GROUP ASSIGNMENT COVER SHEET

Student ID Number	Surname	Given Names
32457375	Soo	Yong Qi
31989632	Tuan Muhammad	Zafri
31861393	Ong	Yi See

***** Please include the names of all other group members.

Unit name and code	Image Processing, FIT3081		
Title of assignment	Assignment 2		
Lecturer/tutor	Dr.Raveendran Paramesran		
Tutorial day and time	Tuesday 12pm	**Campus**	Monash University Malaysia
Is this an authorised group assignment?	☑ **Yes** ☐ **No**		
Has any part of this assignment been previously submitted as part of another unit/course?	☐ **Yes** ☑ **No**		
Due Date	2/5/2023	**Date submitted**	2/5/2023

All work must be submitted by the due date. If an extension of work is granted this must be specified with the signature of the lecturer/tutor.

Extension granted until (date) .... Signature of lecturer/tutor ....

Please note that it is your responsibility to retain copies of your assessments.

Intentional plagiarism or collusion amounts to cheating under Part 7 of the Monash University (Council) Regulations

Plagiarism: Plagiarism means taking and using another person's ideas or manner of expressing them and passing them off as one's own. For example, by failing to give appropriate acknowledgement. The material used can be from any source (staff, students or the internet, published and unpublished works).

Collusion: Collusion means unauthorised collaboration with another person on assessable written, oral or practical work and includes paying another person to complete all or part of the work.

Where there are reasonable grounds for believing that intentional plagiarism or collusion has occurred, this will be reported to the Associate Dean (Education) or delegate, who may disallow the work concerned by prohibiting assessment or refer the matter to the Faculty Discipline Panel for a hearing.

Student Statement:

- I have read the university's Student Academic Integrity Policy and Procedures.
- I understand the consequences of engaging in plagiarism and collusion as described in Part 7 of the Monash University (Council) Regulations http://adm.monash.edu/legal/legislation/statutes
- I have taken proper care to safeguard this work and made all reasonable efforts to ensure it could not be copied.
- No part of this assignment has been previously submitted as part of another unit/course.
- I acknowledge and agree that the assessor of this assignment may for the purposes of assessment, reproduce the assignment and:
    i. provide to another member of faculty and any external marker; and/or
    ii. submit it to a text matching software; and/or
    iii. submit it to a text matching software which may then retain a copy of the assignment on its database for the purpose of future plagiarism checking.
- I certify that I have not plagiarised the work of others or participated in unauthorised collaboration when preparing this assignment.

Signature .... *Date...*

* delete (iii) if not applicable

Signature Tuan Date: 2/5/2023 Signature ______________________ Date: ________

Signature Zafri Date: 2/5/2023 Signature ______________________ Date: ________

Signature Ong Date: 2/5/2023 Signature ______________________ Date: ________

Table of Contents

Part I **3**
Literature review 3
Introduction 3
Background 3
Existing methodologies 3
Ganapathy at el. (2008) 3
Shi et al. (2018) 4
Fomani et al. (2017) 4
Kaur et al. (2014) 5
Similarities 5
Differences 5
Proposed methodology 6
Task Allocation 6
Part II **7**
Overall Methodology 7
Detailed description of your developed algorithms and the Python functions used in each step. 7
Detailed description of the experimental results. 8
Discussion / critical evaluation on the outcomes and any observations 8
References **9**

Part I

Literature review

Introduction

Licence plate is the unique identification of every vehicle. A lot of research has been done on automatic licence plate recognition due to it being very useful in areas such as law enforcement, surveillance, toll and car park payment systems. Licence plate localization is the detection and extraction of licence plates. It is an essential pre-requisite for subsequent tasks in licence plate recognition, such as character segmentation and recognition. In recent years, there are numerous methods to tackle number plate recognition, for this report we will analyse a few of them.

Background

Licence plate localization is a challenging task due to the various factors that affect the quality and condition of the input image. These factors include poor lighting conditions which can cause shadows and reflections that obscure the licence plate as well as varying angle and orientation of the licence plate. Additionally, blurry images, and variation in plate design, such as black or white edges, can also pose difficulties to locate the licence plate.
The localization process generally consists of 2 stages: preprocessing and segmentation. Preprocessing involves the application of filters and other methods to reduce noise and highlight key features of the plate. Segmentation refers to the process of separating the license plate from its background and cropping it out from the image.

