Friedrich-Alexander-Universität
Technische Fakultät

IntroML - Exercise 2 - Thresholding and Histograms

Exercises for
Introduction to Machine Learning
Paul Stoewer, Johannes Enk, Changhun Kim, Yu Gao, Haiting Huang, Mohamad El Hage Ali
Task 2: deadline: 31. May 2024

Exercise 02: Thresholding and Histograms

Exercise 1 Otsu's Thresholding

Many thresholding algorithms exist for image binarization where the goal of this process is to split an image into two parts. For example in document processing, the document has to be split into page and writings. One widely used approach is Otsu's thresholding, which was also discussed in the lecture. In this task, you have to implement Otsu's thresholding according to the lecture slides. Use the code skeleton provided. You are only allowed to use the predefined imported modules (in this case - numpy only). The function should work by running the given *main.py* module.

The steps of Otsu's are as follows:

(a) Create Histogram from image with 256 bins

(b) For every bin assume that this bin is the threshold for binarization and calculate for each threshold:

  (a) Calculate p_0, p_1

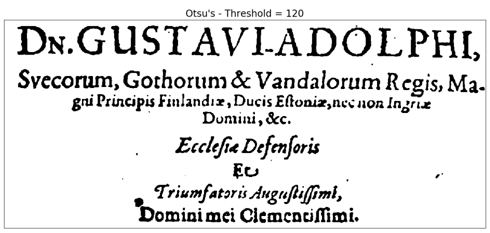
  (b) Calculate μ_0, μ_1

  (c) Calculate between class variance $\sigma^2_{inter} = p_0\, p_1\, (\mu_1 - \mu_0)^2$

(c) The threshold (= bin) with the highest between class variance is the threshold

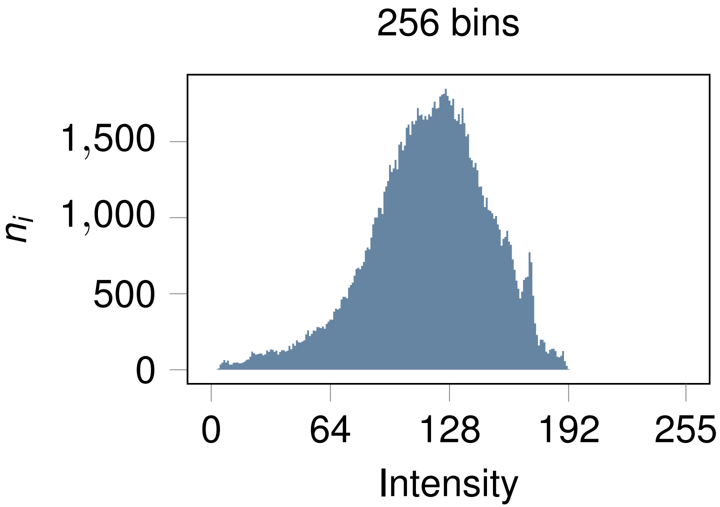
(d) Binarize the image (0,255)

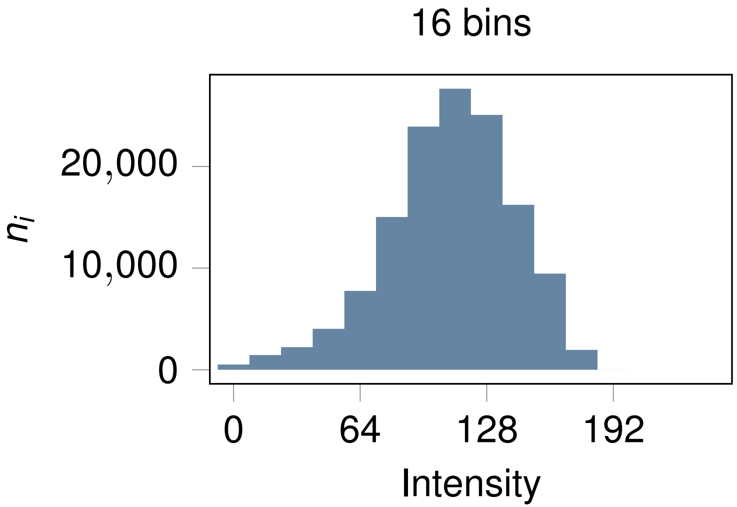
Output Example:

