Single LDR image to HDR translation

Alperen Demirci: 2220765010

Bora Dere: 2220765021

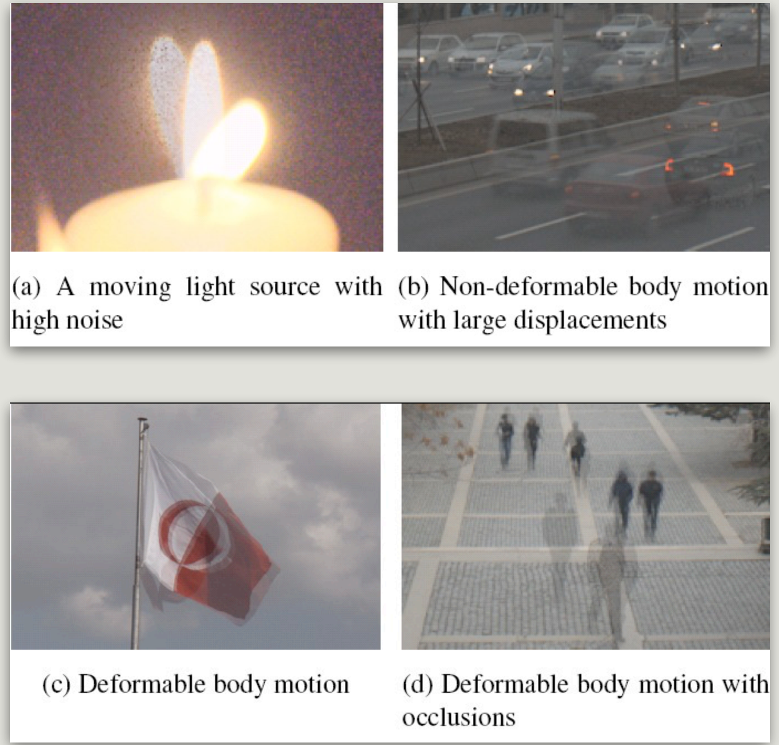
Image: Taken by Alperen Demirci. Located at Zirvekent Blocks ,Mamak ANKARA/Turkiye

Abstract

- High Dynamic Range (HDR) photography has been a prominent topic over the years.
- Traditionally, HDR images are generated by merging multiple Low Dynamic Range (LDR) images captured at different exposures.
- We design a specialized neural network architecture and train it on a carefully curated dataset to enhance local contrast and dynamic range.
- Our results demonstrate that it is possible to reconstruct high-resolution, visually convincing HDR images from a single shot.
- We find this problem highly interesting due to the rapid advances in both HDR imaging and deep learning methodologies. While traditional HDR imaging techniques have been well-studied, single-image HDR reconstruction remains relatively new and offers significant opportunities for innovation.

Introduction

- Producing high-quality HDR images remains a challenging task, involving a variety of computational approaches.
- Exposure bracketing is inherently fragile, especially in dynamic scenes where **motion artifacts, lighting changes, or misalignment** can degrade the final output.
- To address these limitations, HDR reconstruction from a **single LDR image** has emerged as an important research direction.
- Our approach employs an encoder-decoder architecture, where the encoder is a pretrained convolutional neural network (CNN) tasked with extracting meaningful image features into a compact latent space.



(a) A moving light source with high noise (b) Non-deformable body motion with large displacements

(c) Deformable body motion (d) Deformable body motion with occlusions

- Examples for challenging scenes for HDR imaging [1]

Related Work

- One notable example specifically targets the autonomous vehicle domain. In addition to traditional exposure bracketing, they introduce a tri-focus strategy aimed at enhancing disparity estimation for better HDR image construction[4].
- ExpandNet represents another noteworthy contribution to single-image HDR reconstruction. In this approach, the authors develop a multi-branch convolutional network consisting of three parallel CNN streams: a local branch, a dilation branch, and a global branch[5].
- Another line of research focuses on generating multiple synthetic exposures from a single LDR input to aid HDR reconstruction[6].
- Another important study ,which we heavily inspired from, focuses on HDR reconstruction from a single LDR image[3].