Existing methodologies

Ganapathy at el. (2008)

A study by Ganapathy at el. (2008), proposed a method using morphological processes and connected components analysis to locate the licence plate. Based on the paper, the original RGB image needs to be converted into grayscale and subsequently into a binary image in order to be able to work with morphological processes.
The author uses Otsu's thresholding to binarize the image. This is a global thresholding method, the optimal threshold is suggested by minimizing the sum of intra-class variances of the object and background pixels. The results from Otsu's method shown in the study, shows that the characters of the licence plate are clearly separated from its background. Compared to other thresholding methods such as adaptive thresholding and Maximum Entropy Thresholding, the advantage of Otsu's method is that the optimal threshold is automatically selected from a gray level histogram and it is not computationally expensive. (Bangare, 2015). The disadvantage of it is it can be sensitive to noise. The study has a 91% successful detection rate. However, the study

recommends the use of adaptive thresholding techniques to increase the detection rate as adaptive thresholding is effective for images with uneven illumination or variable background intensity.

The next step is to use opening to deal with noise in the binarized image, particularly the areas around the number plates. Erosion removes the small objects around the licence plate and smooths the edges and dilation is used to fill in any gaps or holes which further smooth the edges. The segmentation stage involves the extraction of the license plate region from the binary image. This is achieved by using a region growing algorithm that identifies connected components in the image. The connected component that is most likely to be a license plate is then extracted and subjected to further processing.

The study uses a modified hough transform. Normally, it is used to detect the borders of the number plate, but in the study, the focus is directed to the characters of the licence plate instead. However, the modified hough transform will usually detect more than one candidate region. To tackle this problem, vertical projection is used to evaluate the candidate regions. The most suitable candidate region is then selected and cropped out from the image and this will be the licence plate.

Shi et al. (2018)

In 2018, Shi et al. proposed a different approach from the one above. The detection algorithm proposed uses a convolutional neural network. The algorithm contains 2 main parts, candidate box generation and candidate box classification and regression. The candidate box is generated based on edge detection methodologies, using grayscale conversion, gaussian blur, Sobel edge detection, binarization and closing operation. These preprocessing steps are similar to the study above, the difference is the study above uses hough transform and vertical projection analysis to evaluate the candidate box, whereas this study uses CNN to evaluate the candidate box.
Two cascaded CNN namely C-Net and R-Net are applied, C-Net is used to locate the area where the license plate is present, while R-Net is employed to refine the position and size of the detected license plate based on the output of C-Net. The algorithm is fast and accurate when working with their own collected datasets. However the authors acknowledged that there are still limitations such as, miss detection in actual application where there are multiple licence plate and suboptimal candidate box regression.

Fomani et al. (2017)

Fomani et al. uses grayscale conversion, gaussian blur and histogram equalization before any other steps are taken.
The authors described the disadvantages of using the Canny edge detection algorithm for licence plate detection. First, it is too sensitive to noise and will result in false edge detection. This will also produce spurious edges in the image, especially in regions with texture or fine

details. The next limitation is edge fragmentation, canny algorithm tends to fragment long connected edges into multiple smaller edges. These limitations will cause detection of insignificant areas, which refers to areas in the image that are detected but do not correspond to the actual edges of the licence plate. Therefore the author uses morphological operators and colour segmentation techniques to extract candidate boxes. After the candidate regions have been identified using colour segmentation, the authors apply a series of morphological operations, such as erosion and dilation, to refine the regions and remove any spurious regions. After colour segmentation, their algorithm uses local adaptive thresholding for binarising the image. The local adaptive thresholding is calculated based on the maximum and minimum value of width and height of license plate. Finally, the Canny edge detection algorithm is applied to the refined candidate regions to detect the license plate edges.

Kaur et al. (2014)

This study uses grayscale conversion and bilateral filter instead of gaussian blur. The study also uses adaptive histogram equalization for contrast enhancement. For edge detection, the authors used the Sobel operator. This study uses Otsu's thresholding to binarize the image. The authors use opening and closing to improve the quality of edges. Lastly, connected component analysis is used to locate the licence plates among the candidate boxes. This is done by finding the top 10 contours sorted by area in descending order. Finally, the paper applies a set of geometric constraints to filter out false positives. The proposed approach was found to be efficient and accurate in detecting and extracting number plates.

