Detection, Segmentation & Recognition

Module Code	**:**	CT103-3-M-IPCV - Image Processing and Computer Vision
Intake Code	**:**	APDMF2310AI
Lecturer Name	**:**	Assoc. Prof. Dr. V. Sivakumar
Hand in Date	**:**	1st of March, 2024
Student Name	**:**	Nurlan Amanov
Student Number	**:**	TP077526

Table of Contents

ABSTRACT

The main objective of the project is to create an application that will allow the detection of characters on license plates using image processing algorithms and a vision computer. The system uses algorithms and techniques such as adaptive thresholding, user-interactive region selection, and OCR, which ultimately helps to identify letters and numbers on car images. With complete experimentation and description of the steps of the whole system, there is a complete understanding of the working methods. A critical evaluation reveals the strengths but also the weaknesses, which are incomplete automation, sensitivity to noise and non-ideal operation of the OCR function, which has difficulty in identifying characters. With the identification of the strengths and weaknesses there are recommendations for further improvement of the project, including the elimination of weaknesses and the integration of new features, one of which is machine learning, which will improve the automation of the system. One of the other recommendations is to invest more time in the development of the project and conduct more experiments, which will give a more accurate assessment on the performance of the system and allow other weaknesses to be identified, which will only bring improvements to the system. At this stage, the project is to put the theoretical knowledge into practice and gain new knowledge in the image processing environment and computer vision.

1. INTRODUCTION

The development in the field of image processing and computer vision has brought a huge number of applications that are of great help in any field for human beings. The help is that work is simplified, and some tasks are done automatically without requiring more effort from people. With the advancement of technology, simplifying various tasks has become a major task for a lot of fields, which helps in speeding up the work process and consuming much less resources. One such example is the automatic detection and recognition of number plates, which has found its way into applications ranging from law enforcement to intelligent transport systems. Traditional detection methods utilize sophisticated image processing technologies, but developments in the same field have led to more efficient methods that provide more accurate solutions.

The main objective is to create an application, which will allow optimal number plate detection, minimizing the number of errors. In the future, the main objective is to improve the application, whereby the system will be able to detect number plates despite the elements surrounding them, also minimizing the number of errors.

2. PROBLEM STATEMENT

The ability to highlight just the right information is always a top priority in any job, especially in the task of identifying number plates, given the huge number of factors and things that surround these plates. The main challenge is to develop an effective system that will allow you to identify the characters on the license plate. The complexity of the whole problem lies in the surrounding elements that complicate the task of number plate detection, considering a huge number of factors: environment, lighting and so on.

Accurately identifying license plates from the surrounding background is a major challenge. Changes in lighting, different reflections and so on are the main problem and it is the task of the application to combat such problems by eliminating them, which will reduce the number of number plate detection errors. Character recognition is also one of the main tasks. Creating or using a function that will allow to accurately recognize letters and digits on the number plate without errors.

Such a system is of great use and application, especially to keep an eye on law and order. According to Forbez.kz, in the period January-November 2023, 8.2 million offences were recorded, an increase of 21.4% compared to last year and this is just the offences that were recorded. A robust system will increase efficiency, reducing the need for manual use and increasing the use of automated systems.

3. OBJECTIVES

The main objective of this study, in addition to the development of the application, is to obtain new knowledge in the field of image processing and computer vision, thus expanding the theoretical knowledge and their use in practice, which is the creation of an application for the detection and recognition of number plates, to learn about the different methods that are available currently. By creating an automated system for the recognition of number plates, attention is paid to the choice of different algorithms, which allows us to create a system of interest. The system should be efficient, the documentation should be understandable, which together will help to solve real problems. Creating an effective solution is an integral part of any automated system, to reduce the number of errors and speed up the process of performing the task.

Along with the creation of the system, it is important to document the whole process and clearly describe the work done and the results that the system showed and its analysis. A comprehensive report with a detailed description of the algorithms and functions that were used for implementation.