Dataset

- The dataset we use is a subset of the one introduced by Liu et al. [2] We utilize the same 14 GB training set and, for testing, the 0.52 GB HDR-Eye set. These datasets contain 9,786 and 46 samples, respectively.
- Data was collected across diverse scenes with different exposure settings for LDR images. It consists of the famous HDR-Eye dataset for evaluation.
- It was announced in **CVPR 2020** by researchers from **Google, MediaTek Inc., Virginia Tech, NTU and UC Merced** .
- Dataset includes pairs of 8-bit LDR images in .jpg format and corresponding 16-bit HDR images in .hdr format.

Data Preprocessing

- We have filtered out the high contrast HDR images because in some examples we have seen that this is due to a heavy light source in the image. This makes it more difficult to learn the HDR images so we have dropped images having larger value than 2^{12}(4096). This number is selected via inspecting the distribution of the maximum values of HDR images.
- Another issue was that some LDR/HDR image pairs were misaligned in orientation: the HDR counterparts were rotated or flipped relative to their LDR images. We have implemented a fast and effective, SSIM-based algorithm to **fix these unwanted misalignments**.
- After resolving alignment issues, we applied a **log1p** transformation to the HDR images. This compresses the dynamic range to a logarithmic scale, facilitating normalization and training stability. After that we have normalized these logHDR images using minmax norm, into [0,1] interval.
- For the LDR images, the preprocessing pipeline was straightforward. We first normalized them using the ImageNet statistics since the MobileNetV3 model was trained on that dataset, and then applied min-max normalization to bring their scale in line with the HDR images.

- HDR image having the largest contrast [0.0, 61180.0]

Final Training Pipeline

HDR:

$$HDR_{original} \leftarrow log(1 + HDR_{original})$$

$$HDR_{original} \leftarrow \frac{HDR_{original} - \min(HDR_{original}) + \epsilon}{\max(HDR_{original})}, \quad \epsilon = 10^{-6}$$

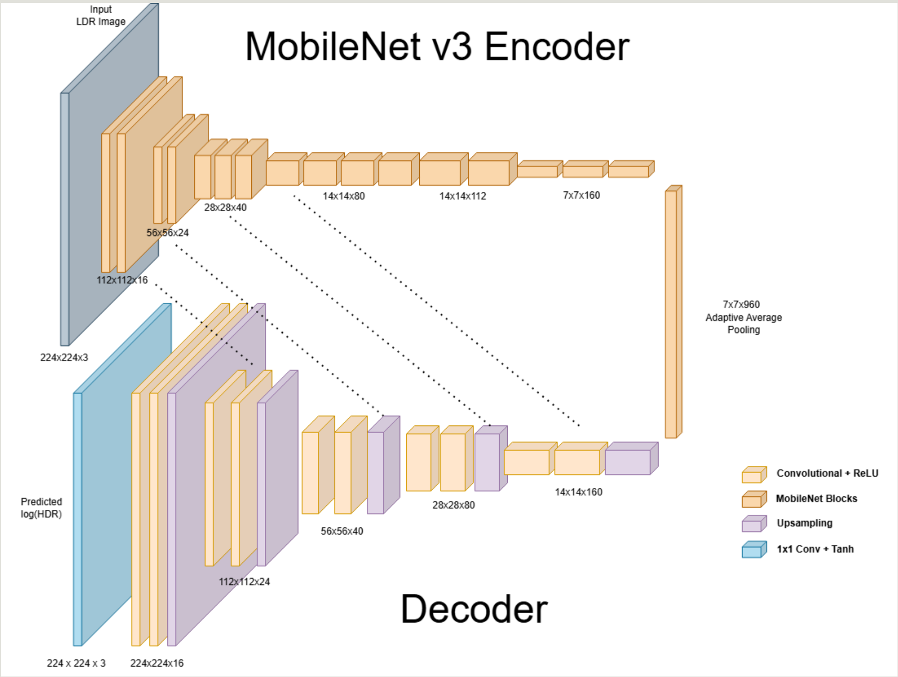
LDR:

