

T.R.
GEBZE TECHNICAL UNIVERSITY
FACULTY OF ENGINEERING
COMPUTER ENGINEERING DEPARTMENT

RECOGNITION OF SPEED SIGNS

MUHAMMED İKBAL YAZICI

SUPERVISOR
DOÇ. DR. HABIL KALKAN

2024
GEBZE

ABSTRACT

The widespread use of autonomous vehicles has increased the importance of image processing techniques to ensure a safe travel experience. Aim of this project is recognition of speed signs from pictures by using image processing methods. Detecting circle-shaped red objects and reading the numbers inside is a method for identifying speed signs. Despite the methodology is fast and simple, results are not good enough to use it in real.

CONTENTS

Abstract	iii
Contents	iv
List of Figures	v
1 Implementation	1
1.1 Finding Red Circles	1
1.2 Finding Digit Contours	3
1.3 Speed Limit Determination	3
Bibliography	6
Appendices	6

LIST OF FIGURES

1.1	Input image (source:)	1
1.2	Red mask applied	2
1.3	Red Circles	2
1.4	Circles and contours	3
1.5	Arrays of each contour	4
1.6	Final Result	5

1. IMPLEMENTATION

Speed sign recognition is consist of three main steps. These steps are finding coordinates of the red colored circles, finding contours of potential digits inside the red colored circles and matching this contours with corresponding numbers.

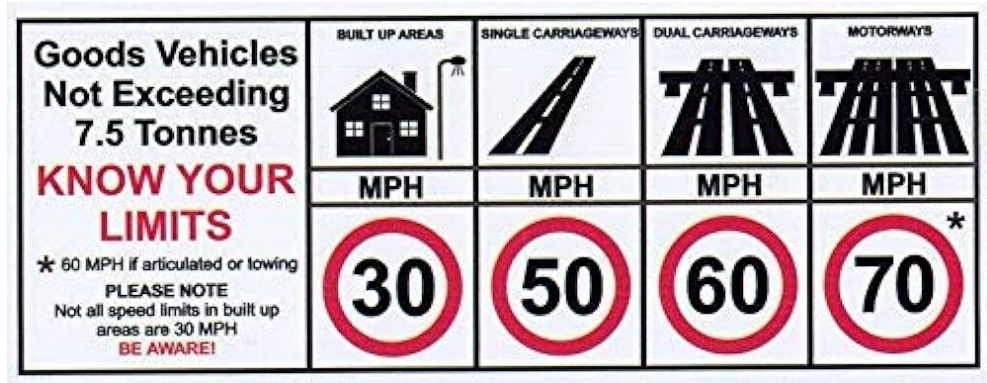


Figure 1.1: Input image (source:)

1.1. Finding Red Circles

To find red circles, it is necessary to first identify all red regions. To do this, the original image is converted to HSV format and a red color mask is applied. In the image obtained as a result of applying the mask, all areas of different colors will appear black and red areas will appear white. After masking circles are found by using Hough circle transform. Figure 1.2 shows the red areas of input image and Figure 1.3 shows the circle positions on the original image.

