Machine Learning to Correct for Lens Imaging Defects

Teaghan O'Briain

University of Victoria, Physics and Astronomy

OVERVIEW

- Developing an idealized lens configuration for a given task is not always realistic, and therefore, the resulting observations can be of lower quality and may even suffer from deformations.
- The image quality can suffer from several defects such as:
  - **Defocus** as a result of not imaging in the focal plane.
  - **Aberrations** such as those from using spherical lenses.
  - **Diffraction** limitations due the size of the lens.
- Solutions to these limitations are not always practical and therefore the obtained images may require processing to enhance the quality [1].
- The aim of this project is to automate this processing using a Machine Learning Neural Network.

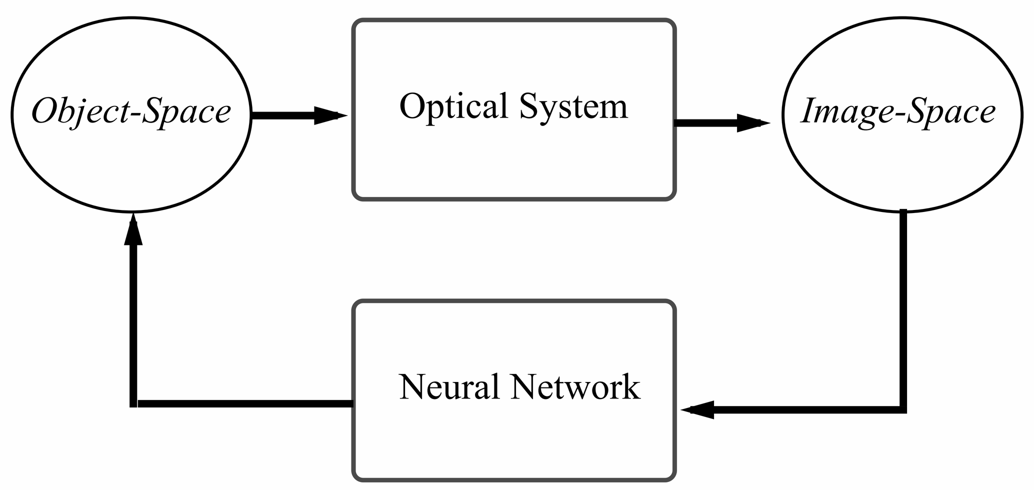


Figure 1: An overview of the proposed method. First the *object* is traced through the optical system to form an *image*. This image is then processed using a Neural Network, which mimics the inverse ray-tracing of the optical system to therefore reproduce the object.

HUMAN NEURAL NETWORK IMAGE PROCESSING

- A useful analogy that shares shocking similarities to this project is the human eye, which can be thought of as a single lens optical system.
- Light rays from real objects travel through the lens of the eye, refract, and then hit the retina at the back of the eye where the photons are detected by photoreceptors.