$$LDR_{original} \leftarrow \frac{LDR_{original} - \min(LDR_{original}) + \epsilon}{\max(LDR_{original})}, \quad \epsilon = 10^{-6}$$

$$LDR_{original} \leftarrow ImgNet1KNorm(LDR_{original})$$

Modelling the Hybrid Autoencoder

- Our primary goal is **not to recover the absolute radiometric values of a true HDR image**, but rather to estimate the relative brightness and contrast variations between pixels, ultimately producing a **visually convincing HDR-like appearance**.
- Therefore, we focus on learning the distributional properties of HDR images and **enhancing the perceptual quality of LDR images by leveraging this knowledge**.
- **Autoencoder: MobileNetv3 Large** model pretrained on ImageNet1K as an encoder, UNet-like structure for both bottleneck and decoder.
- We only learn bottleneck and decoder, encoder is **frozen entirely**.
- We use **PixelShuffling** layers for upsampling and **Instance normalization** inside each block in decoder. Activation function is selected as **ReLU**.



Visualization of our proposed architecture.

Custom Loss

- **LPIPS Loss:** Computes the difference between two images in feature space of VGG19 network. Computes the difference between two images in the feature space of a pretrained VGG19 network. It performs well because it compares perceptual features instead of raw pixel values, penalizing the **loss of geometrically unwanted distortions** and better preserving **semantic content** and **structure**.
  - Encourages the model to generate outputs that *look* more similar to the target, even if pixel-wise differences exist.
  - Complements L1 Loss, which ensures **low-level accuracy**, while LPIPS ensures **high-level perceptual fidelity**.
- $\mathcal{L}_{\text{total}} = \lambda_{\text{L1}} \cdot \mathcal{L}_{\text{L1}} + \lambda_{\text{LPIPS}} \cdot \mathcal{L}_{\text{LPIPS}}$
- $\mathcal{L}_{\text{L1}} = \begin{cases} \frac{1}{N}\sum_{i=1}^{N} \alpha_i \cdot \left|\hat{y}_i - y_i\right| & \text{if alpha map is used} \\ \frac{1}{N}\sum_{i=1}^{N} \left|\hat{y}_i - y_i\right| & \text{otherwise} \end{cases}$
- $\mathcal{L}_{\text{LPIPS}} = \text{LPIPS}(\hat{y}, y)$

Evaluation Metrics

- We use two widely adopted metrics: Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM):

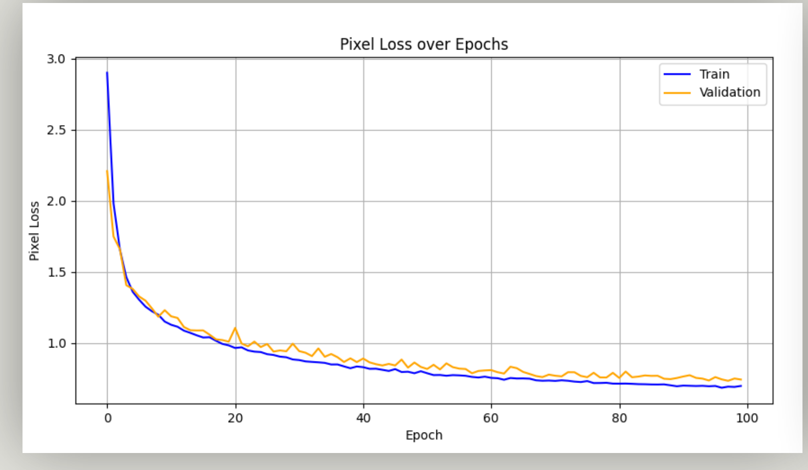
- **PSNR:** Measures the fidelity of the generated image with respect to the ground truth.
- **SSIM:** Evaluates the perceptual quality by considering luminance, contrast, and structure, providing a value between 0 and 1.
- **L1 Loss:** Intuitively good idea. Doesn’t allow for blurry regions like L2 Loss. Crucial for detail preservation.