4. PROPOSED ALGORIGTHM WITH IMPLEMENTATION

4.1. Stages and Flow Chart

Next, we will present the system that allows you to determine the number plates, as well as the stages and algorithms used in the system itself, which was created in MATLAB. To achieve the desired result, as stated earlier, various stages of image processing have been used. It is important that any system is structured, which is easily understood by a human being for further explanation of its working. The system developed from the pre-processing stages to recognize the letters and numbers on the number plate. The system is divided into several stages:

1. Read the image.
2. Convert the image to grayscale.
3. Adaptive thresholding: create binary image using Otsu's method, which chooses the threshold value to minimize the intraclass variance of the thresholded black and white pixels.
4. Display the original image and segmented regions.
5. Interactively select a region of interest (ROI).
6. Crop the binary mask based on the selected ROI.
7. Convert the binary image to double for next step (insertObjectAnnotation).
8. Set the character set to include both letters and numbers.
9. Display the recognized text.
10. Display the annotated image.

Also with the demonstrated steps, Figure 1 shows a Flow Chart, which will also give an idea of the developed system and its working methods. With the steps described and the Flow Chart in place, it will be much easier to understand how the system works and what algorithms it uses, allowing everyone to understand how it works. Further, the operation of each step or algorithm will be described, which will give even more information about how the system works. The entire system has been created in MATLAB. Each step will present a portion of its code to

compare and understand how the system works.

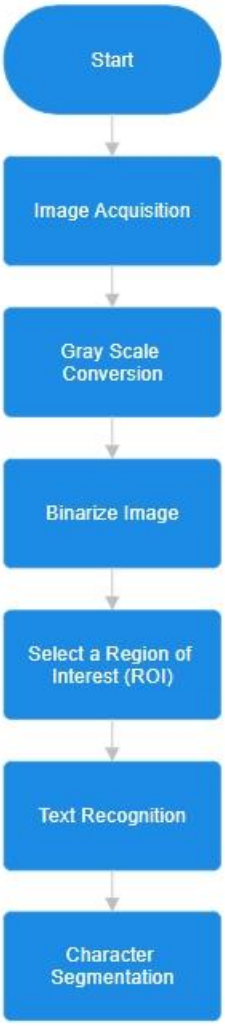

Figure 1. Flow Chart

4.2. Image Acquisition

One of the first and most important steps is to get the image itself. The system can handle .jpg, .bmp, .png and .tif files. The image is obtained by selecting from a pop-up window in front of the user by using uigetfile. Below is the code:

```
% Read the image
[file, path] = uigetfile({'*.jpg;*.bmp;*.png;*.tif'}, 'Choose an image');
s = [path, file];
img = imread(s);
```

4.3. Gray Scale Conversion

After obtaining the image, the next stage Gray Scale Conversion starts. The received image may be RGB and for further steps you need to convert it into a grayscale image. Below is a piece of code that helps with this process:

```
% Convert the image to grayscale (if not already)
gray_img = rgb2gray(img);
```

4.4. Image Binarizing

The next step involves binarizing the image to make it easier to recognize the characters in the image. Adaptive thresholding: create binary image using Otsu's method, which chooses the threshold value to minimize the intraclass variance of the thresholded black and white pixels. Below is a piece of code that helps with this process:

```
% Adaptive thresholding
binary_mask = imbinarize(gray_img, adaptthresh(gray_img, 0.7));
```

4.5. Region of Interest

The next step involves highlighting the region we need, i.e. the region in the image where the license plate is located. Through this process, the system will be able to focus only on the license plate, highlighting a specific region. Below is a piece of code that helps with this process:

```
% Interactively select a region of interest (ROI)
h = drawrectangle;
position = h.Position;
roi = round([position(1), position(2), position(3), position(4)]);

% Crop the binary mask based on the selected ROI
x = imcrop(binary_mask, roi);
```