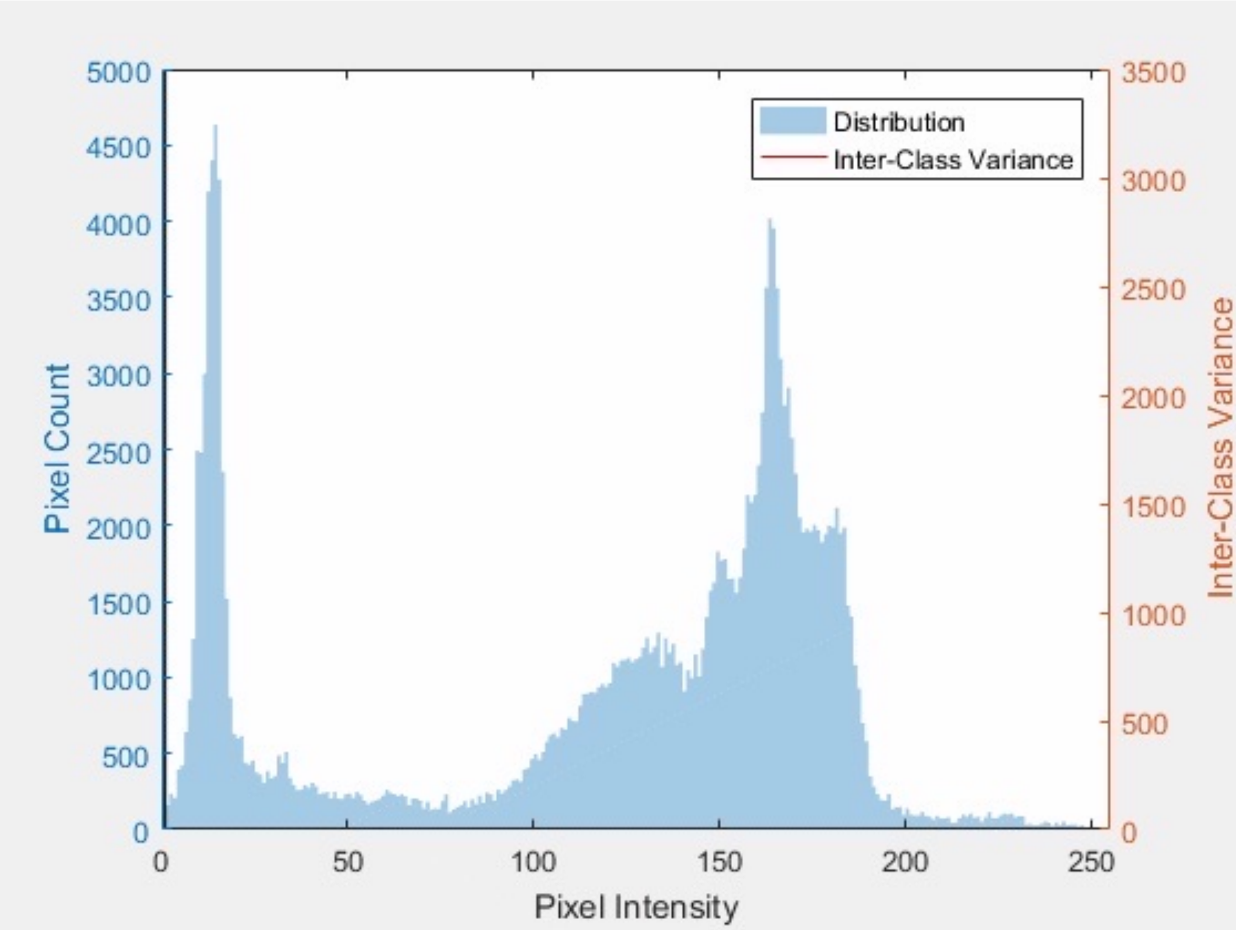


Histograms of Images

Histograms show the distribution of intensity values, grouped into bins.






Distribution
Inter-Class Variance
Pixel Count
Inter-Class Variance
Pixel Intensity
0
50
100
150
200
250
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000
0
500
1000
1500
2000
2500
3000
3500

Otsu's Thresholding [1]

Ideal case:

- Two peaks representing object(s) and background in a histogram
- Deep and sharp valley between the two peaks
- Threshold is the bottom of the valley

Most real images:

- Difficult to detect the valley bottom precisely due to:
  - Flat and broad valley bottom
  - Noise
  - Extreme differences between the heights of the two peaks

Otsu's Thresholding

Probabilities of the class occurrences:

$$p_0 = \sum_{x=0}^{\theta} p(x) \qquad p_1 = \sum_{x=\theta+1}^{x_{\max}} p(x)$$

