

The Hong Kong University of Science and Technology
Department of Computer Science and Engineering
COMP4421 (Fall 2019)

Assignment 3

Total = 100 marks

Due: 11:55pm, Dec 9, 2019

Assignments must be submitted via CANVAS

Late Policy: 10% reduction; only one day late is allowed, i.e. 11:55pm, Dec 10

Overview

This assignment consists of two sections: programming section and written section. Both programming and written parts should be submitted via the CANVAS system. If you would like to finish the written assignment with hand writing, you may scan and upload it as a PDF file.

Programming assignment specifics (60%)

Pre-requirement

2.1.1 Input: The input images are in the folder “input_imgs” (/Sample/input_imgs). You should test all the input images and paste the result images in the PDF file. Note that we will have some other test images for grading.

2.1.2 Language: Matlab/Octave.

2.1.3 Functions to be used: In this assignment, you can use the built-in function for filtering and edge detection. However, **you cannot use any built-in functions relating to the image warping and segmentation.**

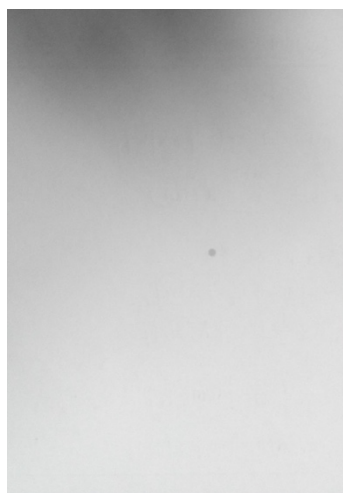
Programming question 1 - Image Warping (40%)

Based on the 4 corner points computed in Assignment 2, apply the image warping to transform the target A4 paper into a standard A4 form, that is, $210n * 297n$ for vertical direction and $297n * 210n$ for horizontal direction, where n is the parameter decided by the user. Inside your own image warping function, you also need implement your own bilinear interpolation. Please describe your image warping algorithm briefly in the PDF file.

The function prototype is given in the image_warping.m.

If you cannot obtain the 4 corner points in Assignment 2, you can manually find the 4 corner points using some tools in Matlab/Octave.

A sample output is provided as follows,

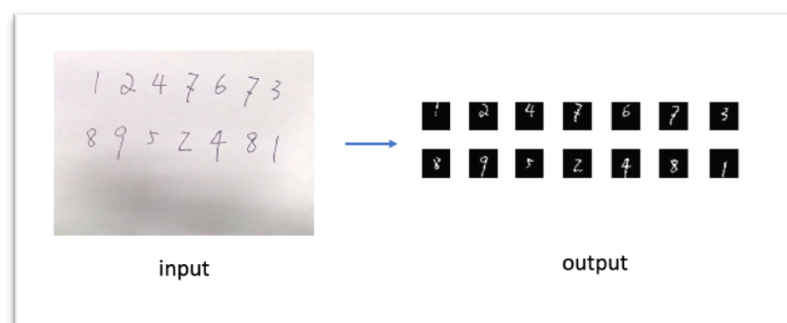


Programming question 2 - Digit Segmentation (20%)

Followed by the Task 2.1, our input images in this task are the transformed A4 paper with some hand-written digits on it. Your task here is to use a certain segmentation algorithm to segment each digit. Note that you can choose any segmentation methods. **However, no matter which algorithm you choose to use, you should implement it by yourself.** Please describe your segmentation algorithm briefly in the PDF file.

The function prototype is given in the digit_segment.m.

A sample output is provided as follows,



Here, I provide one of the algorithms I used in my program for the digit segmentation.

