COMP9517

Lab 1, S2 2018

This lab presents a revision of important concepts from week 1 and 2 lectures, and short questions around them. Most questions require you to use OpenCV, an open source software package that is widely used in the field. OpenCV v2.9.4 is installed on the lab machines and accessible through vlab.

A file containing sample images is available in the link Data Samples.

All questions should be attempted during the lab hour.

The last question (Question 6) is assessable IN THE LAB. Make sure to show your answer to your tutor before leaving the lab. It will NOT be assessed later on outside the lab.

QUESTIONS SNAPSHOT

Use this snapshot to plan your time. The detailed question, along with a revision of concepts and hints to solve the question, appear later.

All questions must be attempted during the lab hour, and ***make sure that you complete question 6*** and show your answer to one of the tutors available in the lab.

1. Let (x, y, z) be the point P in world cords, and (u,v) the corresponding point on the image plane. For this question, refer to the slides for more information.

   (i) Under perspective projection, show that the values of u and v derived geometrically (Image Formation slide 27) are the same when derived using the projection matrix (Image Formation slide 35).

   (ii) Do the same for orthographic projection (slides 29 and 36) and scaled orthographic or weak perspective projection (slides 28 and 37).
2. Read two images, image1 and image2, and save a new image image3 by combining image1 and image2.
3. Read a grey scale image and modify it to 32, 8, and 2 grey level images by using 5, 3 and 1 quantization bars respectively.
4. Read a grey scale image and find its negative.
5. Read a grey scale image and perform contrast stretching to improve the visual quality of the image.
6. Write a function that computes the histogram of a grey scale image and displays a plot. Do not use in built functions. Display the histogram for a sample image on the screen. The answer to this question will be assessed.

I LOADING, SAVING AND DISPLAYING IMAGES

Note: If you are familiar with MATLAB already, the following table may be of help. The function names in MATLAB and OpenCV are quite similar but their syntax is different.

The table also provides basic code snippets to perform the operations. For more information on the syntax, you should refer to the documentation.

MATLAB	OpenCV
Loading/Reading an image	
image1 = imread('circlesBrightDark.png');	import cv2 image2=imread('threads.png',0)
Saving/Writing an image	
imwrite(image3,'C:\Users\ABCD\image3.png');	Cv2.imwrite('C:\Users\ABCD\image3.png',image3)
Displaying an image	
imshow(image3);	Cv2.imshow('Title of the display window',image3)

QUESTION 2: Read two images image1 and image2 and save a new image image3 by combining image1 and image2 as shown in the figure below. Make sure that the two images are of the same size, and if they are not, crop the larger image.