Class mean levels:

$$\mu_0 = \frac{1}{p_0} \sum_{x=0}^{\theta} x \cdot p(x) \qquad \mu_1 = \frac{1}{p_1} \sum_{x=\theta+1}^{x_{\max}} x \cdot p(x)$$

Total mean level:

$$\mu = p_0\,\mu_0 + p_1\,\mu_1$$

Otsu's Thresholding

Class variances:

$$\sigma_0{}^2 = \frac{1}{p_0} \sum_{x=0}^{\theta} (x - \mu_0)^2 \cdot p(x) \qquad \sigma_1{}^2 = \frac{1}{p_1} \sum_{x=\theta+1}^{x_{\max}} (x - \mu_1)^2 \cdot p(x)$$

Within-class variance:

$$\sigma_{\text{intra}}{}^2 = p_0\, \sigma_0{}^2 + p_1\, \sigma_1{}^2$$

Between-class variance:

$$\sigma_{\text{inter}}{}^2 = p_0 (\mu_0 - \mu)^2 + p_1 (\mu_1 - \mu)^2$$

Otsu's Thresholding

Relationship between within-class, between-class, and total variance:

$$\sigma_{\text{total}}{}^2 = \sigma_{\text{intra}}{}^2 + \sigma_{\text{inter}}{}^2$$

Otsu's thresholding criterion:

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}}\, \sigma_{\text{inter}}{}^2 = \underset{\theta}{\operatorname{argmax}}\, p_0\, p_1\, (\mu_1 - \mu_0)^2$$

- Maximizing the squared distance between the 2 mean values
- Multiplied by the prior class probabilities (favoring equal priors);
- Equivalent to minimizing the within-class variance

Exercise 2 Histogram Equalization

You made a picture of your wonderful cat. Unfortunately, the stupid camera can not make good pictures of your cat, as your cat has only few contrast in her awesome fluffy fur. So you decide to enhance the histogram of the picture. In introduction to machine learning you learned that histogram equalization is a good choice, so you choose to implement it:

(a) Load `hello.png` into a numpy array, using, for example, PIL or opencv

(b) Compute the intensity histogram of your cat image, for pixel values between 0 and 255

(c) Compute its cumulative distribution function C

(d) Change the gray value of each pixel:

$$pixelvalue_{new} = g(pixelvalue_{old}) = \lfloor \frac{C(pixelvalue_{old}) - C_{min}}{1 - C_{min}} \cdot 255 \rfloor \quad (1)$$

(e) Save the result as `kitty.png`

(f) During the submission, explain why the background looks like this, i.e., with clear intensity boundaries

Important: Do not use any function that computes histograms automatically, histogram equalization, or cumulative sums. Any computation on the pixel values, or any modification of their values, must be completely done with your own code.[1]

Note: If the histogram is computed correctly, then the sum of its 90 first values will be equal to 249.

Note: If the cumulative distribution is computed correctly, then the sum of its 90 first values will start with the following: 0.001974977

Note: Your result should look as the picture below. Did you know that this cute cat is extremely intelligent and can open doors?

Histogram equalized image:

[1]We would really like to make sure that you all know how to produce histograms, compute cumulative distributions, and do an histogram equalization.

Histogram Equalization

Idea of histogram equalization

- Method to improve images with a limited range of values
- Redistribution of intensities
- Based on histograms
- Effectively spreads out the most frequently used intensities
- Goal: image with uniform distribution of intensities

Histogram Equalization (cont.)

Formal definition

- Transformation $g(\cdot)$ of sample values $f_1, f_2, \ldots$ such that $g(f_i)$ is $[0, 255]$-uniformly distributed:
  - $g_i = g(f_i) \in [0, 1]$
  - Uniform distribution of $p(g_i)$
- Solution based on the cumulative distribution

Histogram Equalization (cont.)