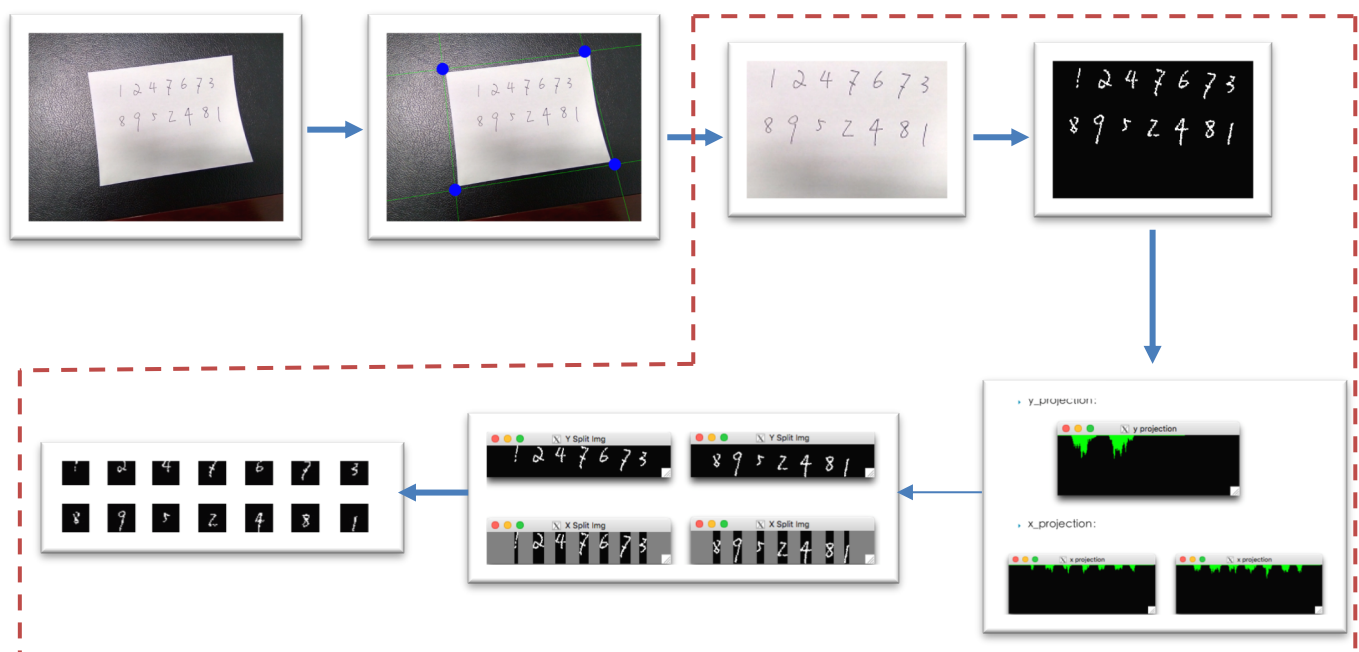
Step 1: Threshold the input images to obtain binary images;

Step 2: Horizontal projection. Count the number of white pixels along horizontal direction. The number of rows for the digits should be equal to the number of peaks in the projected histogram. We can then segment the digits by rows.

Step 3: Vertical projection. For each row, we count the number of white pixels along vertical direction. The number of columns for the digits should be equal to the number of peaks in the projected histogram. We can then segment the digits by columns.

[Note: for step 3 you may face the problem that some digits cannot be separated properly. Here is another idea for Step 3. Scan each column and count the number of black pixels, denoted as N . If $N > H - n$, where H is the height of the row_image obtained in Step 2 and n is a parameter defined by user, the column will be marked. The segmentation will be performed based on the marked column. This idea, which is similar to vertical projection, compares the height of the row_image and thus is easier to control.]

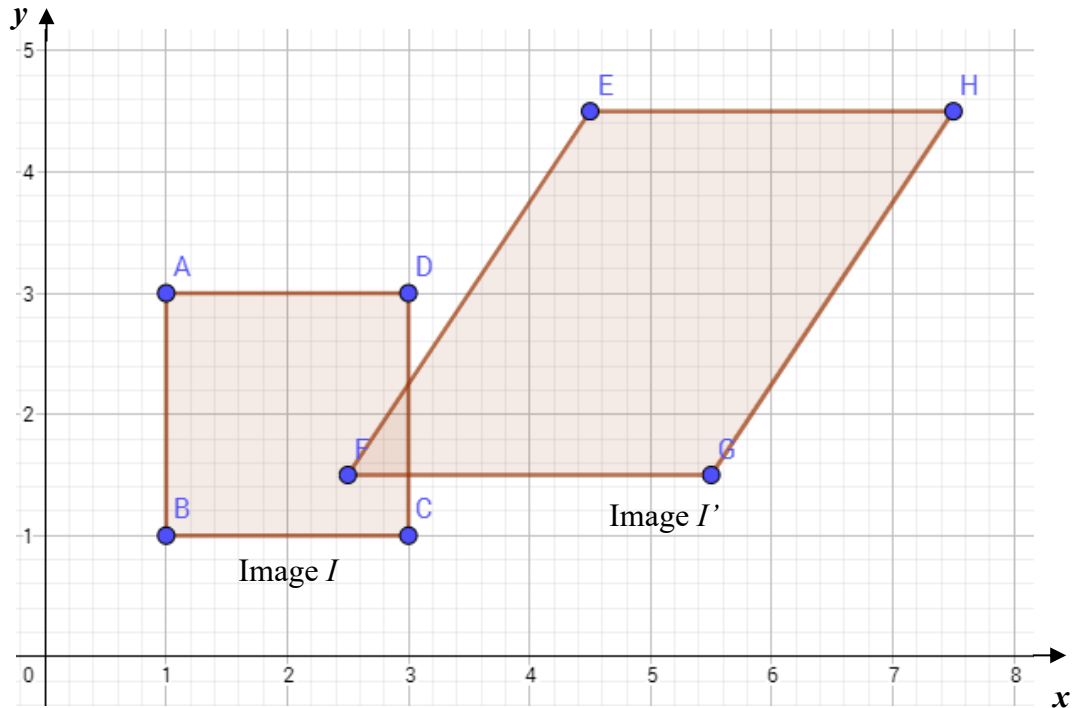
I provide the pipeline combining Assignment 2 and Assignment 3 as follows. Note that only the parts in the orange dash box are required in this assignment.