Similarities

- All four papers use grayscale conversion and edge detection algorithms to locate the license plate.
- Fomani et al. (2017) and Kaur et al. (2014) uses histogram equalization for contrast enhancement.
- All four papers use some form of thresholding, either global or local, to binarize the image.
- All four papers use morphological operations such as opening and closing to refine the candidate regions.
- All four papers use some form of candidate box generation and classification or segmentation to locate the license plate.

Differences

Evaluating candidate regions

i) Connected components

- Ganapathy at el. (2008), Vertical projection (target licence plate characters)

- Kaur et al. (2014), Find contours (target licence plate edges)

ii) CNN

- Shi et al. (2018), CNN and Regression

Binarize image

- Global Otsu's thresholding
- Local adaptive thresholding

Proposed methodology

Our group decided to focus on 2 papers, Ganapathy at el. (2008) and Kaur et al. (2014).
The preprocessing is similar, but the approach for locating the licence plates are different. Both use connected components to locate the licence plate. Ganapathy at el. (2008), uses blob analysis and vertical projection to locate the licence plates based on the characters. Kaur et al. (2014), focuses on the edge of the licence plate and finds contours in the image to locate the licence plates.

Report 1 will be focusing on the study by Ganapathy at el. (2008).

Report 2 will be focusing on the study by Kaur et al. (2014).