Cumulative distribution

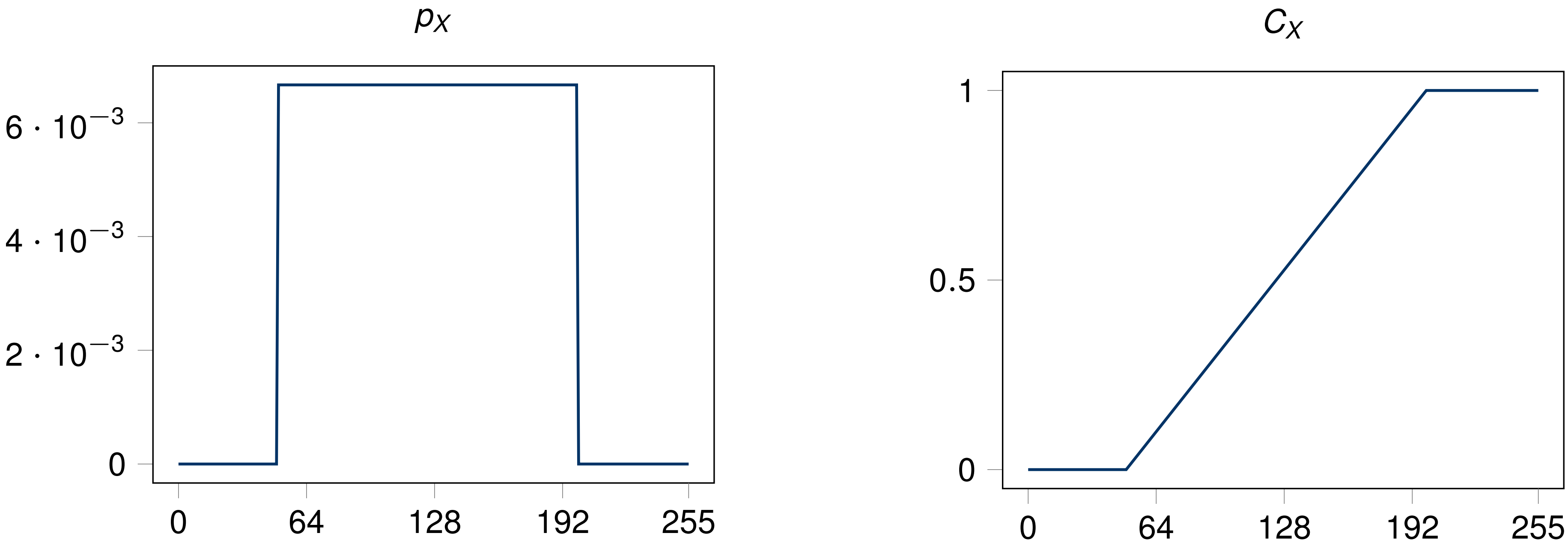
- Let $p_X(x)$ be the probability density function (pdf) of X
- cumulative distribution function (cdf) of X:

$$C_X(x) = \int_{-\infty}^{x} p_X(u)\,\mathrm{d}u$$

- Monotonic function
- Approaches 1 at infinity: $\lim_{x \to \infty} C_X(x) = 1$
- and 0 at minus infinity: $\lim_{x \to -\infty} C_X(x) = 0$

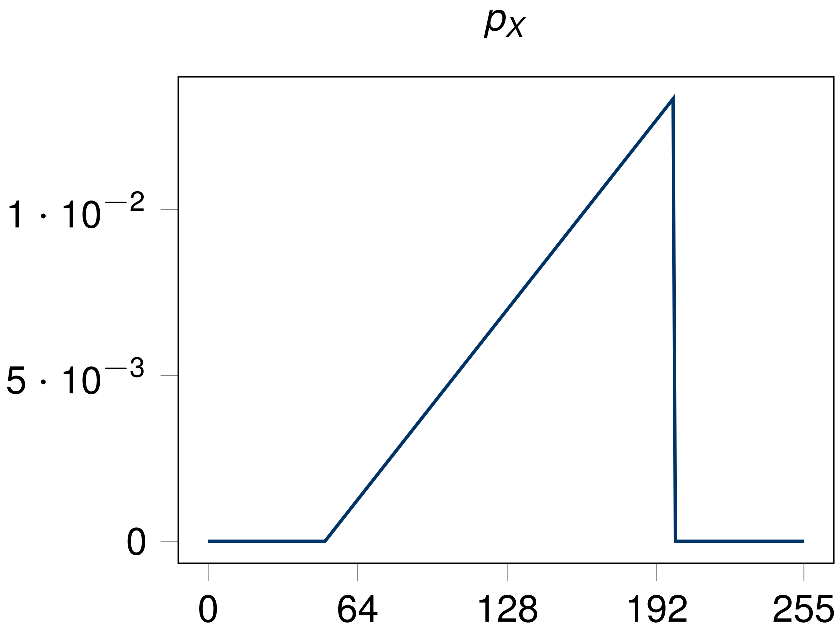
Histogram Equalization (cont.)

