JSS Mahavidyapeeta
JSS Science and Technology University
Sri Jayachamarajendra College of Engineering
Mysuru-570006

Computer Vision (IS654)

Topic: Recognition of Traffic Signs
using Computer Vision.

<u>Submitted by:</u>
Srikanta Vaibhav N
01JST17IS063
Dept. of Information Science
& Engineering, JSSSTU.

Declaration:

I solemnly declare that the project report RECOGNITION OF TRAFFIC SIGNS USING COMPUTER VISION is based on my own work carried out during the course of our study under the supervision of Dr. B S HARISH, Professor, for the award of B.E (IS&E), JSS Science and Technology University, Mysuru. I assert the statements made and conclusions drawn are an outcome of my researchwork.

I further certify that the work contained in the report is original and has been done by me under the general supervision of my supervisor. We have followed the guidelines provided by the university in writing the report. Whenever we have used materials (data, theoretical analysis, and text) from other sources, we have given due credit to them in the text of the report and giving their details in the references.

Srikanta Vaibhav N
01JST17IS063
6th semester, Dept. of IS&E, JSSSTU

Table of Contents:

Content Description Page No.

Abstract

Application of new technology in building human comforts and automation is growing fast, particularly in automobile industry. Automatic detection and recognition of traffic signs for assisting driver to ensure a safe travel have been given practical importance for intelligent traffic system. The proposed method detects the location of the traffic sign in the captured image, based on its geometrical characteristics and using color information by implementing morphological operations. Such signs are then recognized using pattern matching with the normalized keys of pre-computed cluster-centroids in the dictionary.

The algorithm is tested using image set of different traffic sign directions taken under various adverse conditions such as, various backgrounds, orientation and distances. Experimental result shows better performance in detection and recognition of road signs with recognition rate of 93.75%. Computational time is also quite low which makes it applicable for the real time system.

General terms: Object Recognition, Image Processing.
Key terms: Morphological Segmentation, Road sign detection and recognition.

Chapter 1: Introduction

1.1 Problem Statement

Detect and recognize various traffic signs on road, from images, for the benefit of safety of passengers in self-driving automated vehicles using Computer Vision.

1.2 General Introduction

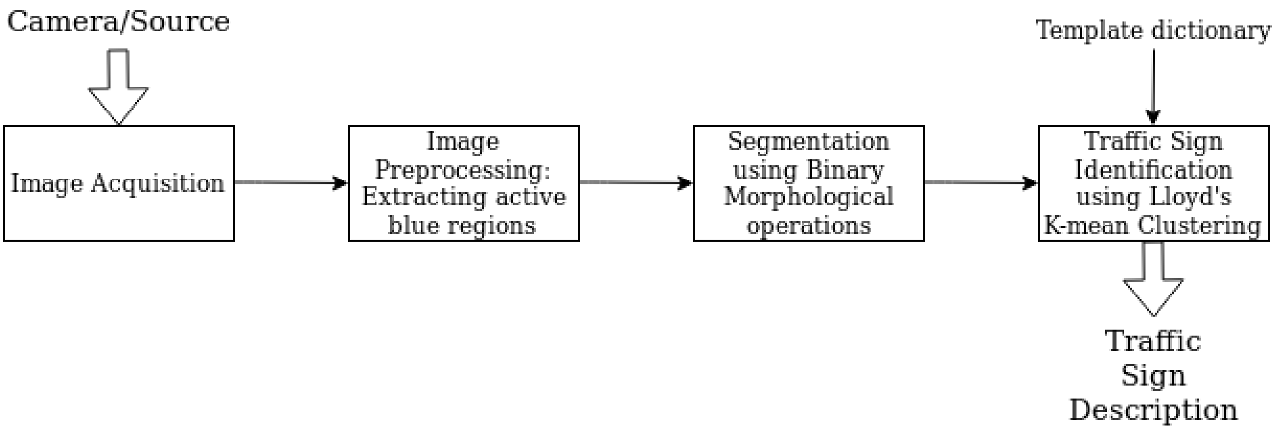
Traffic signs are an integral part of our road infrastructure. They provide critical information for road user, which in turn requires them to adjust their driving behaviour to make sure they adhere with the road regulations currently enforced. There are several different types of traffic signs like speed limits, no entry, traffic signals, turn left or right, children crossing, no passing of heavy vehicles, etc. Traffic signs can provide a wide range of variations between classes in terms of colour, shape, and the presence of pictograms or text.

Traffic sign recognition or classification is the process of identifying which class a traffic sign belongs to. But to achieve such autonomous level, it is necessary for vehicles to understand and follow all traffic rules.

In the world of Artificial Intelligence and advancement in technologies, many researchers and big companies like Tesla, Uber, Google, Mercedes-Benz, Toyota, Ford, Audi, etc. are working on autonomous vehicles and self-driving cars. Self-driving vehicles are which the passenger can fully depend on the vehicle for traveling. So, for achieving accuracy in this technology, the vehicles should be able to interpret traffic signs and make decisions accordingly.

Traditionally, standard Computer Vision methods are employed to detect and classify traffic signs.