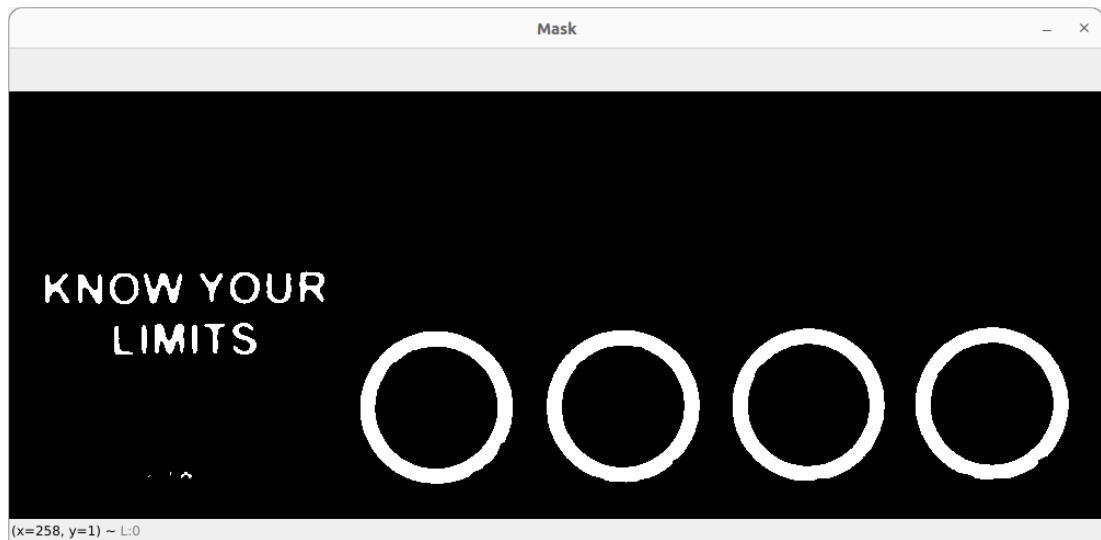


Figure 1.2: Red mask applied

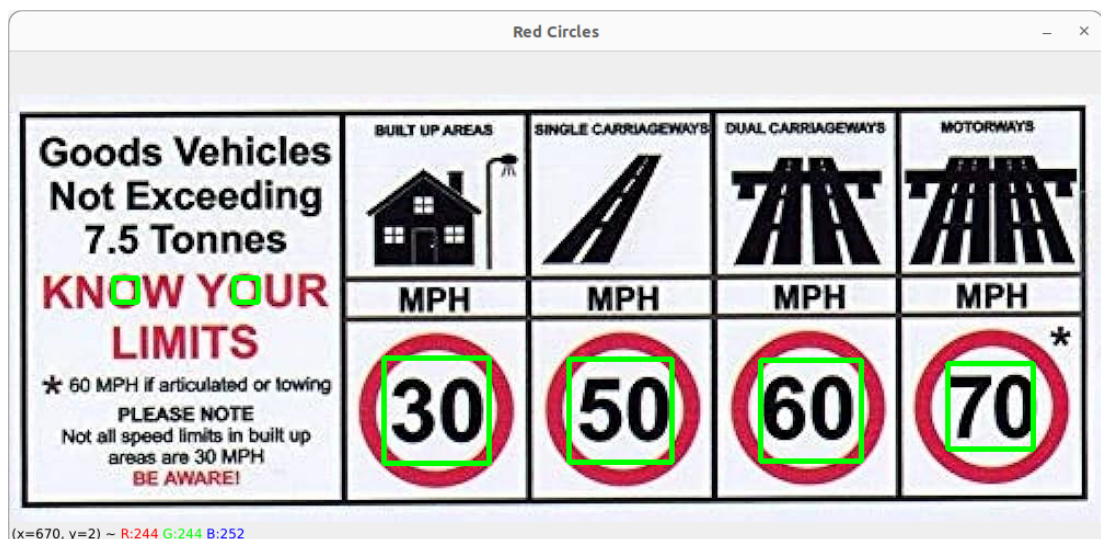


Figure 1.3: Red Circles

1.2. Finding Digit Contours

For every circle found in previous step, area in bounds of circle is copied. Every copy is converted to gray scale and these gray scale images converted to binary images by thresholding. Later, contours in these binary images are found. These contours are candidates for being numbers. Figure 1.4 shows the cropped circle bounds and contours in them.

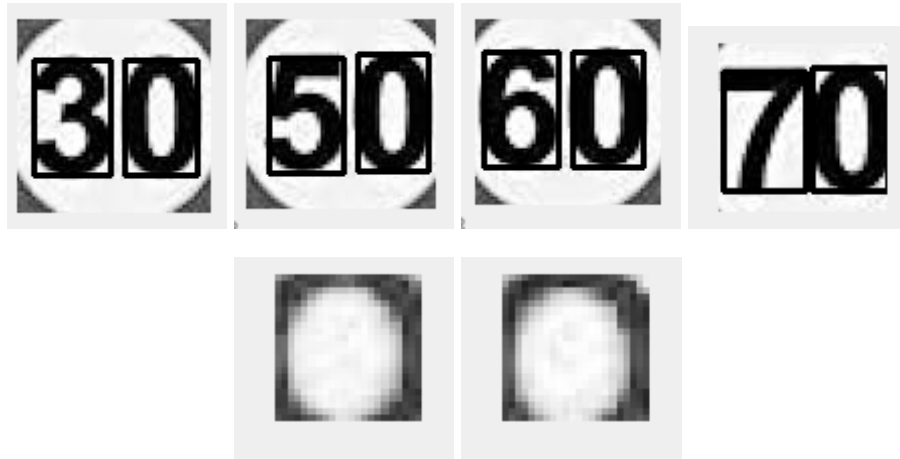


Figure 1.4: Circles and contours

1.3. Speed Limit Determination

Each contour is divided into 15 segments, 5 by 3. Mean intensity of each segment is computed and thresholded. If mean of the segment is black it represented with 1 and if mean of the segment is white it represented with 0. These values are stored in 2D array and each coordinate of the array is compared with pre-determined values for each number. If array match with one of the numbers, this contour is a digit of the speed limit. If there is no match this contour is ignored and if all contours in a circle are ignored this circle also ignored. Figure 1.5 shows the arrays of contours of each circle and Figure 1.6 shows the final result.