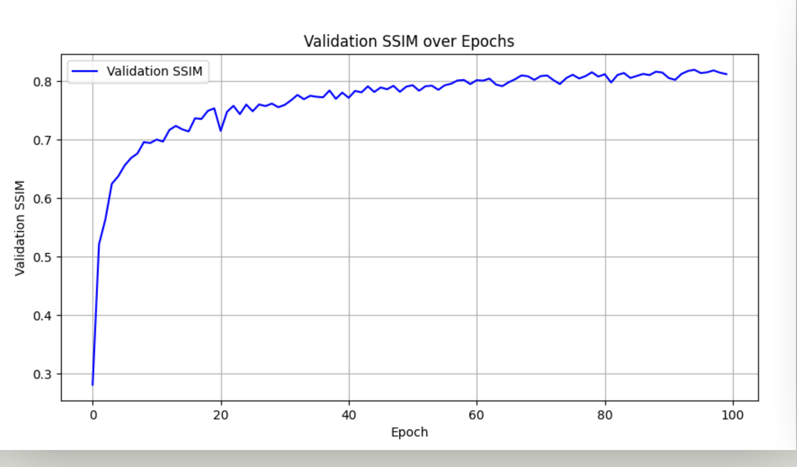
Implementation Details

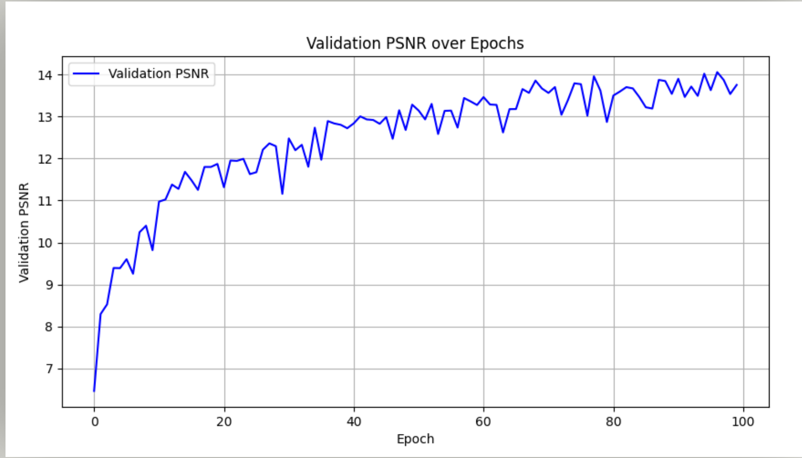
- Hyperparameter experiments include different learning rates (between 1×10^{-3} and 5×10^{-4}), different target domains (logHDR [0-12] or norm_logHDR[0-1]), different generator architectures (hybrid auto encoder and custom UNet), and different losses (different loss weightings). We followed a greedy, "keep-the-best" strategy to tune hyperparameters.
- We use the Adam optimizer with $beta_1 = 0.5$ and $beta_2 = 0.999$ for both networks. The initial learning rate is set to 9×10^{-4}. The batch size is set to 16, and the model is trained for 100 epochs. We have used the half of the dataset since even on using half of the dataset with these settings took **10 hours and approx. 22GB VRAM** on Apple M3 Max.
- We train only the decoder and bottleneck using a custom loss function which relies on both L1 Loss and LPIPS Loss.
- Also, we have conducted our experiments with an input size of 224x224 while our original images are 512x512. After finding the best hyper parameters for training (epochs, learning rate etc.), we have trained our model on 512x512 images.

Training Plots

- Final training configuration is as follows:
  - Learning rate: 9×10^{-4}, Batch size: 16, Number of epochs: 100, Upsampling method: PixelShuffle
  - L1 loss weight: 2
  - LPIPS weight: 3
  - Transformations: Random Vertical & Horizontal Flip







Results

- We have achieved a **PSNR of 14.0585, SSIM of 0.8157 and L1 Loss as 0.6976** on validation set.
- These metrics are calculated with raw model outputs and minmax normalized log HDR ground truth image.
- Note: We feed the inputs using [-1,1] range instead of [0,1]. The reason is that weights tend to update more stable on input range [-1,1] compared to [0,1].
  - Also since we use Tanh for final activation, our model logits are guaranteed to be in range [-1,1] so we can safely calculate our metrics in [-1,1] range instead of [0,1].
- In visualization step, we do the reverse of what we did to the images. This consists of converting the range to [0,1] from [-1,1], taking the exp to eliminate the log.