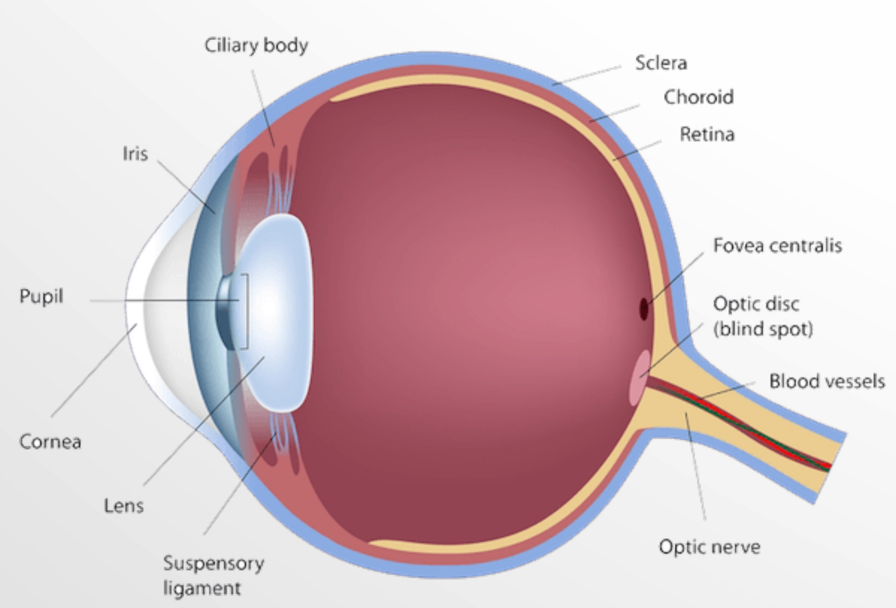
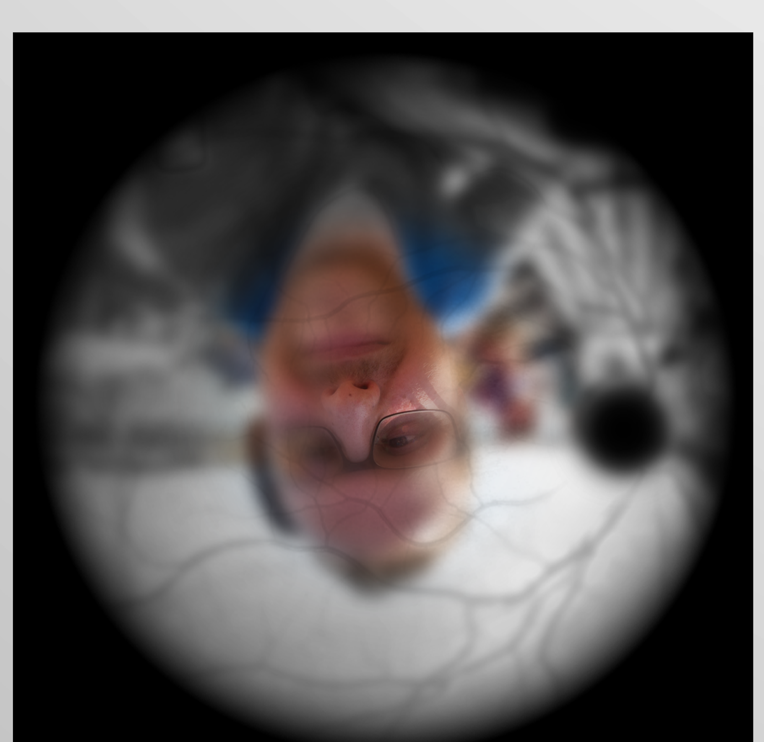


Figure 2: The anatomy of the human eye (taken from [2]).



- The eye cannot focus objects at all distances equally, and therefore, the image formed on our retina is both inverted and blurry.
- In the early stages of our mental development, our brains learn to translate these blurry images into what we *see* today (or imagine).
- This learned translation is done through neurons that connect the photoreceptors to the brain.
- We will perform a similar procedure using a *Machine Learning* Neural Network.

Figure 3: A representation of what an image looks like when it forms at the back of the human eye (taken from [3]).

RAY TRACING with PYRATE

- Our optical system is modelled using the ray tracing tool, Pyrate [4], where a series of lenses with varying radii of curvature, indices of refraction, and thicknesses are compiled in series to resemble the laboratory lens set-up.