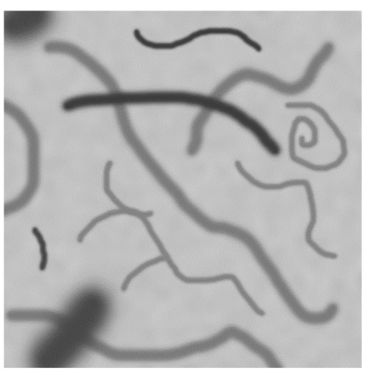


image1

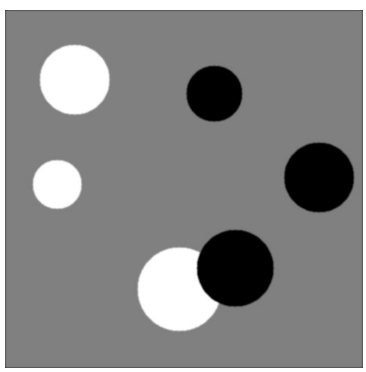


image2

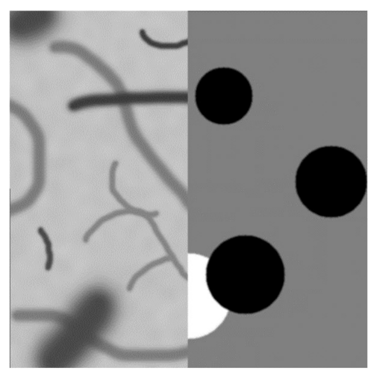


image3

II IMAGE QUANTIZATION

A digital image is created by converting continuous sensed data into digital form. It requires two basic steps: sampling and quantization. Focus on digitizing the amplitude values, also called as quantization. If an image is quantized using 5 different levels of grey values, the image may contain 5 different pixel values, ranging from 0 (black) to 4 (white). The image quality can be improved by increasing the number of levels- if an image is quantized into 256 different grey levels, the grey level resolution increases. In such an image, each pixel needs storage space of 8 pixels ($2^8 = 256$).

256 Grey levels 32 Grey levels 8 Grey levels 2 Grey levels

QUESTION 3: Read a grey scale image and modify it to 32, 8, and 2 grey level images by using 5, 3 and 1 quantization levels respectively, as shown in the figure above. Note the deterioration in the image quality as the number of grey levels decreases.

HINT: To convert a 256-grey scale image to an 8-grey scale image, $= original\ pixel\ value * \frac{7}{255}$. While displaying the image make sure to set the correct colour map.

III IMAGE PREPROCESSING (Image Enhancement)

Sometime captured images may have poor illumination or wrong setting of lens aperture which leads to images of low quality. The pre-processing of such images is an integral step for the success of any computer vision algorithm. The following figure from the book Computer Vision Metrics [1], gives an insight into the importance of pre-processing. The contrast adjustments made to the image have improved the performance of the Sobel edge detector.

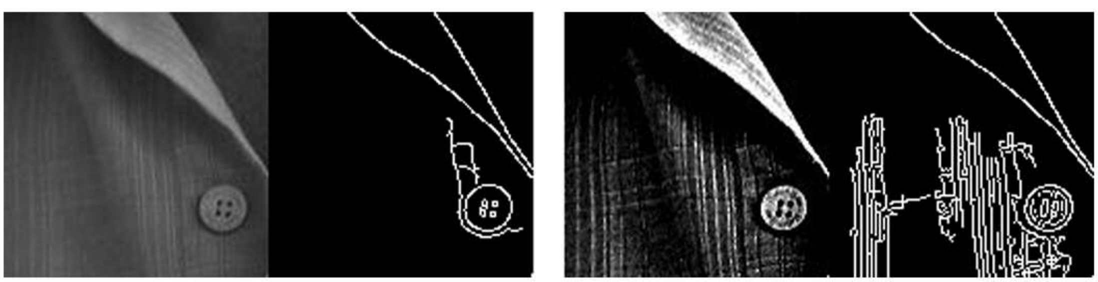


Figure 2-2. *The effects of local contrast on gradients and edge detection: (Left) Original image and Sobel edges. (Right) Contrasted adjusted image to amplify local region details and resulting Sobel edges*

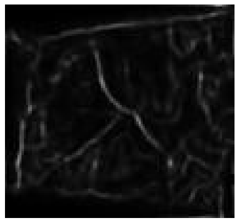
These processes can be broadly divided into two categories: spatial domain methods and frequency domain methods. In this lab session we will be dealing with spatial domain methods.

Grey Level Transformations

This is the simplest and basic image enhancement technique. Assume that ***Or*** is the original image and ***Tr*** is the transformed image.

Image Negative: $Tr = L - 1 - Or$, for an image with grey level in the range 0 to L-1. This enhancement makes the analysis of the image easier although the content of the two images are the same.