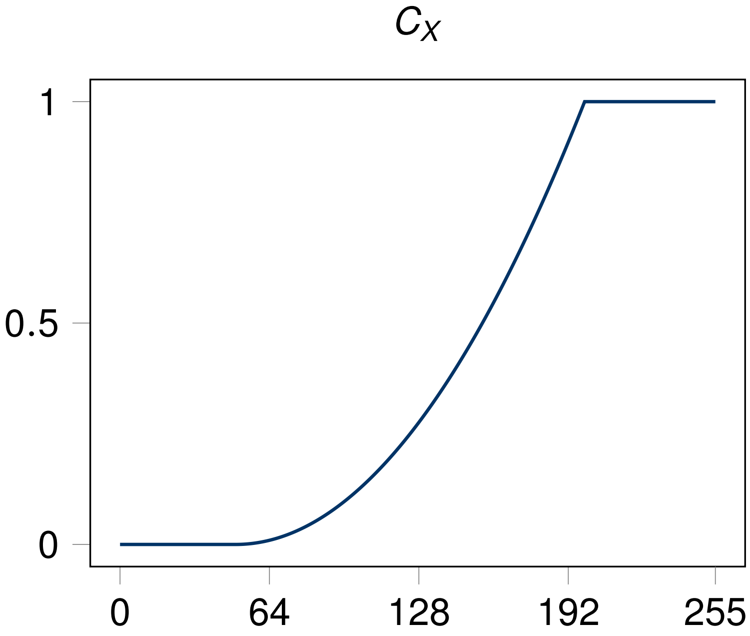
Cumulative distribution: examples



Histogram Equalization (cont.)

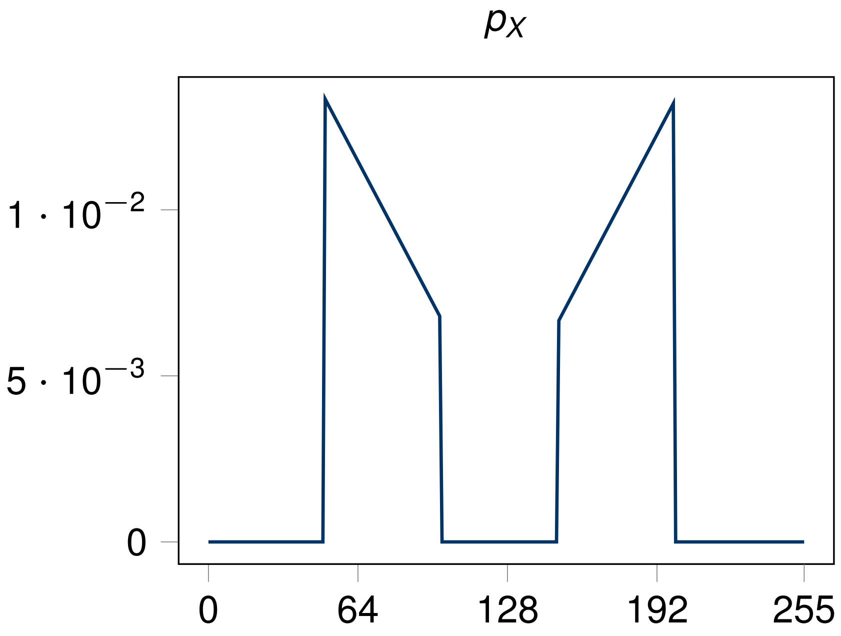
Cumulative distribution: examples

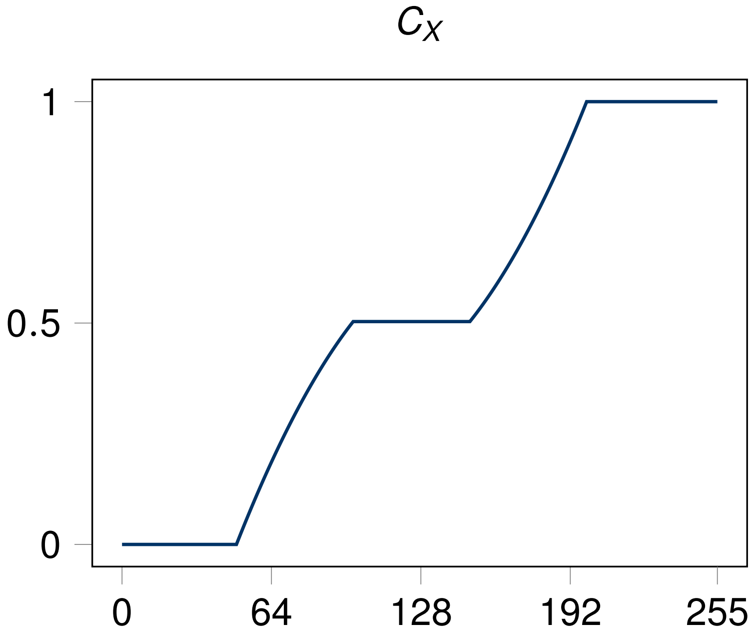




Histogram Equalization (cont.)