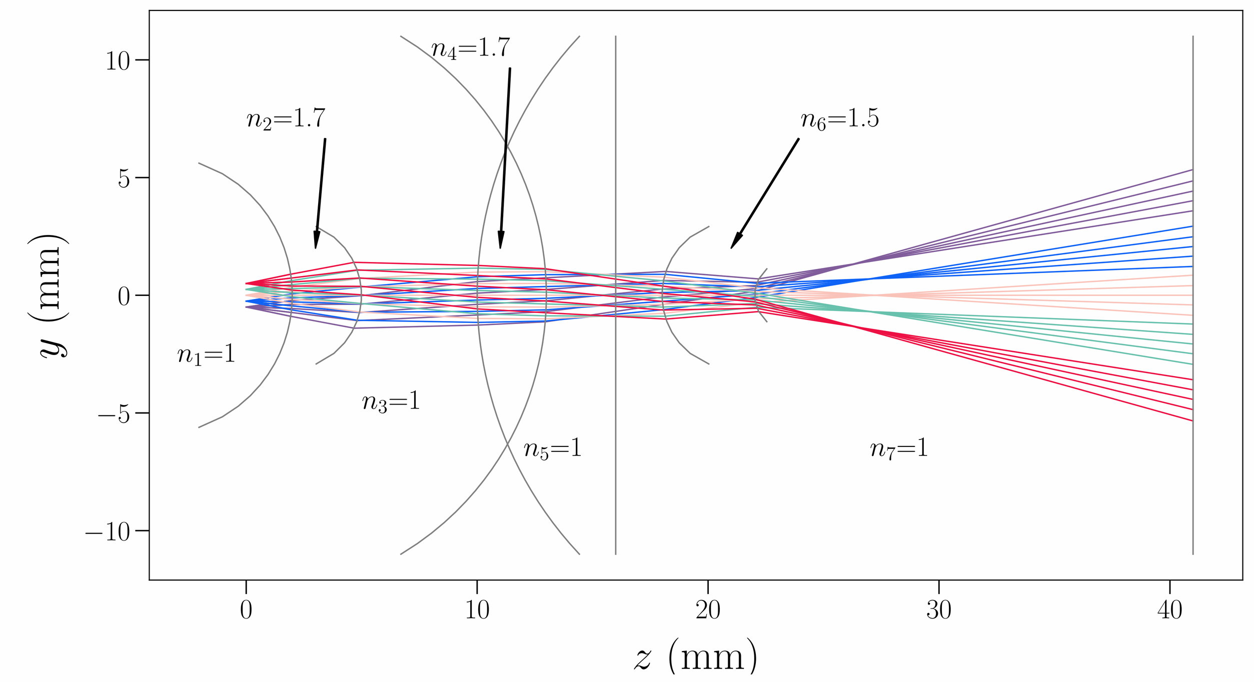


Figure 4: A diagram of the optical system used for this experiment. The first lens is placed 2mm from the object and is 3mm thick with R_1=-5.992mm, R_2 = -3.160mm, and n=1.7; the second lens is placed 10mm from the object and is 3mm thick with R_1= 15.884mm, R_2 = -12.756mm, and n=1.7; and the third lens is placed 18mm from the object and is 4mm thick with R_1= 3.125mm, R_2 = 1.479mm, and n=1.5. A stop aperture is also used at 16mm from the object and the image is formed 41mm from the object.

- The object is represented as an array of point sources (see Figure 5 top left), each of which consists of a bundle of 300 light rays whose directions are distributed radially.
- Each light ray is traced through the optical system to a unique location on the image plane (see Figure 4). This image plane can be split into an array of pixels, emulating a CCD detector (see Figure 5 bottom left).
- The accumulation of all of the ray tracings allows for a distortion translation to be calculated and this deformation was then applied to the MNIST dataset of hand written digits [5] (see Figure 5 right).
- For the MNIST dataset, the pixel grids are 28x28, so the image defects are more severe than the example shown below on the left.

