Text Image Segmentation for Optimal Optical character recognition

Optical character recognition (OCR) aims to recognize texts in imaged documents. It is probably one of the earliest computer vision techniques that have been commercialized successfully. OCR usually involves a series of image processing and recognition tasks including 1) text image binarization that converts a colour/grayscale image into a binary image with multiple foreground regions (usually characters); 2) connected component labelling that detects each binarized character region; 3) character recognition by using some classifiers such as a pre-trained neural network.

In this project, you need to explore and develop various image binarization algorithms targeting the optimal character recognition accuracy. You do not need to develop OCR algorithm but use an open-source OCR software Tesseract: https://github.com/tesseract-ocr/tesseract. You can google the keyword "Tesseract" to search for more information about how to use this software. In addition, you just need to work on either one (or both) of the two sample text images that can be downloaded from the project website in NTULearn.

This project consists of the following tasks:

1) Implement the Ostu global thresholding algorithm for binarizing the sample text images and feed the binairzed images to the OCR software to evaluate the OCR accuracy. Discuss any problems with the Otsu global thresholding algorithm.
2) Design your own algorithms to address the problem of Otsu global thresholding algorithm, and evaluate OCR accuracy for the binary images as produced by your algorithms. You may explore different approaches such as adaptive thresholding, image enhancement, etc., and the target is to achieve the best OCR accuracy.
3) Discuss how to improve recognition algorithms for more robust and accurate character recognition while document images suffer from different types of image degradation. This is an open and optional task. There will be bonus points if you have good ideas on it.

You need to submit your project report in PDF format, and you can append your source code at the end of the project report. There is no standard report template, and you can choose your favourite programming languages such as Matlab, Python, Java, etc.

If you decide to do the project in groups, the maximum group size is two and please indicate the group members clearly in the report cover page. Please also indicate the contribution of each group member at the beginning of your project report.

I will evaluate the report from two major aspects, namely, contents and presentation. For contents, I expect you develop some good image binarization algorithms that demonstrate improved OCR accuracy. For presentation, I will look more at clarity, logical flow, elegance, etc.

Please submit your report through NTULearn by the end of 20 Nov 2020. There will be penalty for late submissions.