Visual Comparison

- Directly displaying a HDR image without any extensive processing is not the best way to feel the effect that HDR brings.
- That is why we visually compare our prediction and ground truth HDR image by after applying **Reinhard Tonemapping**. Formula is pretty simple yet effective: $L_{\text{mapped}} = \frac{L}{1+L}$.
- In some cases, our model generates images that appear dark when viewed in the raw `.hdr` format. Applying Reinhard Tonemapping helps compress the dynamic range into the [0,1] interval, making the content perceptually balanced and visually accessible on standard displays, **regardless of the original HDR intensity range.**

Some Examples from HDR-Eye dataset

Input LDR image

Predicted HDR with Reinhard Tonemapping

Ground truth HDR with Reinhard Tonemapping

Some Examples from HDR-Eye dataset

Input LDR image

Predicted HDR with Reinhard Tonemapping

Ground truth HDR with Reinhard Tonemapping

Some Examples from HDR-Eye dataset

Input LDR image

Predicted HDR with Reinhard Tonemapping

Ground truth HDR with Reinhard Tonemapping

Some Examples from HDR-Eye dataset

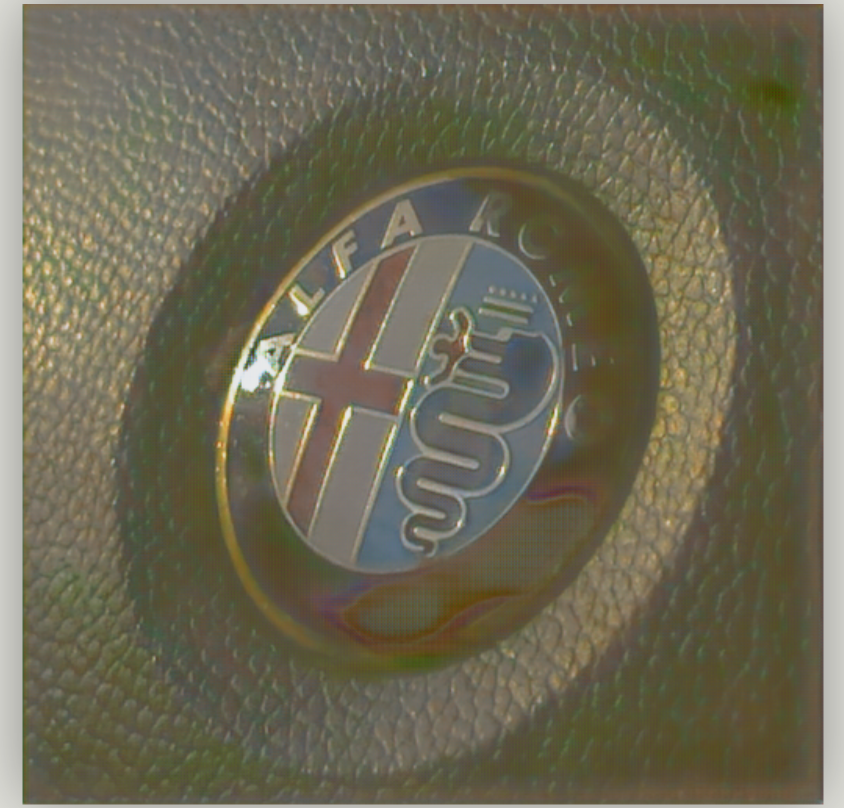
Input LDR image

Predicted HDR with Reinhard Tonemapping

Ground truth HDR with Reinhard Tonemapping

Some Examples from our custom images

Input LDR image



Predicted HDR with Reinhard Tonemapping

Some Examples from our custom images

Input LDR image

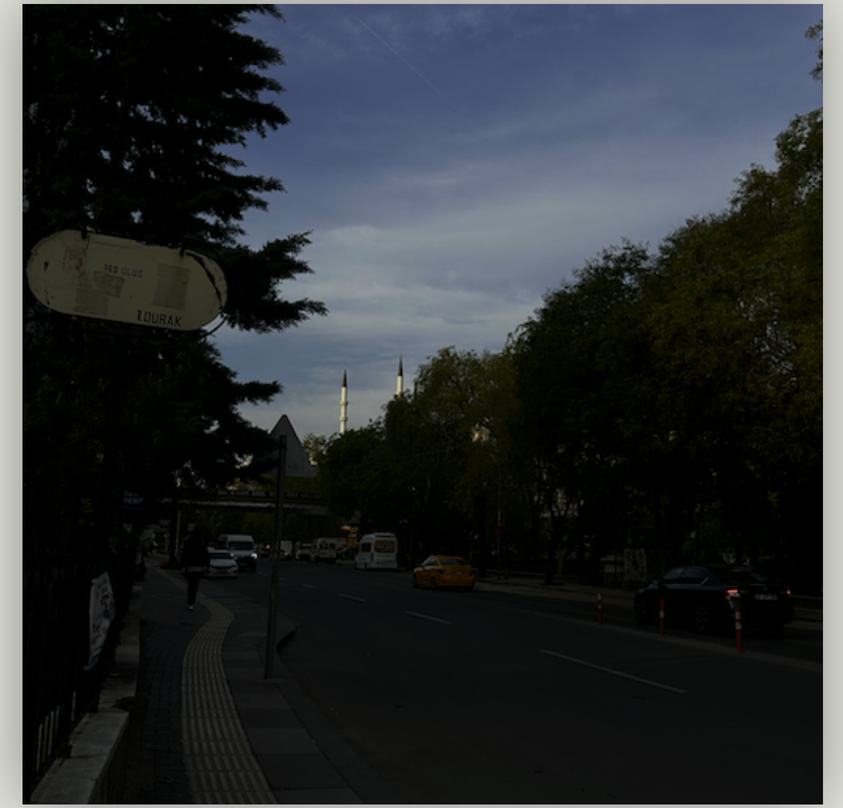
Predicted HDR with Reinhard Tonemapping

Some Examples from our custom images

Input LDR image

Predicted HDR with Reinhard Tonemapping

Some Examples from our custom images



Input LDR image