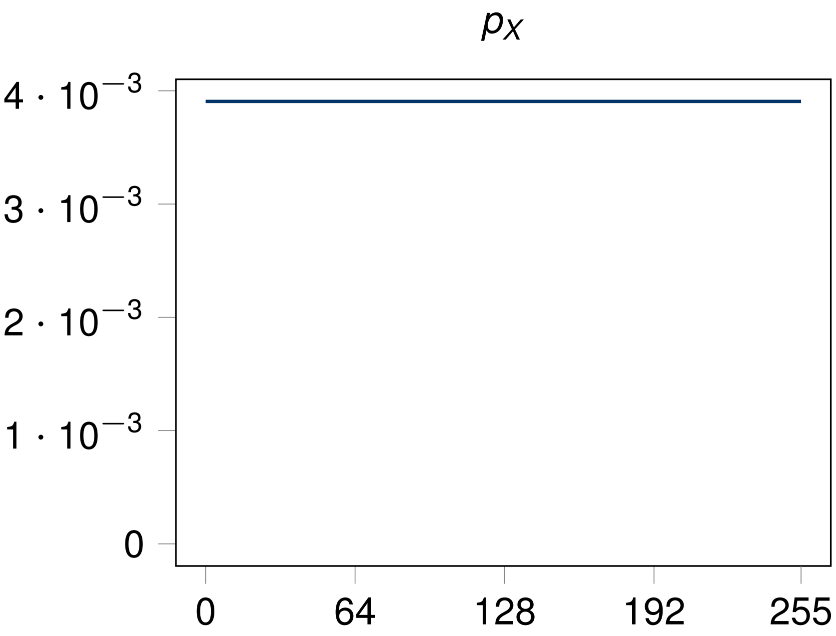
Cumulative distribution: examples

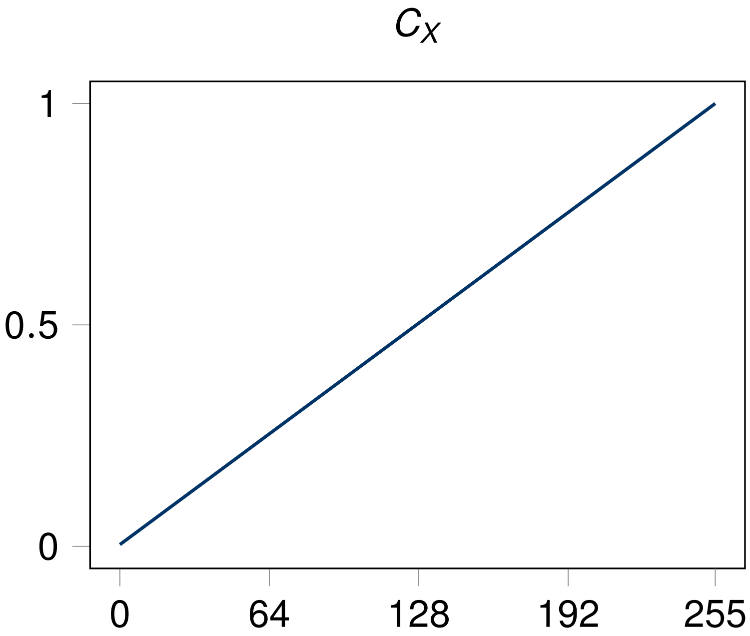




Histogram Equalization (cont.)

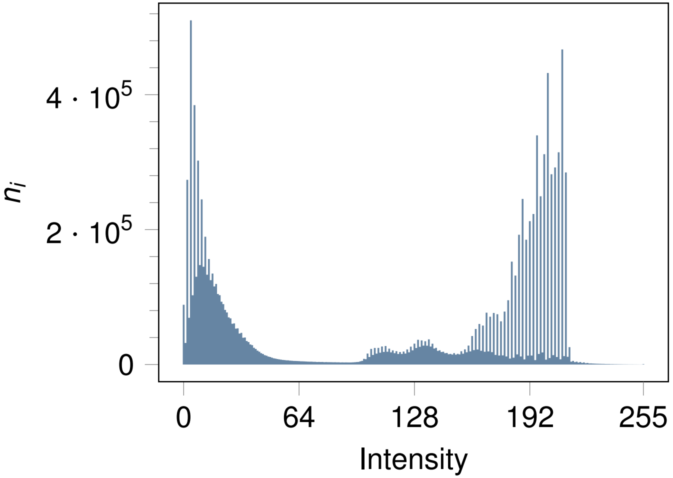
- The transformed intensity values g_i should be $[0, 1]$-uniformly distributed
- This is equivalent to $C_X(x) = x$





Histogram Equalization (cont.)