Task Allocation

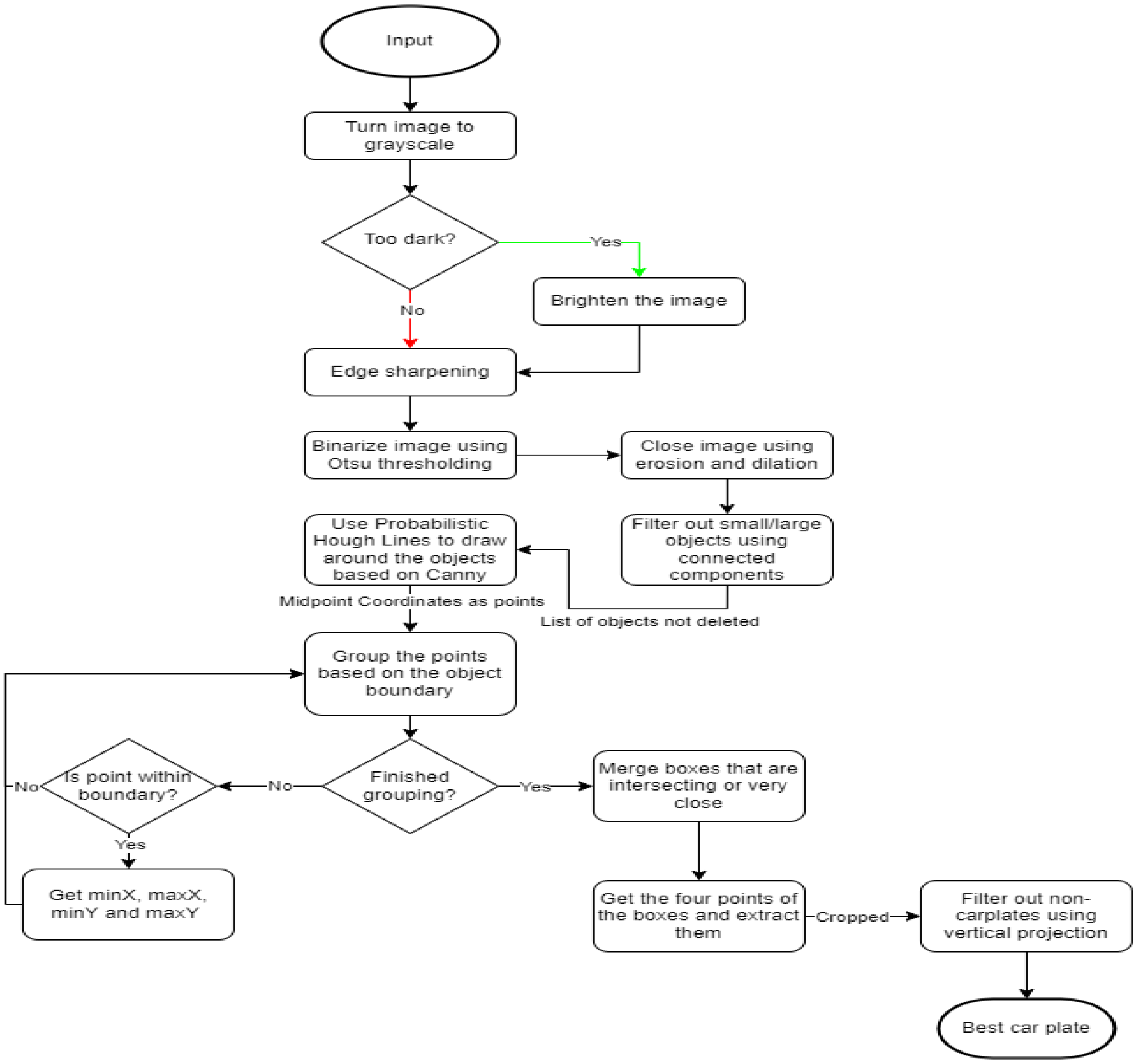
Part I
-All members of the group

Part II
Pre-processing
-All members of the group

Locating
-All members of the group

Part II

Overall Methodology



The flow chart above, describes the overall methodology of the car plate extraction process. With every image being different from each other, we needed to make sure each image is in its best condition for the selection process. Like, if the image is too dark, we brighten it, if edges are too dim, we sharpen them and when those are done, we use Otsu thresholding to remove those that are not dark enough or not white enough.

Based on the normalised image, we can remove those objects that are either too small or too big. After that, those images that survive the removal process, will be eligible for the merging process. Any boxes that are either very close together or intersecting may be merged together. Finally, the objects will be filtered using vertical projection and the best object will be selected.

Detailed description of your developed algorithms and the Python functions used in each step.

The developed algorithm follows a few sets of rules in achieving the best results. The rules that it follows depend on the brightness, size and clearness of an image. The aim is to normalise all of the images since most of the images greatly differ from one another. The process of normalisation depends on various techniques. One of which is binarizing the image using Otsu thresholding, it reduces the range of colour that we need to work with which allows for better accuracy. However, there are a few prerequisite techniques that we need to consider like whether the image is too dark or edges are too dim. In these scenarios, the binarization process will incorrectly categorise the pixels into the wrong range. Hence, 3 functions are used as the prerequisite, the functions are equalise(), canny(), gaussianBlur(). The equalise() function is for contrast stretching, dark images are brightened up. The canny() function is used for sharpening the edges, making sure that the edges will shine through. Its parameters for Hysteresis thresholding are 50 and 500, low and high respectively. The chosen high parameter is to ensure that we will not emphasise the edges that do not have a clear contrast. Following that, we can use gaussianBlur() to remove excess noise which we can finally apply Otsu thresholding on the image.

The next step was to use morphological operation to remove unwanted objects. The main technique used here was closing() function and blob_filter() function. Both techniques are quite popular as they further reduce noises. The closing function is derived from the fundamental operations of erosion and dilation where it first removes small objects which dilation will then reconstruct the lost shape. However, our closing() function includes another dilation operation which only dilates horizontally. The reason for dilating horizontally is that car plates tend to have larger width than its height so it makes sense that dilating horizontally can help merge the characters together, reducing the chances of a character missing from the final output. Besides that, our blob_filter() uses the concept of connected component labelling. The function assigns ID onto each object where the objects will have 5 attributes which is the total area of the object, starting x, y of the objects as well as the total height and width. From those attributes, we may filter out objects that are too long, too wide, small objects or large objects. But, it was not enough for perfectly sized objects that have a width and height of a car plate, so the solution was to use horizontal projection, which checks whether an object is irregularly shaped.

The last step was to locate the car plate. From the previous step, we were able to get an array of information about the objects that were not deleted and an image of the objects left. In order to get the outline of the objects, we needed line detection algorithms which needed the use of a canny algorithm. The line detection algorithm used the concept of probabilistic Hough lines to draw several smaller lines which midpoints are drawn around the objects. By using the information from blob analysis, the boundary of the objects is known, which we could use to group the points together while also getting the coordinates of the four corners. Consequently, some bounding boxes will have intersections, so we would fuse any bounding boxes that are either intersecting or close together using our own method fuseIntersectingBoundingBoxes(). With that, the final process is through the method filterGetBestImage() which returns an object that has the feature of a car plate.

Detailed description of the experimental results.

