abs/1706.03762, 2017.

[43] R. Wang, Q. Zhang, Chi-Wing Fu, Xiaoyong Shen, W. Zheng, and J. Jia. Underexposed photo enhancement using deep illumination estimation. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6842–6850, 2019.

[44] Yuqing Wang, Zhaoliang Xu, Xinlong Wang, Chunhua Shen, B. Cheng, H. Shen, and Huaxia Xia. End-to-end video instance segmentation with transformers. *ArXiv*, abs/2011.14503, 2020.

[45] Zhou Wang, A. Bovik, H. Sheikh, and Eero P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13:600–612, 2004.

[46] Chen Wei, W. Wang, W. Yang, and Jiaying Liu. Deep retinex decomposition for low-light enhancement. In *BMVC*, 2018.

[47] Shangzhe Wu, J. Xu, Yu-Wing Tai, and C. Tang. Deep high dynamic range imaging with large foreground motions. In *ECCV*, 2018.

[48] Qingsen Yan, Dong Gong, Qinfeng Shi, A. V. Hengel, Chunhua Shen, I. 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)*, pages 1751–1760, 2019.

[49] Qingsen Yan, Lei Zhang, Yu Liu, Yu Zhu, Jinqiu Sun, Qinfeng Shi, and Yanning Zhang. Deep hdr imaging via a non-local network. *IEEE Transactions on Image Processing*, 29:4308–4322, 2020. doi: 10.1109/TIP.2020.2971346.

[50] W. Yang, Shiqi Wang, Y. Fang, Yue Wang, and Jiaying Liu. From fidelity to perceptual quality: A semi-supervised approach for low-light image enhancement. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3060–3069, 2020.

[51] Runsheng Yu, Wenyu Liu, Yasen Zhang, Zhi Qu, D. Zhao, and Bo Zhang. Deepexposure: Learning to expose photos with asynchronously reinforced adversarial learning. In *NeurIPS*, 2018.

[52] L. Yuan and Jian Sun. Automatic exposure correction of consumer photographs. In *ECCV*, 2012.

[53] Qing Zhang, Ganzhao Yuan, Chunxia Xiao, L. Zhu, and W. Zheng. High-quality exposure correction of underexposed photos. *Proceedings of the 26th ACM international conference on Multimedia*, 2018.

[54] Yonghua Zhang, Jiawan Zhang, and Xiaojie Guo. Kindling the darkness: A practical low-light image enhancer. *Proceedings of the 27th ACM International Conference on Multimedia*, 2019.

[55] Minfeng Zhu, P. Pan, W. Chen, and Y. Yang. Eemefn: Low-light image enhancement via edge-enhanced multi-exposure fusion network. In *AAAI*, 2020.

[56] X. Zhu, Weijie Su, Lewei Lu, Bin Li, X. Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *ArXiv*, abs/2010.04159, 2020.

[57] K. Zuiderveld. Contrast limited adaptive histogram equalization. In *Graphics Gems*, 1994.

[58] T. Çelik and T. Tjahjadi. Contextual and variational contrast enhancement. *IEEE Transactions on Image Processing*, 20:3431–3441, 2011.