4.6. Text Recognition and Character Segmentation

The next step involves using the selected region and recognizing the text in it. Using the ocr function, which uses the Tesseract OCR engine to perform automatic text detection and recognition. Below is a piece of code that helps with this process:

```
% Convert the binary image to double for insertObjectAnnotation
```

```
x_double = double(x);

% Set the character set to include both letters and numbers
ocrOutput = ocr(x_double, 'TextLayout', 'Line', 'CharacterSet',
'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789');

% Display the recognized text
disp('Recognized License Plate:');
disp(ocrOutput.Text);

% Display the annotated image
ocrImage = insertObjectAnnotation(x_double, 'rectangle', ocrOutput.WordBoundingBoxes,
...
    ocrOutput.WordConfidences);

figure;
imshow(ocrImage);
title('Recognized License Plate');
```

5. EXPERIMENTAL RESULTS

After describing the processes or the code of our system, we need to put the system into practice. For practical results we will use photos of license plates that were taken from the internet, and there is no common region for each license plate. Each license plate is linked by the fact that it has English letters in its numbers, thus the symbols to be determined are English letters and Arabic numerals. Next, the results will be provided. The total number of photos is 10, i.e. 10 photos were used to test the system's performance.

Figure 2. Experiment result with plate 1

Figure 3. Experiment result with plate 2

Figure 4. Experiment result with plate 3

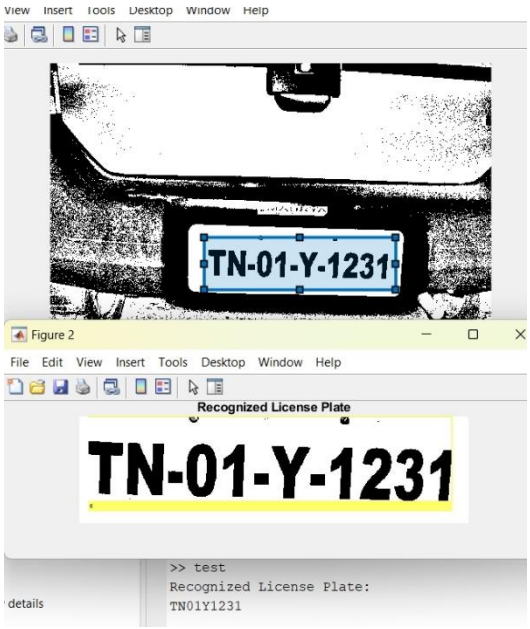

Figure 5. Experiment result with plate 4

Figure 6. Experiment result with plate 5

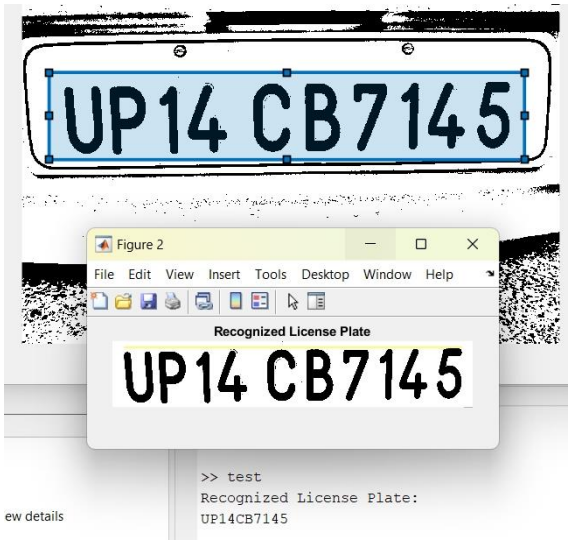

Figure 7. Experiment with plate 6

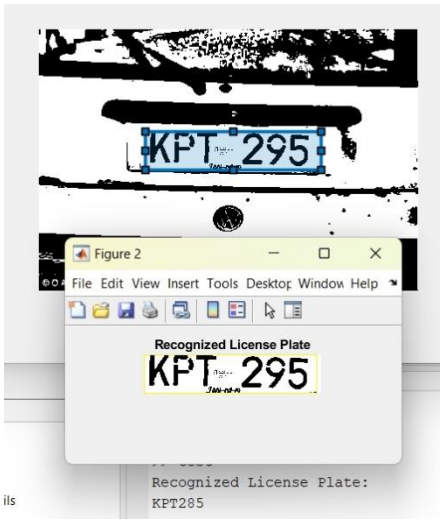

Figure 8. Experiment with plate 7

