

1 Histograms and Pixel Operations

1.1 Linear Pixel Operations (10 Points)

Edit the Jupyter Notebook `Affine_Transformation.ipynb`. The notebook loads several degraded versions of a grayscale image and displays it. In this exercise, we use histograms to understand the type of degradation, then correct the images.

- a) In OpenCV, histograms can be calculated using the function `cv2.calcHist`. The signature is not very self-explanatory, though. Please look up the function signature on the internet. We use the call `cv2.calcHist(([image], [0], None, [256], [0,256]))`. Please describe each term.

`[image]` : The images from which we want to receive the histogram. It has to be wrapped in a list.

`[0]` : The channels we want the histogram of. As we use a grayscale image, only `[0]` (black) is possible. For rgb images you can use `0,1` or `2` to get a histogram of 1 of the 3 channels.

`None` : Here we could pass a mask, to limit the area from which the histogram is made.

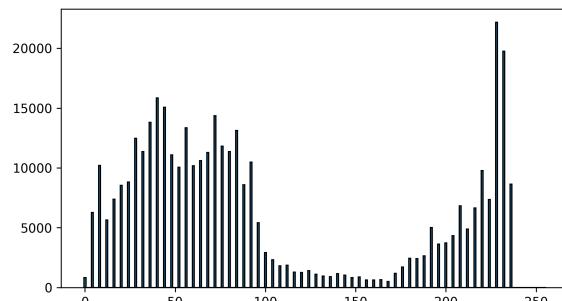
`[256]` : Here we can pass our histograms bin count. We use `256`, to get every grayscale as its own bin.

`[0,256]` : The colorspectrum we want to use. Here we use the whole 8bit range. Excluded ranges will be also excluded from the histogram.

- b) Describe quantitatively, what is the problem with `kodim15_gray_low_contrast.png`.

As the histogram shows, the picture only exists of gray in the range from around 100 to 160. Other shades of gray are not appearing. This low variance causes the contrast in the picture to be very bad.

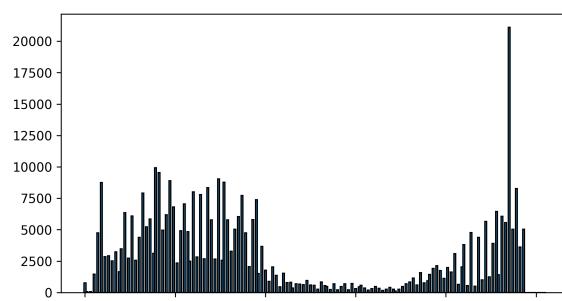
- c) Extend the notebook to correct the image.



- d) Describe quantitatively, what is the problem with `kodim15_gray_ueberbelichtet.png`.

The image is too bright. We can also see this in the histogram, as there are no low values, representing dark shades.

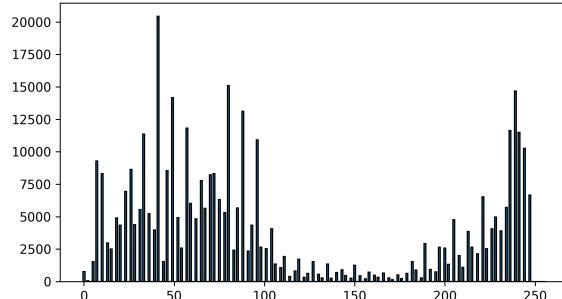
- e) Extend the notebook to correct the image.



- f) Describe quantitatively, what is the problem with `kodim15_gray_unterbelichtet.png`.

The image is too dark. We can also see this in the histogram, as there are no values above 100, representing the brighter shades.

- g) Extend the notebook to correct the image.



I needed the following time to complete the task: 30min

1.2 Non-Linear Pixel Operations (10 Points)

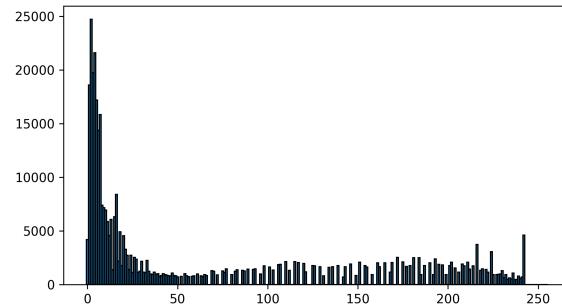
Implement a non-linear correction of the image. We use the function

$$\Phi(x) = \frac{e^x}{1 + e^x}$$

. The function is defined on $[-3, .., 3]$.

- a) Prepare the function for use as a pixel operation. Adapt the color values to match the definition range of the function, then apply the function on the image `kodim15_unterbelichtet.png`.
- b) The image is an rgb image, not grayscale. Describe in your own words, how the different channels need to be treated for brightness or contrast correction. What would happen, if different functions are applied to each channel?

All channels need to be treated the same way. If you apply different functions to the channels, the colors in the image will shift in weird ways, depending on how much the functions differ.



1.3 Non-Linear Pixel Operations with parametrized Function (10 Points)

Finally, we want to use the exponential function

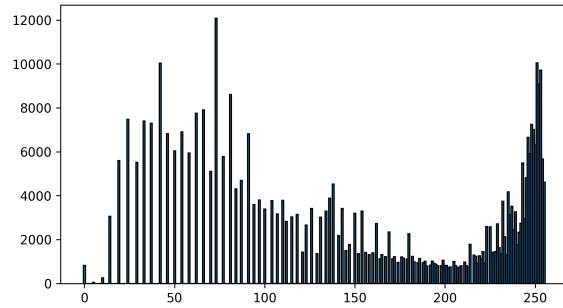
$$\Phi(x) = \frac{G - 1}{e^{a \cdot (G - 1)} - 1} \cdot (e^{ax} - 1)$$

. Hereby, G is the upper bound of our value range, i.e. $G - 1$ is 255. The parameter a can be selected for the intended correction.

I needed the following time to complete the task: 30min

- Prepare the function for use as a pixel operation. Adapte the color values to match the definition range of the function, then apply the function on the image
The color values are already matching the definition range. `kodim15_unterbelichtet.png`.
- Determine a good value for a . Describe your procedure for determining a good value in your own words.

I used -0.01 . The color values in the picture are heavily on the dark side, so one needs a negative value to brighten the picture up. Just by trying out, -0.1 is too bright and -0.001 not noticeable enough. For my eyes everything from -0.008 to -0.01 was in an acceptable range. Even -0.02 already was too bright again.

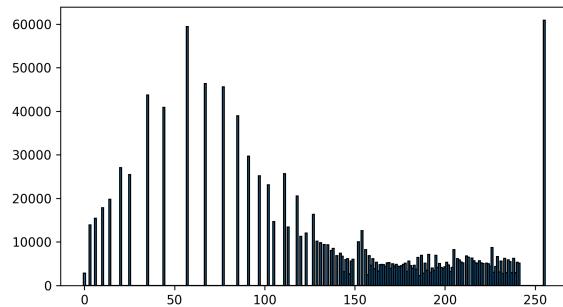


I needed the following time to complete the task: 15min

1.4 Histogram Egalization (30 Points)

We now want to implement histogram equalisation.

- Implement a method that corrects an image by histogram equalisation. .



I needed the following time to complete the task: 60min