Submission before: 02.11.2015
Discussion on: 03.11.2015
Submission on stud.ip, submission folder for sheet.
Please submit a zip file containing the .m files for Matlab programming tasks.

Organizational issues
Practice sheets will be uploaded one week before the practice session. All assignments can and should be prepared and submitted in groups of 3 persons.
Please sign up for a feedback session group on Stud.IP!
You will be admitted to the final exam for the computer vision lecture if you receive at least 50% in $N - 2$ of the sheets.

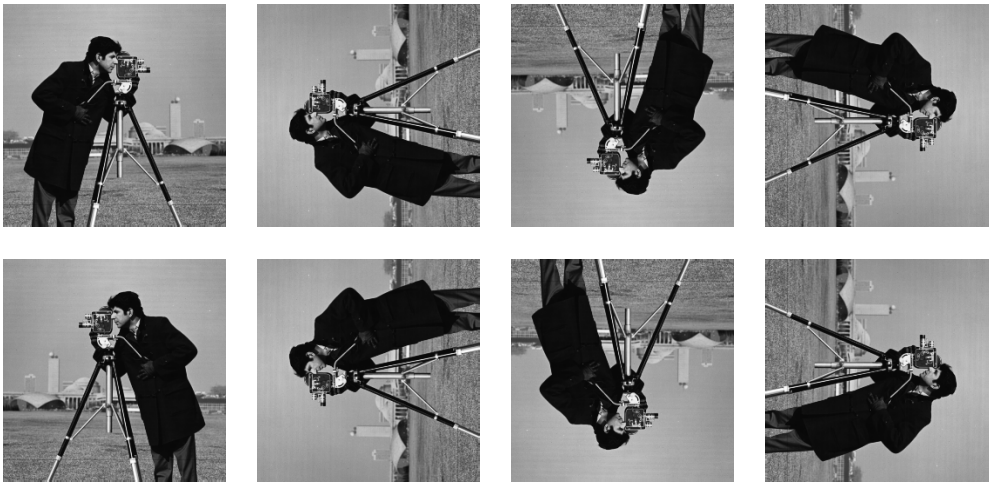
Questions 1 *(Matlab intro/ matrix manipulation - 5p)*

Write Matlab scripts or functions that implement the following tasks (please indicate the tasks in the comments in your code).

(a) Read image 'cameraman.tif' (use imread) and display it (imshow).

(b) Write a function that marks a region of an image with a white rectangle and another function that extracts a rectangular region from an image (only use basic matrix operations, no specialized Matlab functions). Use these functions to mark and extract the head of the cameraman.

(c) Display eight flipped and rotated versions of the image in a subplot. Implement your own functions for mirroring and rotating (again, do not use build-in functions like `flip`, `fliplr`, `flipud`, `rot90` etc. – you may use basic matrix calculation):



Questions 2 *(Noise - 5p)*

(a) Write a function to add Gaussian noise to an image. The function should allow one parameter to specify the variance (you may use `rand` or `randn`, but not `imnoise`). Demonstrate the function with an image of your choice.

(b) Write a function to add impulse (salt-and-pepper) noise to your image. Allow a parameter to specify the percentage of distorted pixels.

(c) Temporal smoothing: create a series of n distorted images ($n = 2, 4$, or 8, either Gaussian or impulse noise) and apply the technique of "temporal smoothing" from the lecture.

Questions 3 *(Point operators - 5p)*

(a) Read image 'coins.png'. Calculate an inverted version of the image (inverted: high grayscale values become small grayscale values, small grayscale values become large grayscale values).

(b) Plot a histogram of the initial grayscale image using the Matlab function `histogram`. Then write your own function to display a histogram (you may use the function `bar`). Hint: use the `help` command to learn more about these functions.

(c) Binarization: find a threshold value to separate foreground from background. Show the resulting binary mask.

(d) Use the mask to display only the foreground items.

Questions 4 *(Dyadic operators/video processing - 5p)*

For this exercise you need access to a camera (e.g. webcam, camera of your smartphone, etc.).

(a) Acquire a single image from your camera and display it. Then write a loop that continuously acquires and displays images.

(b) Motion detection: use the method of "difference images for motion detection" from the lecture to show moving objects in your video stream. Experiment with different parameter settings.

(c) Background subtraction: first acquire a static background image for reference. Then start a videostream in which only foreground objects (i.e. objects not present in the background image) are shown, i.e. remove the background from your video.

Hint: for this exercise it suffices to work with grayscale images. If your camera provides color images (RGB), you can convert them by calling `rgb2gray`.