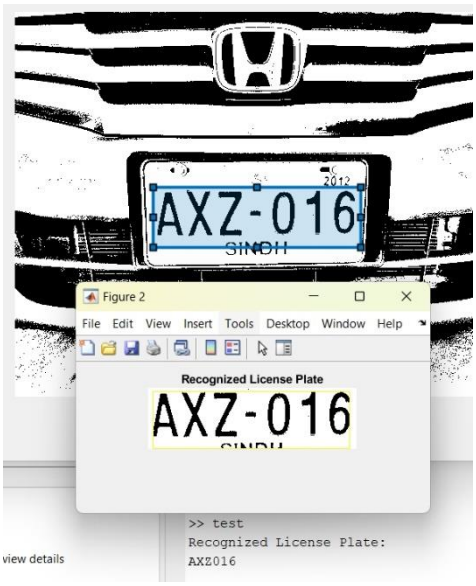

Figure 9. Experiment with plate 8

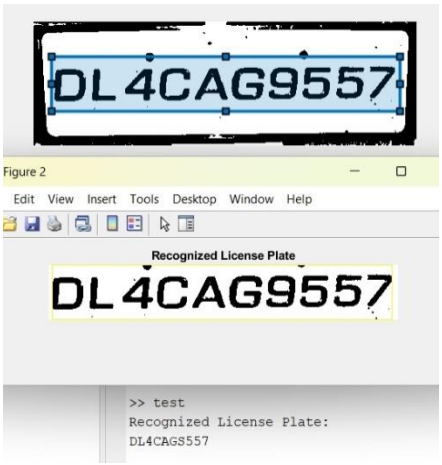

Figure 10. Experiment with plate 9

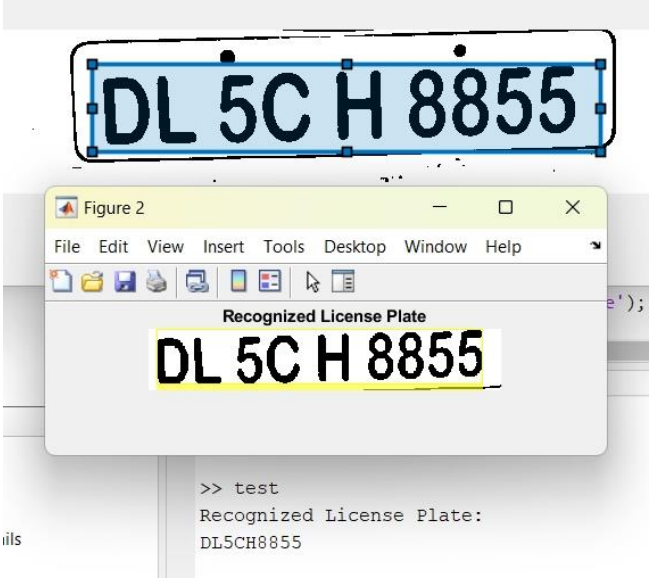

Figure 11. Experiment with plate 10

After conducting experiments using 10 license plates, positive results were obtained, but this does not yet indicate that the system is fully operational and ready for use, only that its development is moving in the right direction and in the early stages can identify pre-processed characters in an image.

6. CRITICAL EVALUATION AND ANALYSIS

The success of any automatic system is directly related to how accurately and efficiently it can produce results, in this case there is no exception. Critical analysis and evaluation of the system is important during the development phase. At the moment the system is able to identify symbols in the processed image, but the system does not allow the use of any photo with a license plate, because in some photos, after going through the process of processing, it is difficult to identify the symbols in the photo, even for the human eye, thus the lighting and various factors are still a problem in image processing. Thus, the system is sensitive to noise, at times when adaptive thresholding misperceives some parts of the image. Also, in one of the experiments, the system made an error in character recognition, thus recognizing the character "9" as the character "S", indicating the inferiority of the function itself, which is used for character recognition. Thus, with a strong distortion of characters, the function fails to cope with the recognition of characters. When creating an automated system, one of the goals is to minimize user intervention, but at this stage of development, the system requires manual selection of character regions on number plates, which is not ideal for an automated system.

