Exercise Session 2 - 3

Document Image Analysis

Rolf Ingold, Anna Scius-Bertrand

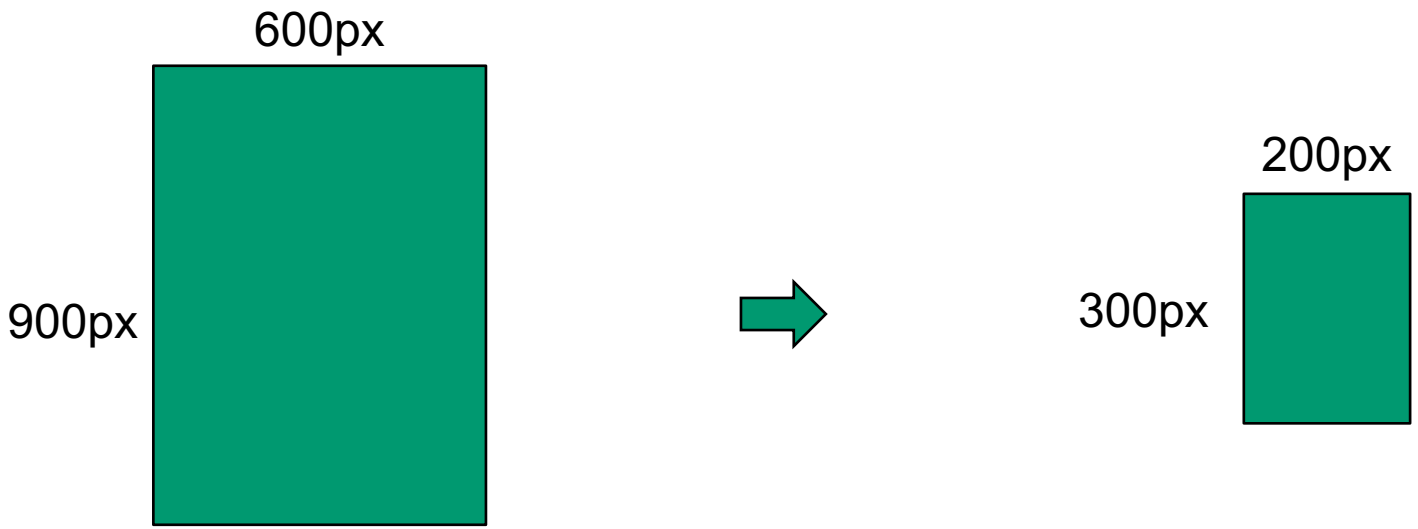
DIVA Group, University of Fribourg, Switzerland

UNI FR

UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

Assignment 1 – Down-scaling Image

- Main part of assignment 1: **Implement your own down-scaling algorithm.**
  - Your algorithm should down-scale the image's height and width by factor 3.
    - e.g., a 600x900 image becomes a 200x300 image.
  - Apply your algorithm to the images on ILIAS.



Feed-back assignment 1

- Library:
  - Pillow, Open cv, Numpy
- Down-scale by 3:
  - divide by 3 -> convert in int
  - Loop over row and colum
    - Take value of one pixel over 3
    - Mean of 3 pixels or a grid 3 by 3
- Bilinear Up-sampling
  - Weighted average of 4 nearest neighbour pixel values
- To improve:
  - Do not duplicate 3 times your code for each image
  - Simplify your code
  - Don’t loop over the channel, always the same value
  - for x in range(0, array_width-2, 3):

Pillow wrap-up

from PIL import Image

- Image.open(img_path) -> open image
- image.size -> (weight, heigh)
- image.getpixel(x, y) -> access pixel value from a x, y position
- image.putpixel((x, y), (R, G, B)) -> add color in an image at position x, y
- image.show() -> show image
- image.save(save_path) -> save image

