University of Lincoln Assessment Framework
Assessment Briefing 2020-2021

NOTE: All Assessment Briefings should be made available prior to the commencement of the module, clearly signposted on the module Blackboard site as well as included in any module handbook or briefing document.

Module Code & Title: CMP3108M Image Processing

Contribution to Final Module Mark: 100%

Description of Assessment Task and Purpose:

Requirements

This assessment comprises two assessed components, as detailed in the following page.

1. A report (in PDF format) that describes your approach to the tasks (maximum 4 pages, including figures but not the cover page). Weighting: 30% of this assessment.
2. A file containing all functions written in MATLAB/Octave code with clear comments and requested figures. Weighting: 70% of this assessment.

Tasks

Download and unzip the file 'Assignment_Input.zip' from Blackboard. You should obtain:

- A dataset of 10 images containing screws and washers.
- Two MATLAB script m-files named 'Task1to4.m' and 'Task5to6.m'.
- A folder containing the ground truth images.

Complete the m-files to perform the corresponding tasks described below. As a guide, a few lines for performing the tasks have already been added to the script. You need to add the lines to implement the other steps. Ensure you add appropriate comments to your code to briefly explain what each section is doing. You CAN use any built-in function but not any custom functions written by others (e.g. from Matlab File Exchange).

The assessment is about solving an object recognition problem through image processing. The circumstance generalises to an industrial setting, like analysing objects on a conveyor belt. For this purpose we will have a camera mounted straight above the incoming objects. The objects are photographed against a neutral background (white) clearly distinguishable from the objects themselves (metallic grey). In the dataset we have pictures of a mix of fasteners, with the target of identifying them automatically.

Task1 – Pre-processing (15%)

Add code to the MATLAB script Task1to4.m to load the image 'IMG_01.jpg' and convert it to grey-scale. Then reduce the image size from its original size to half by bilinear interpolation. With the re-sized image, produce its histogram. Enhance your image to facilitate binarisation of the image (e.g. contrast adjustment). Produce another histogram based on the enhanced image. Next, binarise the image so that the screws and washers can be detected. Display the re-sized image, histograms before and after enhancement, enhanced image and the binarised image to the screen and record them in your report. Discuss and justify the enhancement strategy you adopted.

Task2 – Edge Detection (10%)

Continue to write your code in the MATLAB script Task1to4.m to apply edge detection techniques to the enhanced image you produced in Task 1. The target is to get the edges of the objects in the image. Display the detected results and explain which edge detection algorithm you have chosen and why.

Task3 – Simple Segmentation (15%)

Continue to write your code in the MATLAB script Task1to4.m to automatically segment the objects using any of the resulted images obtained from Task1 and Task2. Write down the steps you took in the report and include the resulted image(s). Note the segmented image should be binary and only contain the screws and washers (as accurate as possible) on a black background. An example of the segmentation is shown below (Figure 1).

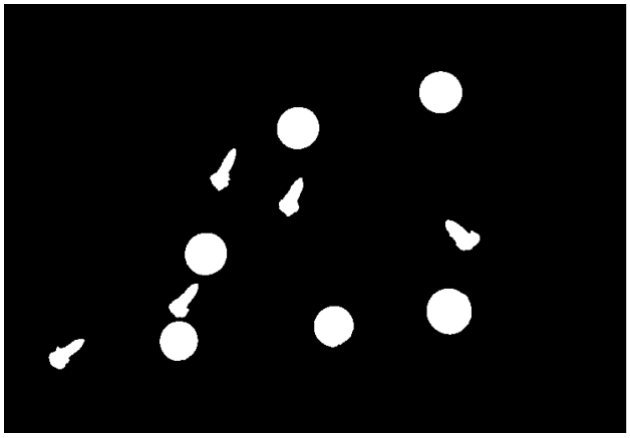

Figure 1, example of resulted segmentation from task 3.

Task4 – Object Recognition (10%)

Complete the script Task1to4.m and use appropriate feature(s) to differentiate small screws to washers in 'IMG_01' from your segmented results obtained in Task3. Label the objects appropriately with different colour (e.g all small screws blue and all washers red). Output your results to the screen and write down your solution in the report. An example result is shown in Figure 2.

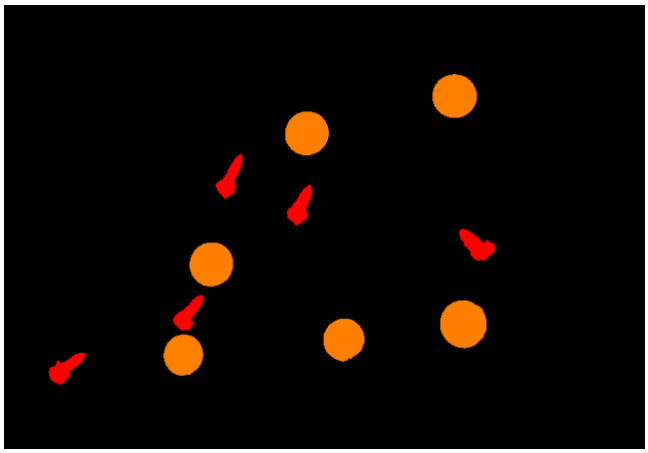

Figure 2, example of resulting recognition from task 4.

