HDR Image Construction from a LDR Dataset for Deep Learning

Michael John Curry

SUNY New Paltz

August 24, 2022

- The best way to improve the performance of a machine learning model is to train it on more data. The more examples the model has to learn from, the better it will be able to recognize which differences in images matter and which do not.
- One easy way of getting more data is to use the data you already have. If we can transform the images in our dataset in ways that preserve the class, we can teach our classifier to ignore those kinds of transformations.
- Some examples are rotation, translation, flipping, brightness, and contrast.

- Are there other methods to increase the amount of data contained in an image?

- High-dynamic-range (HDR) imaging, an important field in image processing, computer graphics/vision, and photography, is a technique that allows a greater dynamic range of exposures than traditional imaging techniques. It aims to accurately represent a wide range of intensity levels captured in real scenes, ranging from sunlight to shadows.

- Conventional HDR imaging mainly uses special HDR cameras to capture HDR images. These cameras are prohibitively expensive for general users.

- All images captured by consumer, pro-consumer, and most professional cameras. These images make up the vast majority of the images captured each day.

- A method to reconstruct HDR images from the visual content captured by low-dynamic-range (LDR) cameras using specially designed algorithms.
- One common approach is to capture multiple exposures of a scene and then combine them.

- If the scene was not originally captured with multiple exposures, are there ways to bring back the lost information?

- GAN networks, Deep neutral networks, and virtual environment rendering techniques are all used to create a HDR image from a single LDR capture.
- All these approaches are computationally expensive.

- Is there a naive approach?

Original Image

Figure 1: Original Image

- Data loss in both Highlights and Shadows

Multiply Blending Mode

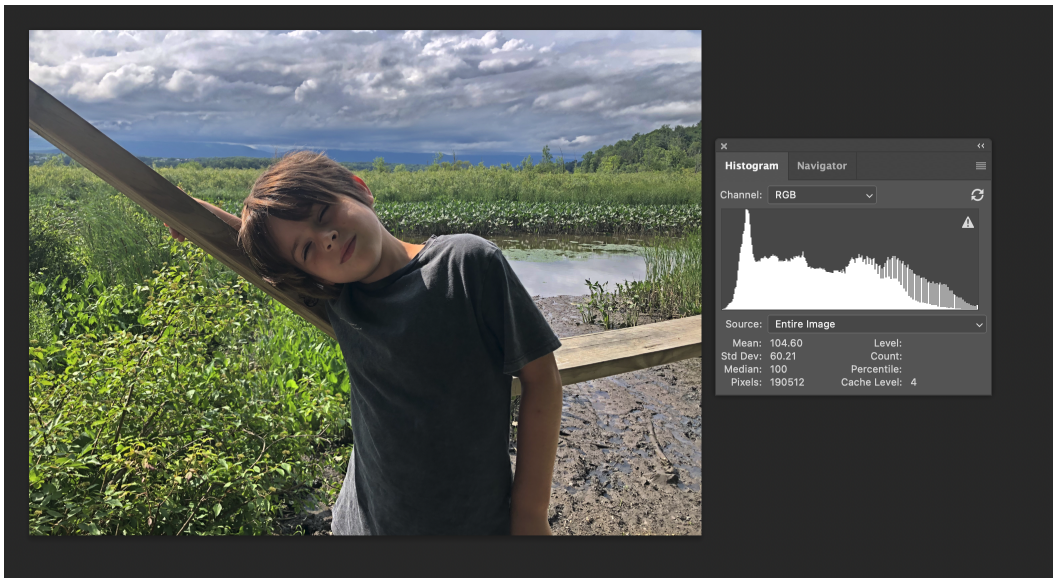

Figure 2: Multiply

- All pixel luminosity values in the lightest third have Multiply blending mode applied.

$$f(a, b) = ab$$

Screen Blending Mode

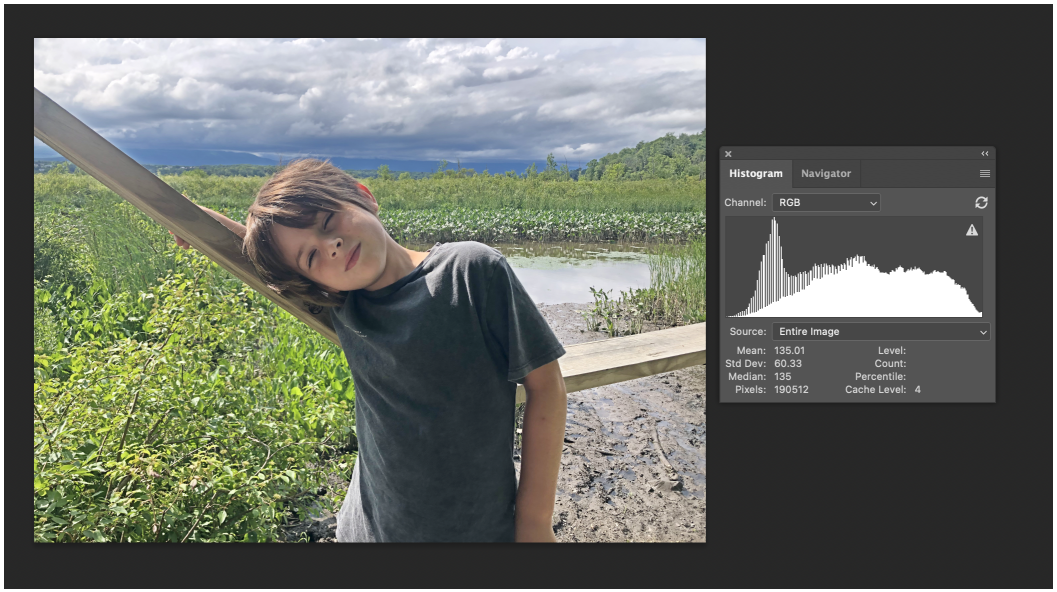

Figure 3: Multiply

- All pixel luminosity values in the darkest third have Screen blending mode applied.

$$f(a, b) = 1 - (1 - a)(1 - b)$$

Combine all three images

Figure 4: Combined

- All three images are stacked.

- Highlight information is taken from the image with the Multiply blending mode applied.
- Mid-tone information is taken from the original unprocessed image.
- Shadow information is taken from the image with the Screen blending mode applied.