Solution example

```
new_width1 = im1.width // 3
new_height1 = im1.height // 3
new_image_array1 = np.zeros( shape: (new_height1, new_width1, 3), dtype=np.uint8)
for i in range(new_height1):
    for j in range(new_width1):
        new_image_array1[i, j] = im1.getpixel((j*3, i*3))
new_image1 = Image.fromarray(new_image_array1)
```

```
for i in range(new_image_width):
    for j in range(new_image_height):
        temp = np.zeros(3)
        for x in range(3):
            for y in range(3):
                temp += image.getpixel((i*3+x,j*3+y))
        temp /= 9
        temp = temp.astype(int)
        new_image.putpixel((i,j),tuple(temp))
```

Assignment 2

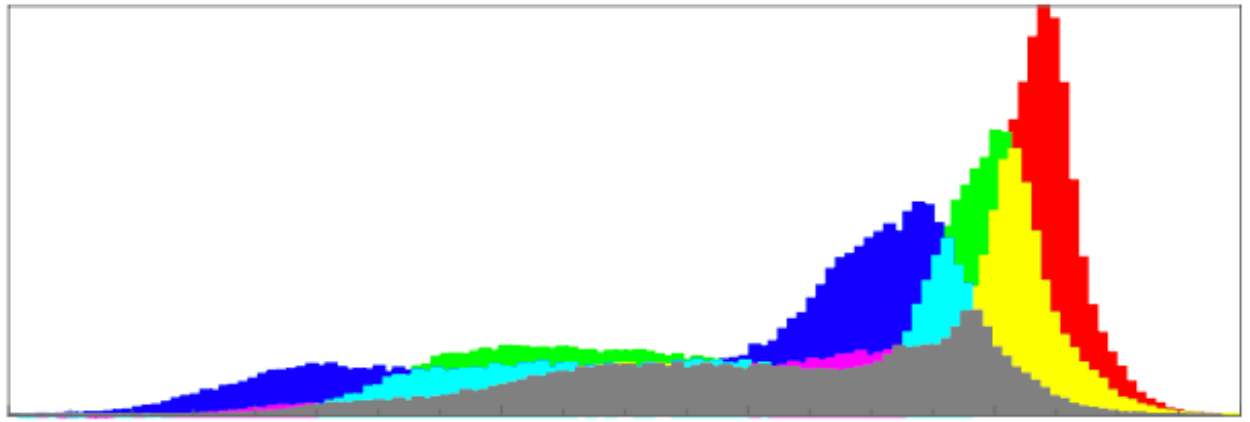
- Deadline: March 12th end of the day

- Objectif:
  - Perform histogram on a grayscale and a RGB image.
  - Realize a representation of a histogram of a RGB image.

- Convert an RGB image in a grayscale image. Use your own implementation.
- Conceive an algorithm to perform a histogram on a grayscale image. Apply it on the image in grayscale.
- Apply your algorithm on each channel of the RGB image.

Assignment 2

- Represent the histogram of a colour image. More information in lesson 2, slide 12. An example is shown in the figure below.



- Submit on ILIAS:
  - Your code (one file should be enough),
  - Five images: the original image, the grayscale image, the histogram of the grayscale image, the 3 histograms of the RGB image, the representation of the histogram of a colour image.
  - A file text with your name, surname and a brief description of your algorithm.

Assignment 3

- Deadline: March 19, end of the day

- Objective: Perform several binarization algorithms on a RGB image of a document and compare the different method.

- Conceive an algorithm to perform a binarization based on a simple non adaptative thresholding.
- Based on a global thresholding, conceive another binarization algorithm.
- Design an algorithm to perform binarization based on local adaptative thresholding.
- Apply your algorithms on an RGB image of a document. Examples are provided in ILIAS.
- Optional: conceive and apply a fourth binarization algorithm of your choice.
- Submit on ILIAS:
  - Your code (one file should be enough).
  - At least four images: the original image, one binary image for each method.
  - A file text with your name, surname and a brief description of your algorithms.

Questions?