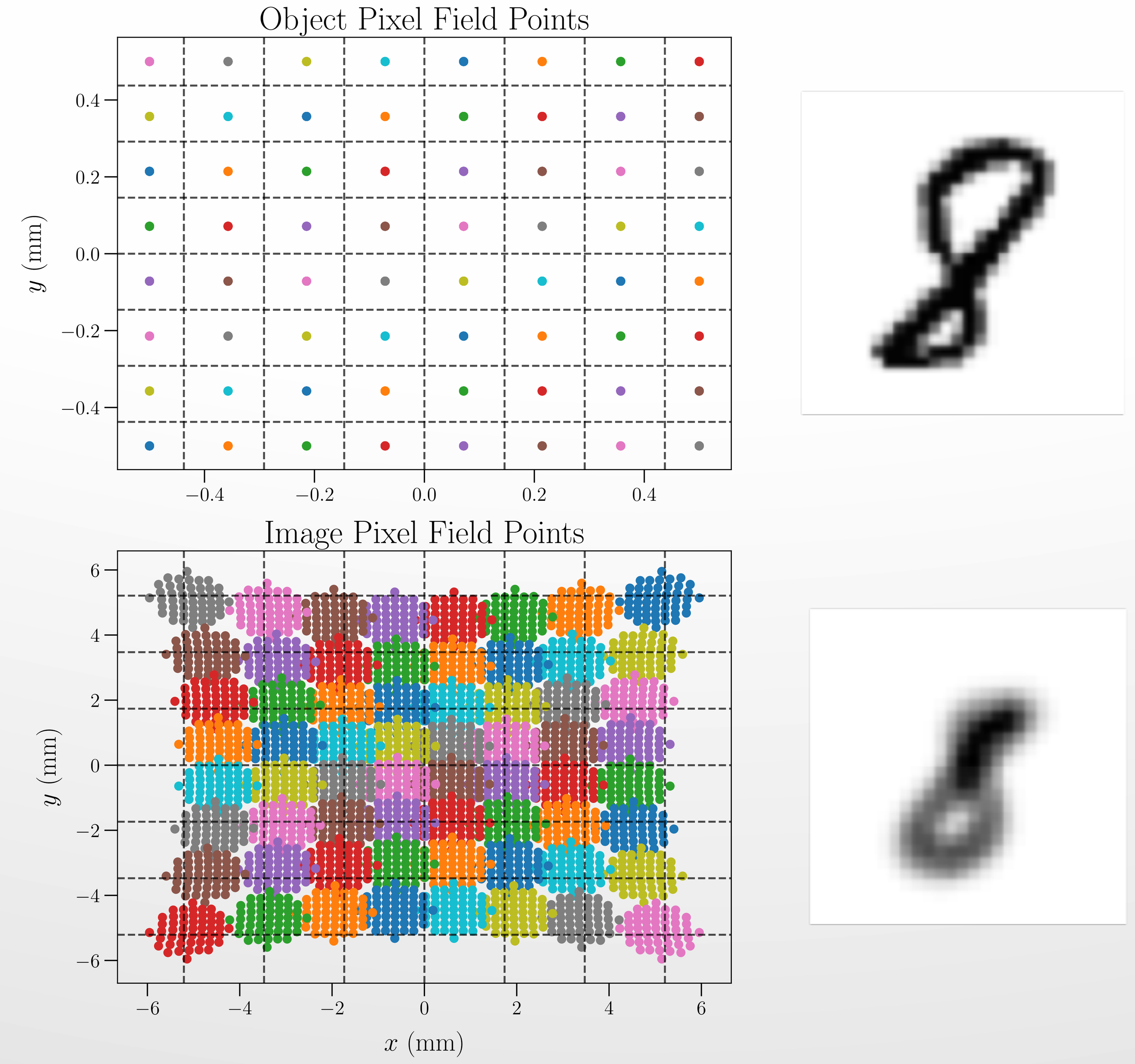


Figure 5: The left plots show an example of how an 8x8 grid of *object* pixel field points – each emitting 70 light rays – are traced through the optical system and produce a smeared distribution of *image* field points. The grids shown on the image side are meant to emulate a CCD detector where – in many cases – multiple object field points contribute to a single image pixel. The right plots show how a similar deformation applied to a 28x28 grid of object field points – each emitting 300 light rays – distorts the produced image.

COMPUTER NEURAL NETWORK IMAGE PROCESSING

- Similar to how the human eye produces blurry, inverted images and the neurons in the brain learn – from experience – how to process these images to produce nicely resolved images that we *imagine*, we can use a machine learning Neural Network that *learns* from our dataset.
- The architecture for our Neural Network is an auto-encoder that takes the deformed images as *inputs* and learns to produce the original object as the *output* (see Figure 6).
- Initially, just like the human brain, the network is not effective at processing the images. However, over time – as the network sees more samples – it learns how to accurately translate from the *image-space* to the *object-space* until it reaches a desired level of performance (see Figure 7).