QUESTION 4: Read a grey scale image and find its negative, a sample is shown in the figure below:


Original Image

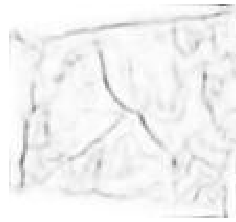


Image Negative

Contrast Stretching: Contrast in an image is a measure of the range of intensity values within an image, and is the difference between the maximum and minimum pixel values. The full contrast of an 8-bit image is 255(max)-0(min)=255, and anything less than that results in a lower contrast image. Contrast stretching attempts to improve the contrast of an image by stretching (linear scaling) the range of intensity values. Let a and b are the min and max pixel values allowed in an image (8-bit image, a=0 and b=255), and let c and d be the min and max pixel values in a given image, then the contrast stretched image is given by the function:

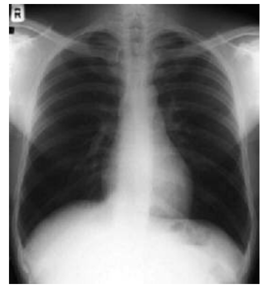
$$Tr = (Or - c)\left(\frac{b - a}{d - c}\right) + a$$

Notes:

- If there are spurious pixels with very high or very low values, they seriously affect the values of c and d, so it is advisable to choose c and d in such a way that 5% of the pixels will have a value less than c and 5% of pixels will have a value more than d respectively.
- Values of a and b can be chosen depending on the amount of contrast that has to be obtained for full contrast on an 8-bit image 0 and 255; for lesser contrast, modify a and b appropriately.

QUESTION 5: Read a grey scale image and perform contrast stretching to improve the quality of the image. Shown below is a poor contrast X ray image (Picture from [2]) and its contrast stretched version.

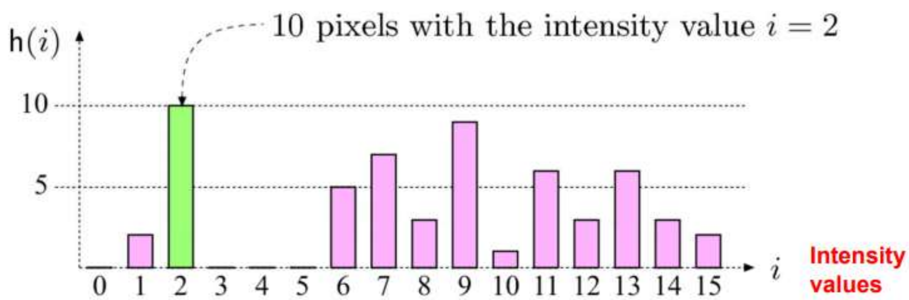
Original Image



Contrast Stretched image

Histogram

Histogram of an image shows the frequency of pixel intensity values. It only gives statistical information and nothing about the location of the pixels. For a digital image with grey levels from 0 to L-1, the histogram is a discrete function $h(Or_k) = n_k$, where $\boldsymbol{Or_k}$ is the kth grey level and $\boldsymbol{n_k}$ is the number of pixels with a grey level $\boldsymbol{r_k}$.



i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
h(i)	0	2	10	0	0	0	5	7	3	9	1	6	3	6	3	2

Figure 1: Histogram (Picture from [3]).

QUESTION 6: Write a function that computes the histogram of a given grey scale image and displays a plot. Do not use in built MATLAB or OpenCV functions.

REFERENCES

[1]. Krig S. (2014) Image Pre-Processing. In: Computer Vision Metrics. Apress, Berkeley, CA, https://link.springer.com/chapter/10.1007/978-1-4302-5930-5_2#citeas

[2]. http://cursa.ihmc.us/rid=1GJRS5FYJ-HBJGJG-1FF0/Cindy%20and%20Melonie's%20CMAP%20Digital%20Imaging%20Processing.cmap.cmap

[3]. http://web.cs.wpi.edu/~emmanuel/courses/cs545/S14/slides/lecture02.pdf

[4]. http://www.cs.uu.nl/docs/vakken/gr/2011/Slides/07-projection.pdf

[5]. https://cg.informatik.uni-freiburg.de/course_notes/graphics_03_projections.pdf

[6]. https://www.scratchapixel.com/lessons/3d-basic-rendering/perspective-and-orthographic-projection-matrix/projection-matrix-introduction