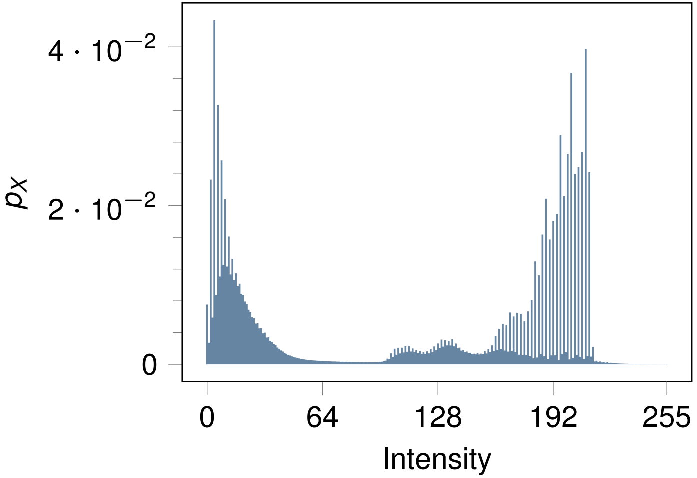
Example



➔

Histogram Equalization (cont.)

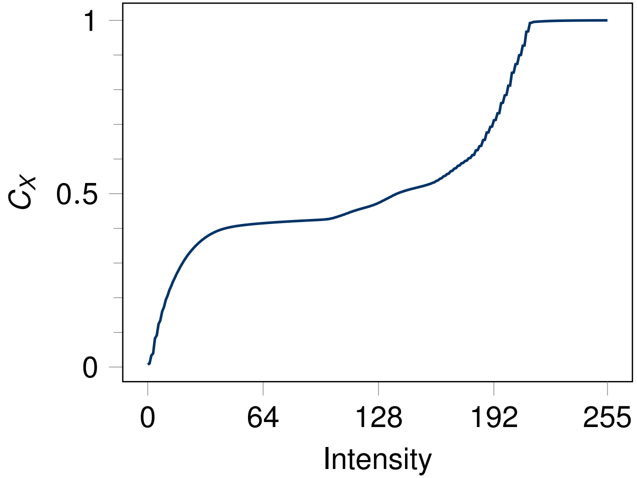
Example



➔

Histogram Equalization (cont.)