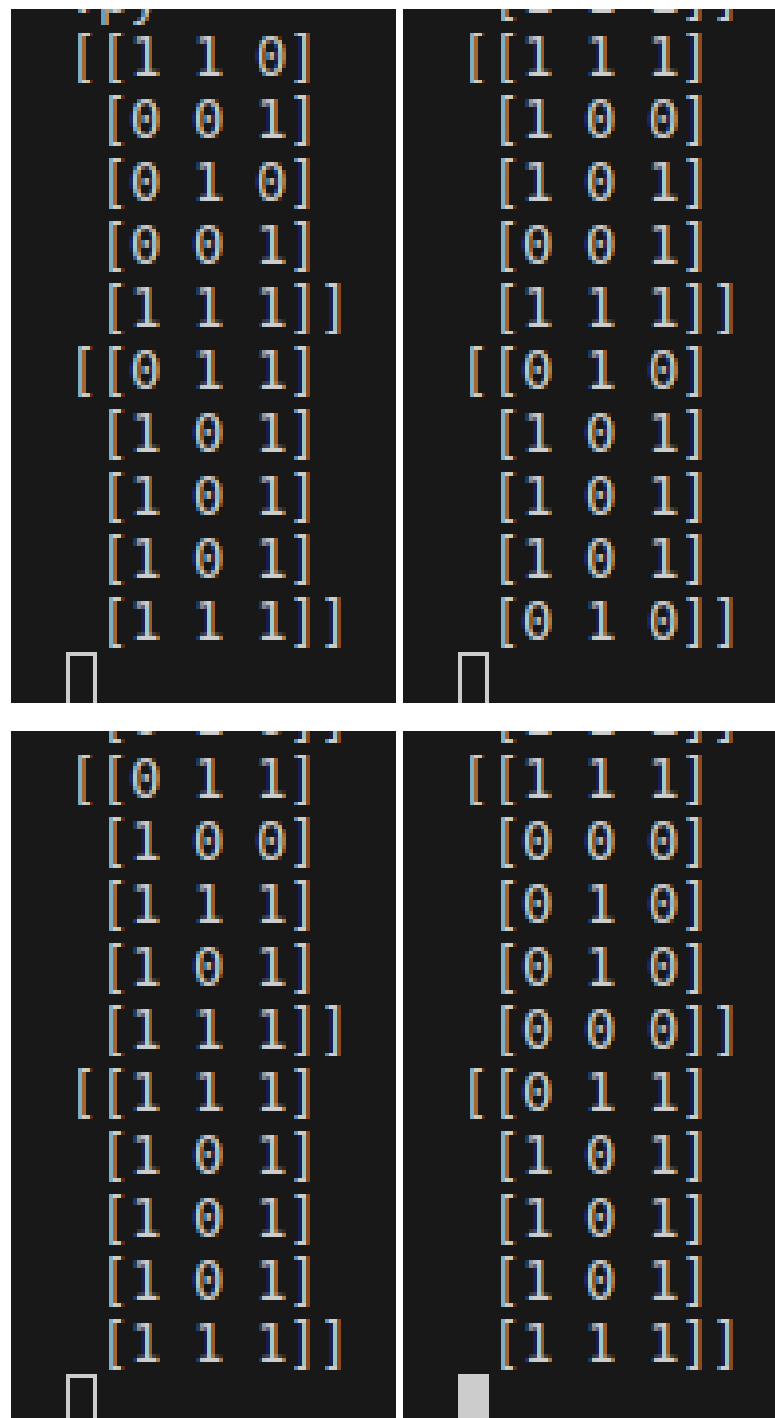


Figure 1.5: Arrays of each contour

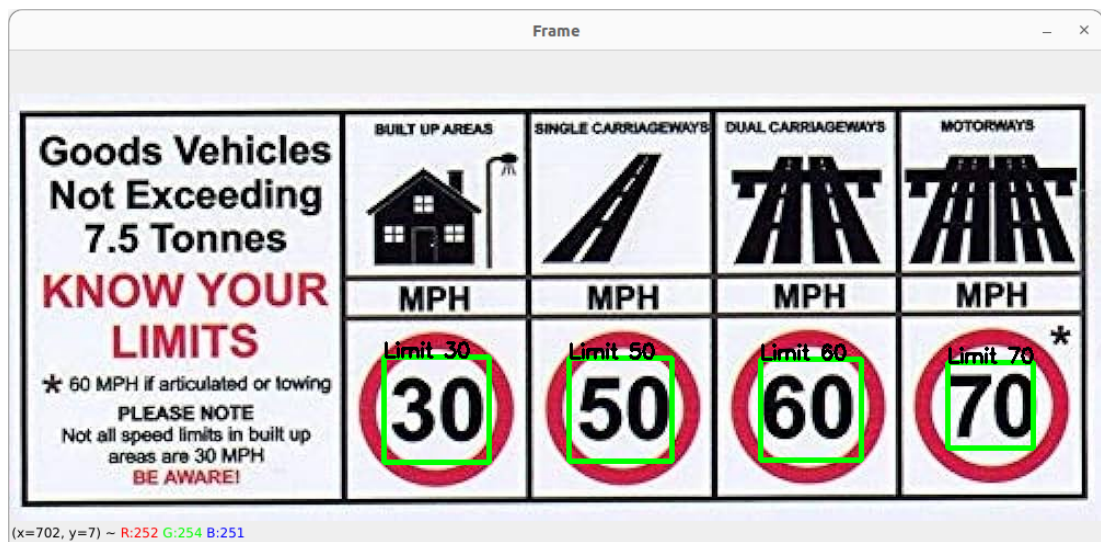


Figure 1.6: Final Result

APPENDICES

Appendix 1: Source code of the project

Github Link