

# Computer vision course

## Lab 1 - Intro to OpenCV, pixel manipulation, transformations

### Task 1

Implement the OpenCV Hello world discussed during the lectures - open and visualize an image. Use it to open the grayscale and color images provided.

### Task 2

Edit the software in Task 1 by adding instructions for printing the number of channels of the image opened. Also, save the output of `waitKey` into a char variable and print it before exiting. Check the OpenCV documentation if you do not know how to get the number of channels.

### Task 3

Write a program that creates two images of size 256x256, one channel. Write the pixels using the `at()` function in order to create:

- a vertical gradient in the first image;
- an horizontal gradient in the second image.

Show the images on screen. Expand the software to create and visualize two other images, size 300x300, one channel, with:

- a chessboard with squares of size 20
- a chessboard with squares of size 50

### Task 4

Edit the software in Task 2 by adding a function that checks if the number of channels of the input image is 3. If so, it sets to 0 the first channel and visualizes the image. Which color is missing? Try other versions that set to 0 the second or the third channel. Which color is missing in such cases? What is the color coding used by OpenCV?

### Task 5

Edit the software in Task 2 by adding a function that checks if the number of channels of the input image is 3. If so, it creates three images with the same size of the input image, one channel, containing the values found in the first, second and third channel of the original image. Visualize such images.

### Task 6

Edit the software in Task 1 by creating two more images: a rotated version and an affine transform defined using 3-point correspondences.