Written assignment specifics (40%)

Written question 1 - Image Transformation

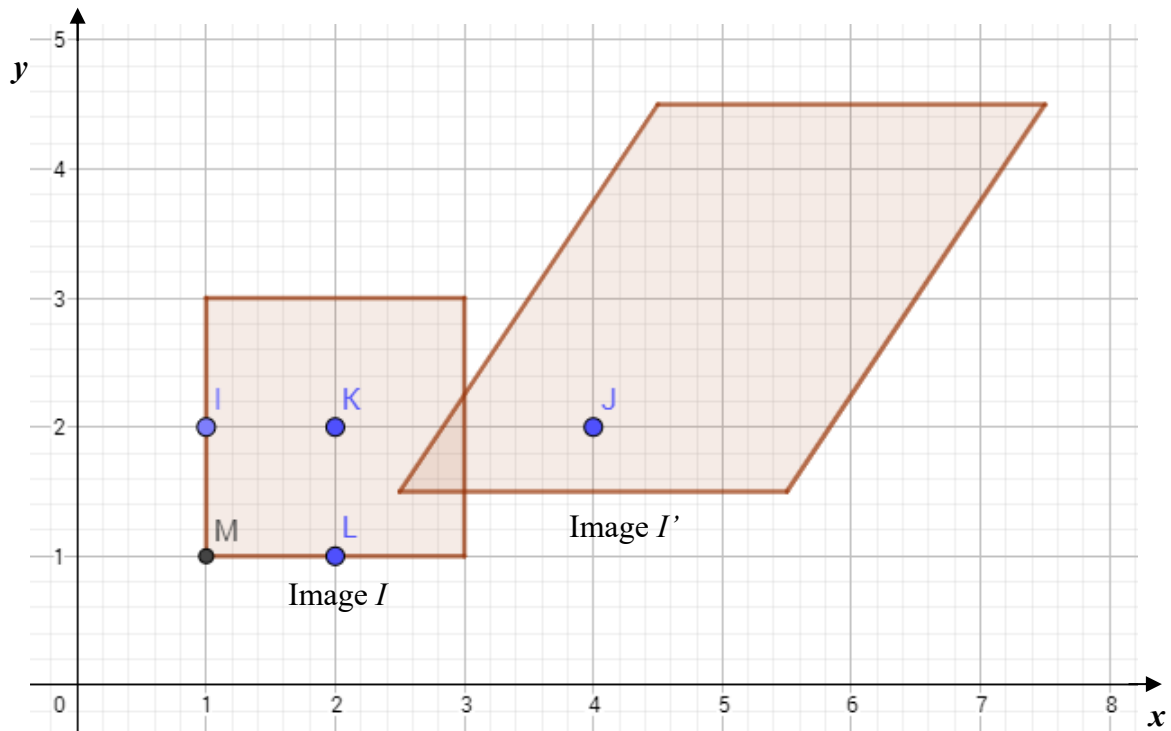
There is an image I shown as follows. The transformation is defined as $T(\vec{x}) \equiv \vec{S}\vec{x}$, where $\vec{S} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ and $\vec{x} = \begin{pmatrix} x \\ y \end{pmatrix}$. After the transformation using T , point A, B, C and D are transformed to point E, F, G and H respectively.



(Hint: For a 2x2 matrix $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$, the matrix inverse is $A^{-1} = \frac{1}{ad-bc} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$.)

1a) Find the transformation matrix \vec{S} . Please show all your steps clearly.

Suppose the intensity values of point I, K, L and M are 2, 5, 9 and 15 respectively.



1b) Find the intensity value of point J using Bilinear Interpolation. Please show all your steps clearly.

Written question 2 - Quantization

Given a 4x4 gray-level image A as follows:

13	115	24	231
128	200	64	55
55	255	74	35
178	29	1	80

2a) Compute the quantization (uniform quantization) of A from 256 levels (8 bits) to 16 levels (4 bits).

2b) Compute the column-wise improved gray-scale (IGS) quantization of A from 256 levels (8 bits) to 16 levels (4 bits).

2c) Find the root-mean-square error and mean square signal-to-noise ratio of the result from your answers in 2a) and 2b).

Written question 3 – Huffman coding

Given Huffman code in the below table.

<u>Symbol</u>	<u>Code</u>
C	01
E	10
M	00
O	110
P	1111
R	11101
T	111000
U	111001

3a) Decode the encoded string “011100011111110011110001011101” with the given Huffman code.

3b) Compute the Huffman code for each symbol in the words “image processing” and fill in the following table. Show your steps clearly.

<u>Symbol</u>	<u>Code</u>
a	
c	
e	
g	
i	
m	
n	
o	
p	
r	
s	