Task5 – Robust Method (30%)

You are likely to find that the script you've written so far will not work if you change 'IMG_01.jpg' to another image from the provided dataset. Write code in the MATLAB script Task5to6.m to recognise and segment the objects through a series of image processing techniques you choose. From the dataset, you can see that we actually have three types of objects. On top of the short screws and washers you've encountered in 'IMG_01', there are also some long screws in some of the other images. Therefore, your ultimate goal is to write an automatic and robust method that is able to accurately segment and recognise the three types of objects (i.e. long screws, short screws and washers) from all provided images.

Your solution must minimise the amount of hard thresholds in order to make the algorithm as

robust as you can. It should NOT be image specific (i.e. a set of parameter values or thresholds for each image). It should NOT involve any training (i.e. a machine learning based approach).

In your report, explain each step you have taken and why you have used it. Select one particular image other than 'IMG_01.jpg', and illustrate the outcome of each processing stage by adding example figure(s) to your report.

For Task5, you can optionally save all the resulted images when running the algorithm on the entire dataset in a file called 'output'. This output file can be zipped up together with your code and submitted through 'supporting material upload'.

Task6 – Performance Evaluation (20%)

For this task, we have provided you with the ground truth images (contained in the Assignment_GT.zip file). The ground truth images are of the same size as the dataset image after being rescaled by Task 1. They are basically grayscale images correspond to images in the input dataset with only three labelling values. The washers are labelled with 1, small screws value 2, big screws value 3 and background value 0. The final task would be to evaluate your method by reporting its performance on the entire dataset against the provided ground truth using Dice Score, Precision and Recall. Continue to write your code for this task in script Task5to6.m, output your evaluation results to the screen and record it in report as well.

Learning Outcomes Assessed:

On successful completion of this component the student will have demonstrated competence in the following areas:

- [LO1] critique the theoretical knowledge of image processing, including how to process and extract quantifiable information from images.
- [LO2] apply image processing techniques to solve practical problems.

Knowledge & Skills Assessed:

Subject Specific Knowledge, Skills and Understanding: academic report writing, literature searching, referencing, mathematics, algorithm development, project planning, designing and management, image processing techniques including image formation, intensity transformation, spatial filtering, morphological operations, image segmentation, feature representation and description and recognition evaluation, etc.

Professional Graduate Skills: independence and personal responsibility, adaptability, verbal communication, written communication, creativity, critical thinking, IT skills, problem solving, research skills, effective time management, working under pressure to meet deadlines.

Emotional Intelligence: self-awareness, self-management, motivation, resilience, self-confidence.

Career-focused Skills: An understanding of the range of skills and attributes required by employers from the imaging field, a range of strategies to present skills and attributes to employers.

Assessment Submission Instructions:

The deadline for submission of this work is included in the School Submission dates on Blackboard. You must make an electronic submission of your report in PDF format together with a zip file containing all source code files (i.e. two m-files which produce the desired results and display the outputs) by using the assessment link on Blackboard for this component. Make sure the MATLAB scripts are correct and functional and do not display any error message. Put all the files (excluding the provided images and compress the folder into a zip file for submission. Name your zip file and PDF report using this format: LastName_ FirstName_StudentNo. The report should be submitted through TurnItIn and the zip file should be uploaded as supporting material. You must attend the lectures for further details, guidance and clarifications regarding these instructions.

Date for Return of Feedback:
15 working days after submission deadline. Please refer to the hand-in spreadsheet on Blackboard for feedback date.

Format for Assessment:
See CRG. Marks allocation for each task is distributed as below:
Task 1 (15%), Task 2 (10%), Task 3 (15%), Task 4 (10%), Task 5 (30%) and Task 6 (20%).
The weighting between report and code is 30% and 70%, respectively.

Feedback Format:
Written feedback via blackboard. Face to face feedback can be given upon request from student.

Additional Information for Completion of Assessment:
This assessment is an individually assessed component. Your work must be presented according to the School of Computer Science guidelines for the presentation of assessed written work. Please make sure you have a clear understanding of the grading principles for this component as detailed in the accompanying Criterion Reference Grid. If you are unsure about any aspect of this assessment component, please seek the advice of a member of the delivery team.

Important Information on Dishonesty & Plagiarism:
University of Lincoln Regulations define plagiarism as 'the passing off of another person's thoughts, ideas, writings or images as one's own...Examples of plagiarism include the unacknowledged use of another person's material whether in original or summary form. Plagiarism also includes the copying of another student's work'.

Collusion is defined as when a student submits work for assessment done in collaboration with another person as entirely their own work or collaborates with another student to complete work which is submitted as that other student's work. Collusion does not apply in the case of the submission of group projects, or assessments that are intended to be produced collaboratively.

Plagiarism and collusion is a serious offence and is treated by the University as a form of academic dishonesty. Students are directed to the University Regulations for details of the procedures and penalties involved.

For further information, see www.plagiarism.org