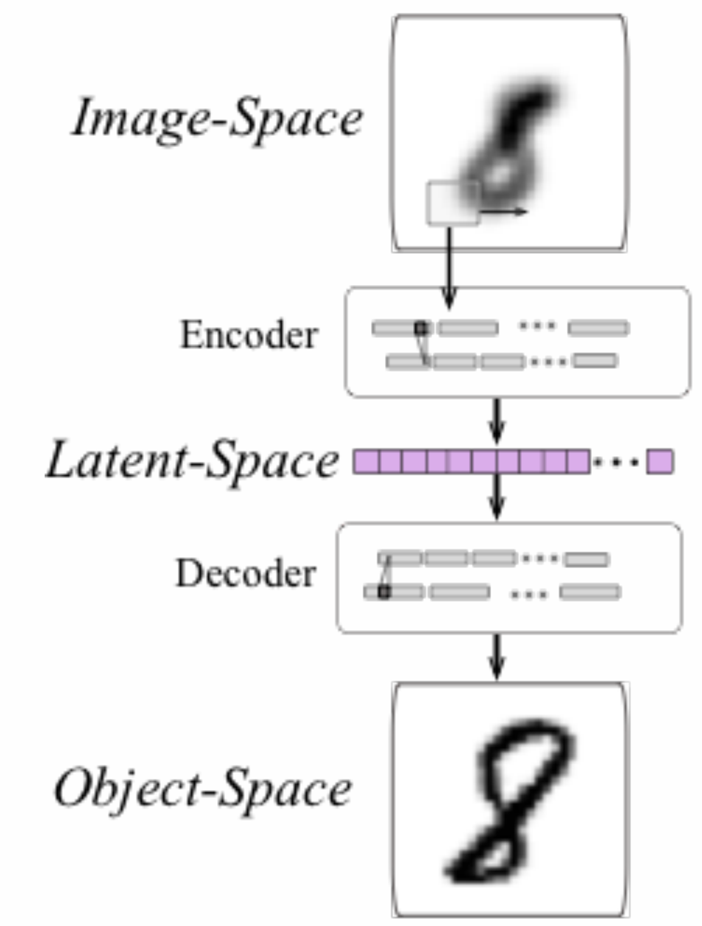


Figure 6: A diagram of the auto-encoder architecture. Both the encoder and decoders consist of two convolutional hidden layers with 32 and 64 filters (in subsequent order according to the encoder). The latent-space consists of 128 nodes.

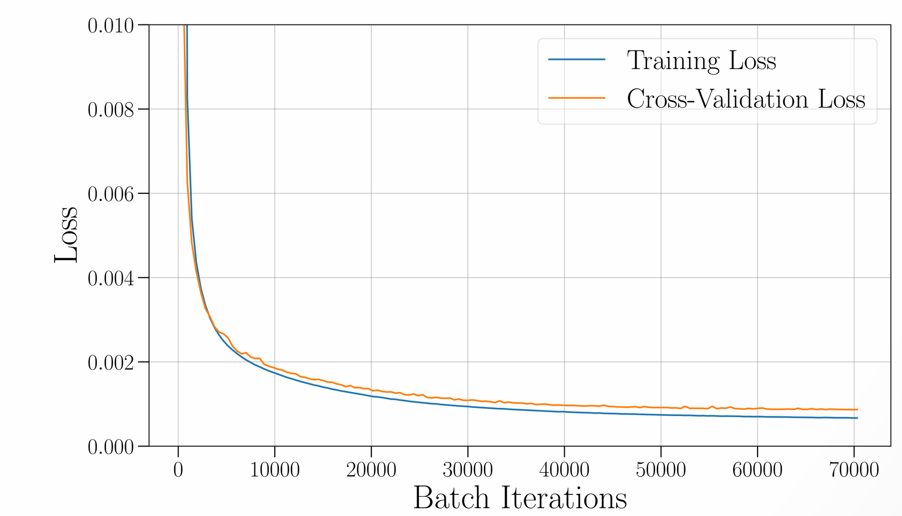
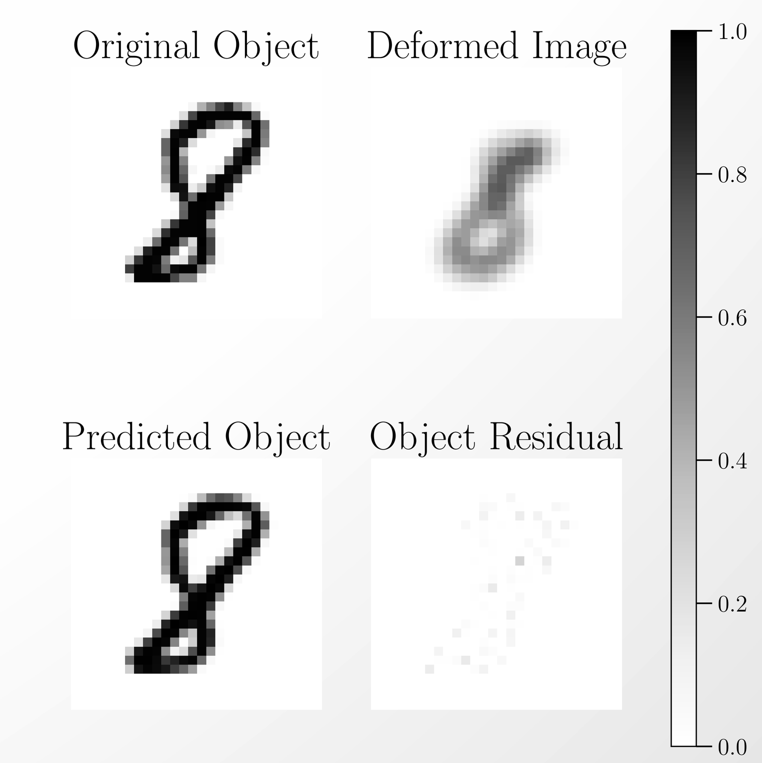