Of all the images from set 1 and 2, we were able to achieve a 93% pass rate where 56 out of 60 images showed a perfect conditioned car plate. In a scenario where multiple actual car plates are detected, only one of them would be showcased which has the best car plate features.

In the whole process, the erosion, dilation and horizontal dilation were the most important parameters for the detection of car plates. The reason is that the erosion was used to remove noises, but if the iteration is too high then the characters will lose their features and disappear entirely so an erosion constant that works for most images is used. Dilation and horizontal dilation was useful for merging objects but just like erosion, if dilation iteration were to be too high then it would merge with other unwanted objects. As long as the iterations do not negatively affect the outcome, then the iteration constants will not be changed.

In our testing process, we found that there were four types of car plate images, car plates with two rows, small and big gaps between characters, angled car plates and closed-up car plates. Previously, we would try different combinations of the erosion and dilation iterations but this would not work the best, as almost every single image would have different values. So, horizontal dilation was introduced to close the gaps between large gapped car plates but this was also not enough as two rowed car plates would not be affected with this new parameter while angled car plates may possibly merge with other objects. So, the horizontal dilation's kernel was kept at a minimum, a value that does not affect outcome negatively. Eventually, merging the close/intersecting bounding boxes was the best choice as it fixes the issue surrounding the dilation iteration number so with this, all car plate types would be able to be detected correctly with constant iteration values and kernel sizes.

Discussion / critical evaluation on the outcomes and any observations

The outcome could have been further improved where the constant values can be scaled with the size of the image. Furthermore, some images had uneven lighting which would throw off the accuracy, so a possible fix was to only brighten the part where it is darker. Furthermore, this algorithm has an average runtime of 5 seconds which would not be suitable for live detection.

References

1. Musoromy, Zuwena & Ramalingam, Soodamani & Bekooy, Nico. (2011). Edge detection comparison for license plate detection. 11th International Conference on Control, Automation, Robotics and Vision, ICARCV 2010. 1133 - 1138. 10.1109/ICARCV.2010.5707935. Retrieved from https://www.researchgate.net/publication/224216764_Edge_detection_comparison_for_license_plate_detection/citation/download
2. Sunil L. Bangare, Amruta Dubal. (2015) Reviewing Otsu's Method For Image Thresholding. Retrieved from http://www.ripublication.com/ijaerdoi/2015/ijaerv10n9_20.pdf
3. Ganapathy, V. & Lui, Dennis. (2008). A Malaysian Vehicle License Plate Localization and Recognition System. *Journal of Systemics, Cybernetics and Informatics*. 6. https://www.researchgate.net/publication/238757836_A_Malaysian_Vehicle_License_Plate_Localization_and_Recognition_System
4. Y. Shi, and Y. Chen, "License plate detection based on convolutional neural network and visual feature", International Conference on Mechanical, Control and Computer Engineering (ICMCCE), pp. 514-519, 2018.
5. Babak Abad Fomani, and Asadollah Shahbahrami, "License plate detection using adaptive morphological closing and local adaptive thresholding", 3rd International Conference on Pattern Recognition and Image Analysis (IPRIA), pp. 146-150, 2017. https://www.semanticscholar.org/paper/License-plate-detection-using-adaptive-closing-and-Fomani-Shahbahrami/17482f01446fba0638e6fd2c8e872ab6c5cbbe99
6. Kaur, Sarbjit & Kaur, Sukhvir. (2014). An Efficient Approach for Number Plate Extraction from Vehicles Image under Image Processing. https://www.researchgate.net/publication/262843164_An_Efficient_Approach_for_Number_Plate_Extraction_from_Vehicles_Image_under_Image_Processing/citation/download
7. Praveen. (2021, December 15). License Plate Recognition using OpenCV Python. *Medium*. Retrieved from: https://medium.com/programming-fever/license-plate-recognition-using-opencv-python-7611f85cdd6c
8. Ganapathy, V., & Lui, D. (2008). A Malaysian Vehicle License Plate Localization and Recognition System. *Journal of Systemics, Cybernetics and Informatics*. *6*(1), 13-20. Retrieved from https://www.researchgate.net/publication/238757836_A_Malaysian_Vehicle_License_Plate_Localization_and_Recognition_System
9. Bligny, T. J. Y. (2013). Car Reidentification System using Image Processing. *AALBORG UNIVERSITET*. Retrieved from https://www.researchgate.net/publication/258222322_Car_Reidentification_System_using_Image_Processing