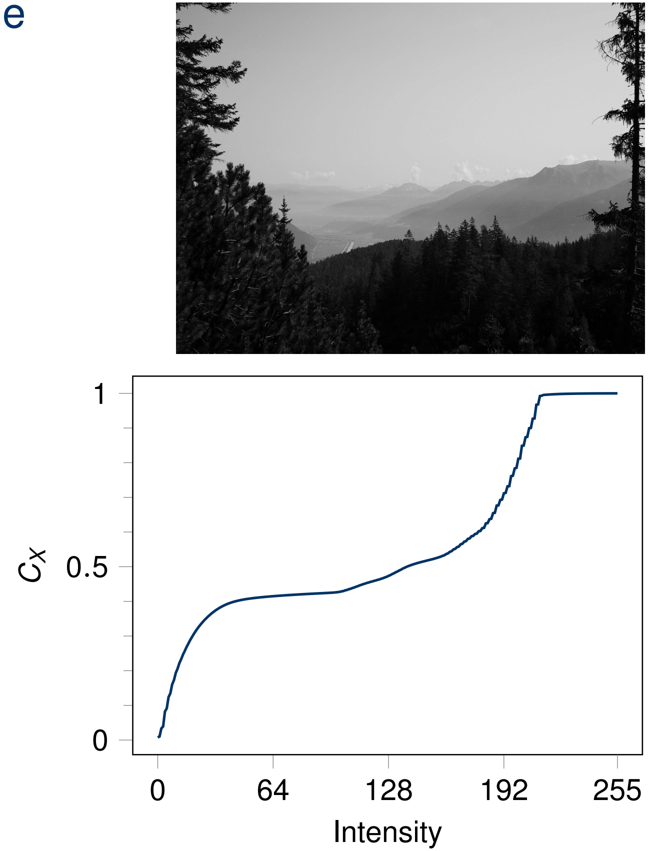
Example



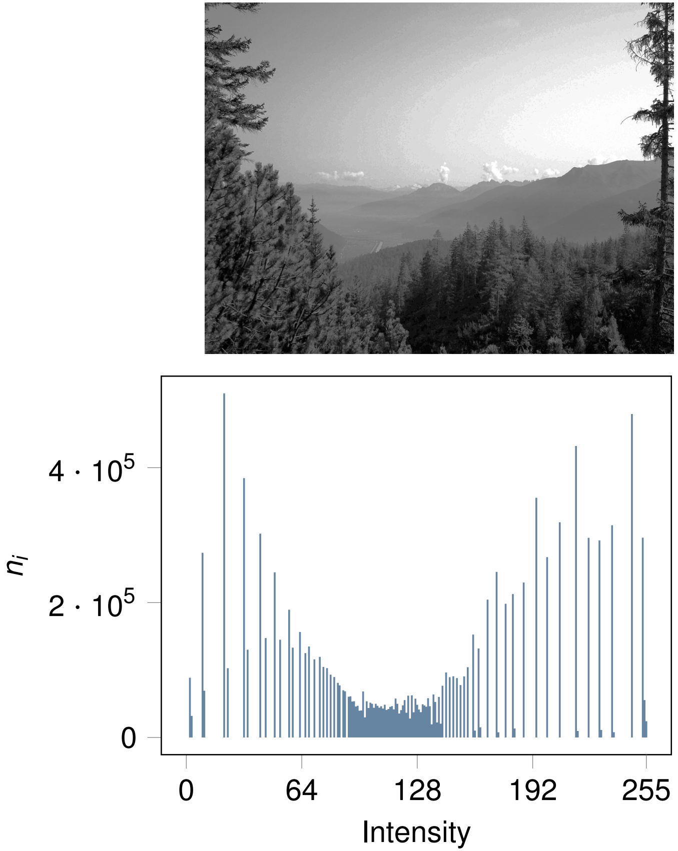
➔

Histogram Equalization (cont.)

Example

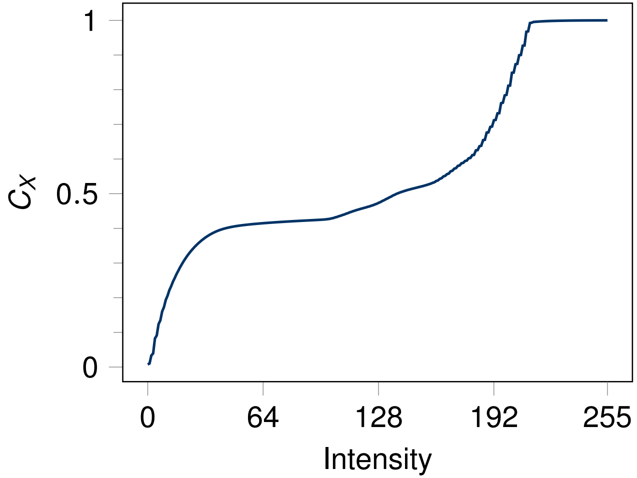


➔

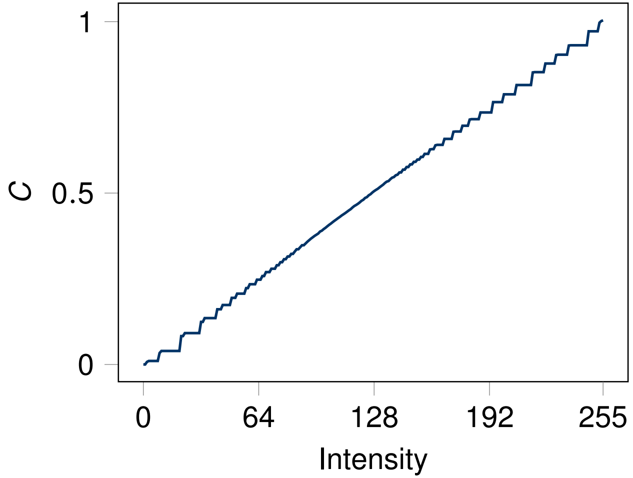


Histogram Equalization (cont.)

Example



→



Histogram Equalization (cont.)

Resulting image:

- Approximately uniform distribution of the intensity values
- Pixels are spread evenly across the entire range of intensity values
- Highest possible contrast

Questions?