Figure 7: The training progress of the network over time.



- Once the network has been trained, it can now receive new deformed images as inputs and produce how that image *should* look like if it were imaged more precisely.
- The auto-encoder is capable of correcting images well and produces the object counter-part accurately, as seen in Figures 8 and 9.
- The results provide confidence that the model has learned the inverse ray-tracing of the optical system, which was the initial objective.

Figure 8: An example of an MNIST test sample. The original object is shown in the top left. When this object is traced through the optical system, the image produced is shown in the top right. The Neural Network (Figure 6) is able to take the deformed image as an input and predict the object, seen in the bottom left. The residual between the original and predicted object is shown in the bottom right.

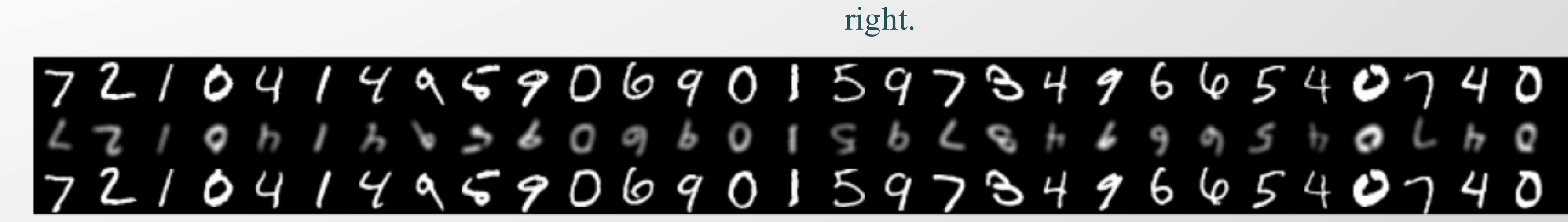


Figure 9: Thirty more test examples of the Neural Network results. The top row shows the original MNIST objects, the middle row shows the distorted images, and the bottom row shows the model's predictions.

REFERENCES

[1] Jiangpeng Rong, Shiyao Huang, Zeyu Shang, and Xianghua Ying. Radial lens distortion correction using convolutional neural networks trained with synthesized images. In Asian Conference on Computer Vision, pages 35–49. Springer, 2016.

[2] From kisspng, accessed 23 March 2019, https://www.kisspng.com/png-human-eye-diagram-retina-anatomy-eye-anatomy-3344144/download-png.html.

[3] Lorne Whitehead. Understanding the Remarkable Accuracy of Colour Perception. University of Victoria, Physics and Astronomy Colloquium, 21 February 2018.

[4] Moritz Esslinger, Johannes Hartung, Uwe Lippmann, and Thomas Heinze et al., "Pyrate: Optical raytracing with Python", https://www.github.com/mess42/pyrate. GitHub Repository. Accessed 08 March 2019.

[5] Yann Lecun and Corina Cortex "MNIST handwritten digit database." (2010).