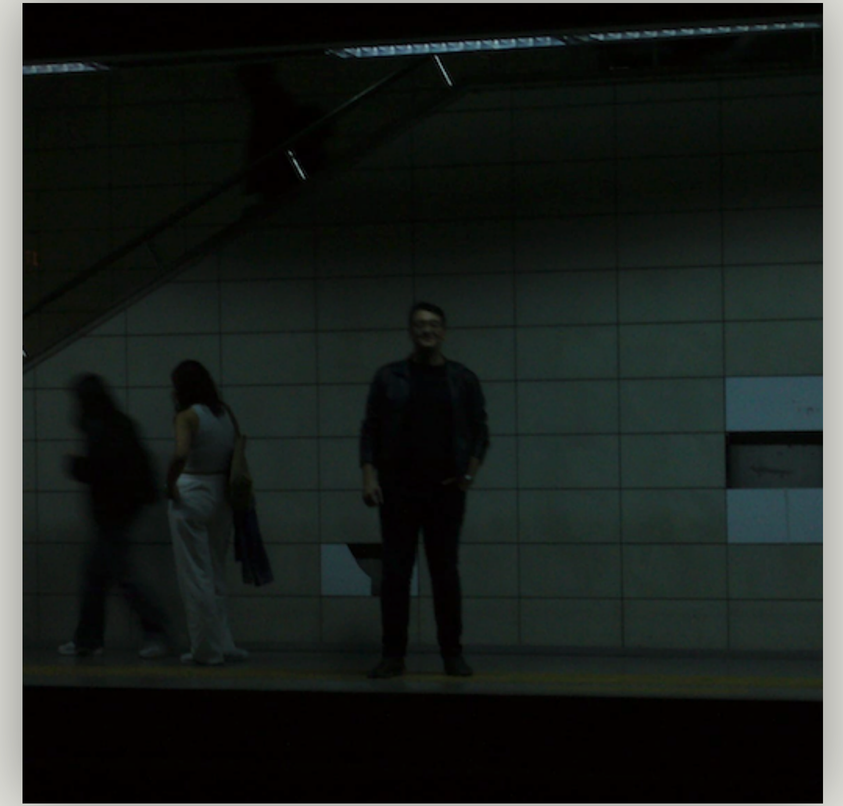
Predicted HDR with Reinhard Tonemapping

Some Examples from our custom images

Input LDR image

Predicted HDR with Reinhard Tonemapping

Some Examples from our custom images



Input LDR image

Predicted HDR with Reinhard Tonemapping

Discussion

- Our proposed hybrid autoencoder model demonstrated promising results even though it is a small-scale model.
- Qualitative results confirmed that the model is capable of enhancing contrast, recovering highlights, and producing perceptually appealing outputs across a wide range of scenes.
- However, the model still exhibits limitations in challenging scenarios.
- There exists a mosaic pattern in our models predictions. We think that this may be due to the pixel shuffling process. We may need to find better upsampling methods for future research.
- We have shown that our methodology works as a proof of concept. This is exciting since we use an encoder which was trained on a whole different task(classification) yet we can still use the features from this model for a totally different task(image2image translation).
- Also, our model is unreasonably good at low light image enhancement as can be seen from the examples. One may use this model to not only translate LDR images to logHDR, but to translate low light images to visually enhanced versions .

Open Questions

- How can we improve robustness in low-light regions?
- Can attention mechanisms or transformers improve artifact handling and global consistency?
- Would jointly training the model with perceptual or adversarial losses further enhance realism without sacrificing stability?