The region selection function allows you to focus only on the region you need when using the recognition, which only increases the accuracy of the license plate symbols. At this stage of development, there is no alternative to automatic detection of this region. The use of Optical Character Recognition (OCR) significantly increases the efficiency of the system when recognizing characters, showing positive results that are almost flawless when considering the experiments described in this documentation. The use of adaptive thresholding shows positive results in stability under changing light and contrast, if we consider the experiments described in this documentation, because the characters are perfectly visible and easy to recognize by the system.

7. CONCLUSIONS AND RECOMMENDATIONS

The development of an automated system, using image processing algorithms and computer vision, allows us to gain valuable knowledge about the creation of similar applications that allow us to recognize different things, in this case number plates. The system created shows positive results, but requires further development, which will allow for more extensive use, with potential for use in professional areas, but for the time being, can be considered a university project to use the theoretical knowledge in practice. Negative aspects of the system are sensitivity to noise, problems with distorted characters and dependence on the user, which does not allow it to have the full title of automated. Thus, with the continued development of the system there are recommendations for further development of the system:

1. Fully automate the process, thereby eliminating the manual process of selecting a region that has number plate characters. Taking this into account, there is a chance to implement machine learning, which will improve the efficiency and accuracy of the system's operation.
2. One of the following recommendations is to improve the pre-processing of the image, thereby introducing noise removal algorithms to better identify the characters on license plates.
3. Further, conducting many more experiments. To use many more images for the test, more than 10. To more accurately evaluate the system and its results.
4. Using more hardware, thereby using not only images but also video, on which the system will be able to identify the number plates. Receive the image itself directly through the camera in real time, to analyze the number plate.

Now, the system shows a lot of promise, but it is not perfect yet and has room for improvement and with the implementation of new features, the system can be used not only for number plate recognition, but also in other areas where such technology will help in efficient and accurate results.

REFERENCES

Forbes Kazakhstan. (2024, January 18). The number of driving offences and fines for them is growing in Kazakhstan. *Forbes Kazakhstan*. Retrieved March 1, 2024, from https://forbes.kz/news/2024/01/18/newsid_316335

MathWorks. (n.d.). ocr function. MATLAB Documentation. Retrieved 1 of March 2024, from https://www.mathworks.com/help/vision/ref/ocr.html

MathWorks. (n.d.). ROI creation overview. MATLAB Documentation. Retrieved 1 of March 2024, from https://www.mathworks.com/help/images/roi-creation-overview.html

MathWorks. (n.d.). uigetfile function. MATLAB Documentation. Retrieved 1 of March 2024, from https://www.mathworks.com/help/matlab/ref/uigetfile.html

APPENDIX A

Code of the License/Number Plate Recognition created in MATLAB

```
close all;

% Read the image
[file, path] = uigetfile({'*.jpg;*.bmp;*.png;*.tif'}, 'Choose an image');
s = [path, file];
img = imread(s);

% Convert the image to grayscale (if not already)
gray_img = rgb2gray(img);

% Adaptive thresholding
binary_mask = imbinarize(gray_img, adaptthresh(gray_img, 0.7));

% Display the original image and segmented regions
figure;
imshow(binary_mask)

% Interactively select a region of interest (ROI)
h = drawrectangle;
position = h.Position;
roi = round([position(1), position(2), position(3), position(4)]);

% Crop the binary mask based on the selected ROI
x = imcrop(binary_mask, roi);

% Convert the binary image to double for insertObjectAnnotation
x_double = double(x);

% Set the character set to include both letters and numbers
ocrOutput = ocr(x_double, 'TextLayout', 'Line', 'CharacterSet',
'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789');

% Display the recognized text
```

```
disp('Recognized License Plate:');
disp(ocrOutput.Text);

% Display the annotated image
ocrImage = insertObjectAnnotation(x_double, 'rectangle', ocrOutput.WordBoundingBoxes,
...
    ocrOutput.WordConfidences);

figure;
imshow(ocrImage);
title('Recognized License Plate');
```