Deep Learning for Exposure Normalization on Regions of Interest in Digital Images

Vincent Wang Fat Chung   Rohit Aggarwal   Benjamin Bay

vwfchung@stanford.edu   rohitagg@stanford.edu   bay1@stanford.edu

CS230

Objective

To utilize Deep CNNs to generate high dynamic range (HDR) image representations using a single low dynamic range (LDR) photograph input image.

Figure 1: Left to right: an underexposed LDR image, an overexposed LDR image, an image generated with OpenCV's tone-mapping algorithms[10][11] (baseline), and the ground truth.

Data

Fairchild HDR Photographic Survey[4]

- 1035 LDR captures in 106 scenes, approximately 9 exposures per scene

Figure 2: Three example scenes from the ***Fairchild HDR Photographic Survey*** [4] dataset. The human eye is capable of sufficiently wide dynamic range to perceive detail in both regions, while machine sensors fail to do the same in LDR imaging mediums.

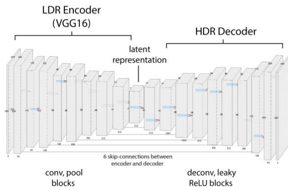
Data preparation

- $224 \times 224 \times 3$ 8-bit per pixel RGB images used (compatible with VGG-16), normalized at input to 0.0-1.0 value
- Data augmentation: scaling, croppping, mirroring, rotations
- Train/Development/Test split: 80/10/10, splits were performed on a per-scene basis to guarantee that the development and test scenes are completely new to the model
- Image registration (alignment) using OpenCV Error Correlation Coefficient[5] (issue root caused through error analysis)

Table 1: Dataset augmentation and splitting, no. of samples.

Dataset split		
Train	Dev	Test
10318	1260	1260

Model



- Total parameters: $22,467,992$, with 58.96% trainable and 41.04% non-trainable in Keras model[9]
- Three major blocks: the LDR Encoder, the compressed latent layer and the HDR Decoder
- LDR Encoder is the VGG-16 with pre-trained ImageNet weights[8]
- HDR Decoder (upsampling deconvolution layers) to reconstruct image, intended to mirror decoder downsampling convolutional layers
- Skip connections added between encoder and decoder to enable efficient information transfer from decoder to encoder

Architecture and Hyperparameter Considerations

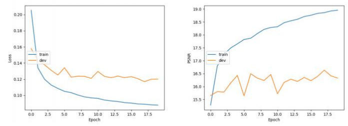
- Tried both LeakyReLU and ReLU activations, sigmoid activation for output layer
- Dropouts used to prevent overfitting
- L2 regularization for intermediate layers (bias and kernel)
- Transfer Learning through VGG-16 ImageNet weights
- In VGG-16 encoder, only the layers with direct skip connection to decoder were set trainable

Loss func: Total Variation + L1 Loss

$$\mathcal{L} = \sum_{i,j} \sqrt{(y_{i,j+1} - y_{i,j})^2 + (y_{i+1,j} - y_{i,j})^2} + \frac{1}{N} \sum_{i,j} |y - \hat{y}|, \tag{1}$$

where N is the number of pixels.

Results



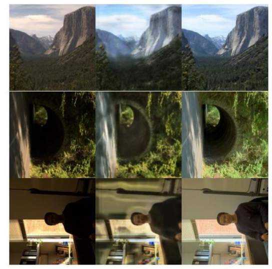


Figure 3: Development dataset examples through the final model. Left to right: Input (x), Reconstruction ($\hat{y}$), and Ground Truth (y).

Figure 4: Under- and overexposed input reconstructions.

Figure 5: Feature reconstruction error on dome.

Table 2: Dataset PSNR metrics for baseline (OpenCV tone-mapping) and final neural network (Convolutional Autoencoder) models.

Model	Dataset PSNR [dB]		
	Train	Dev	Test
Baseline (OpenCV)	27.90	27.80	27.81
Neural Network	16.35	16.10	15.05

Discussion

Our model functions as a proof-of-concept, since far more data would be needed to raise our test PSNR of 15.05 dB above our baseline of 27.81 dB and fellow researchers.

Future improvements

- Incorporate hybrid network components (GAN architechture, etc.)
- Discriminate for image quality metrics (e.g., PSNR or SSIM)
- Increase computation power (up from 1 AWS GPU)
- Train and tune more layers in the network
- Perform more systematic hyperparameter tuning

With only 106 scenes, our model learned to reconstruct objects present in over 10,000 images. To avoid overfitting, this task requires much more data than we were able to acquire in reasonable time.

Acknowledgements

- Thanks to CS230 TA Abhijeet Shenoi for counseling us on our project
- Thanks to Professor Andrew Ng, Kian Katanforoosh, and the Stanford Center for Professional Development for this research project opportunity
- For our implementation, see `https://github.com/BayBenj/cs230-proj`

[1] S. Mann, "Compositing multiple pictures of the same scene," in *Proc. IS&T Annual Meeting, 1993*, pp. 50–52, 1993.

[2] 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," in *Computer Graphics Forum*, vol. 37, pp. 37–49, Wiley Online Library, 2018.

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

[4] M. D. Fairchild, "The hdr photographic survey," in *Color and imaging conference*, vol. 2007, pp. 233–238, Society for Imaging Science and Technology, 2007.

[5] K. M. Williams, R. W. Schulte, K. E. Schubert, and A. J. Wroe, "Evaluation of mathematical algorithms for automatic patient alignment in radiosurgery," *Technology in Cancer Research & Treatment*, vol. 14, no. 3, pp. 326–333, 2015.

[6] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.

[7] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th international conference on Machine learning*, pp. 1096–1103, ACM, 2008.

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

[9] F. Chollet *et al.*, "Keras." `https://keras.io`, 2015.

[10] P. E. Debevec and J. Malik, "Recovering high dynamic range radiance maps from photographs," *ACM SIGGRAPH 2008 classes*, p. 31, 2008.

[11] E. Reinhard and K. Devlin, "Dynamic range reduction inspired by photoreceptor physiology," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 11, no. 1, pp. 13–24, 2005.