- Is it possible to adapt the model to mobile hardware for real-time HDR enhancement?
- How to reduce blurred artifacts? Can we mitigate the mosaic pattern which occurs in our predictions?
- Can we work on different color spaces and improve the performance?

References

- [1] Tursun, O.T., Akyüz, A.O., Erdem, A. and Erdem, E. (2015), The State of the Art in HDR Deghosting: A Survey and Evaluation. Computer Graphics Forum, 34: 683-707. https://doi.org/10.1111/cgf.12593
- [2] Yu-Lun Liu, Wei-Sheng Lai, Yu-Sheng Chen, Yi-Lung Kao, Ming-Hsuan Yang,Yung-Yu Chuang, and Jia-Bin Huang. 2020. Single-Image HDR Reconstruction by Learning to Reverse the Camera Pipeline. In IEEE Conference on Computer Vision and Pattern Recognition.
- [3] Eilertsen, Gabriel & Kronander, Joel & Denes, Gyorgy & Mantiuk, Rafal & Unger, Jonas. (2017). HDR image reconstruction from a single exposure using deep CNNs. ACM Transactions on Graphics. 36. 10.1145/3130800.3130816.
- [4] Alper Koz, Barış Demirkılıç, Yunus Bilge Kurt, Ahmet Oğuz Akyüz, Sinan Kalkan, A. Aydın Alatan, and Alan Chalmers. 2021. HDR Image Construction from Trifocal Multiexposure Images. In 2021 IEEE 23rd International Workshop on Multimedia Signal Processing (MMSP). 1–5. https://doi.org/10.1109/MMSP53017. 2021.9733440
- [5] Debattista. 2018. ExpandNet: A Deep Convolutional Neural Network for High Dynamic Range Expansion from Low Dynamic Range Content. CoRR abs/1803.02266 (2018). arXiv:1803.02266 http://arxiv.org/abs/1803.02266
- [6] Phuoc-Hieu Le, Quynh Le, Rang Nguyen, and Binh-Son Hua. 2022. Single-Image HDR Reconstruction by Multi-Exposure Generation. arXiv:2210.15897 [eess.IV] https://arxiv.org/abs/2210.15897