1.3 General Block Diagram



1.4 Applications

Traffic sign detection and recognition plays an important role in expert systems, such as

- advanced driver-assistance systems
- automatic self-driving cars

1.5 Challenges

- Lighting conditions – there are differences in capturing images by daylight and night, or under the influence of a light source. Thus, shade of colours of objects can be seen differently by illumination changes. Shape based approach for traffic sign detection seems to be a good choice for solving this problem.

- Blurring and vibration by a moving vehicle, therefore the camera must be fixed properly.

- Occlusion – any kind of objects that block face of traffic signs, for example by trees, vehicles, pedestrians, poles or any objects on the road.

- Damage – traffic signs can be damaged not only by sun-shine, but also by vandalism or weather over time (strong breeze, storm, raining). They can be then dirty, scribbled over, tilted, rusty etc.

- Similarity –some of objects in traffic scene are similar to traffic signs, especially on the advertisements placed around the road.

1.6 Motivation

My choice of this topic is motivated by the following factors:

- Traffic sign recognition has high industrial potential in intelligent autonomous vehicle and driver assistance system. Today, driving safety is becoming a popular topic in many fields, from small projects to large car factories.

- In finding a good solution for driving and safety problems under the already cited challenges, such as, lighting conditions, occlusion, and damage of traffic signs.

- In developing an intelligent vehicle system which reduces the number of accidents due to traffic rules violation. In our modern age, around 1.3 million people die on roads each year. This number would be much higher without our road signs.

1.7 Objectives of the project

- To learn and implement various concepts of Computer Vision for detecting traffic signs.

- To build and train a traffic sign recognition model used in autonomous vehicles for accurate prediction of the correct road sign.

Chapter 2: Literature Survey

1. “The German traffic sign recognition benchmark: a multi-class classification competition”, by J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, in Proc. IEEE IJCNN, 2011, pp. 1453–1460.

This paper proposes the design and analysis of the “German Traffic Sign Recognition Benchmark” dataset and competition. It is a multi-category classification competition held at IJCNN (International Joint Conference on Neural Networks) 2011.

A comprehensive, lifelike dataset of more than 50,000 traffic sign images had been collected. The dataset comprises 43 classeswith unbalanced class frequencies. Participants have to classify two test sets of more than 12,500 images each. Here, the results on the first of these sets, which was used in the first evaluation stage of the two-fold challenge, are reported.

The best-performing methods in the competition were:

- Implementation of convolutional neural network (CNN) by teams IDSIA, from Switzerland and sermanet, from United States.

- Fast Intersection Kernel Support Vector Machine (IK-SVM) over concatenated HOG features by team VISICS, from Belgium.

- Discriminant Analysis on HOG features and Vector Quantization by team noob, Australia.

The results of the competition show that state-of-the-art machine learning algorithms perform very good in the challenging task of traffic sign recognition. The participants achieved a very high performance of up to 98.98% correct recognition rate which is similar to human performance on this dataset.

2. “Detection of traffic signs in real-world images: The German traffic sign detection benchmark”, by S. Houben, J. Stallkamp, J. Salmen, M. Schlipsing, and C. Igel, in Proc. IEEE IJCNN, 2013, pp. 1–8.

This paper proposes a real-world benchmark data set for traffic sign detection together with carefully chosen evaluation metrics, baseline results, and a web-interface for comparing approaches.

“German Traffic Sign Detection Benchmark” presented a competition at IJCNN (International Joint Conference on Neural Networks) 2013. The competition attracted 18 teams to submit over 110 results with 6 teams decided to publish their approaches in papers that were accepted for publication in the IJCNN proceedings.

The best-performing methods in the competition were:

- Pixel-wise color classification for ROI extraction and recognition by team LITS1.

- Variants of the integral channel feature detector by team VISICS, from Belgium.

- A coarse-to-fine algorithm for traffic sign detection by team wgy@HIT501.

In their evaluation, they separate sign detection from classification, also measured the performance on relevant categories of signs to allow for benchmarking specialized solutions.

The considered baseline algorithms represent some of the most well liked detection approaches such as the Viola-Jones detector based on Haar features and a linear classifier relying on HOG descriptors.

Further, a recently proposed problem-specific algorithm utilizing shape and color in a model-based Hough like voting scheme was evaluated.

3. "Towards Real-Time Traffic Sign Detection and Classification" by Yi Yang, Hengliang Luo, Huarong Xu and Fuchao Wu,2014, IEEE.

This paper aims to deal with real-time traffic sign recognition, i.e. localizing what type of traffic sign appears in which area of an input image at a fast processing time.

To achieve this objective, a two-module framework (detection module and classification module) is proposed:

In detection module:

- The input colour image is transformed to probability maps by using colour probability model.

- A color probability model is proposed to deal with color information of traffic signs, so as to enhance the specific colors (e.g., red, blue and yellow) of traffic signs and suppress background colors as well as to reduce the search space of following algorithms and detection time.

- Then the road sign proposals are extracted by finding maximally stable extremal regions on these maps.

- Lastly, an SVM classifier which prepare with colour HOG features is used to further filter out the false positives and classify the present proposals to their super classes.

In classification module:

- They used CNN to classify the detected traffic signs to their sub-classes within each super class.

- In order to improve computational efficiency, three CNNs with simple structure were trained.

Demonstration on the GTSDB benchmark (German Traffic Sign Benchmark) shows that their method achieved comparable performance to the state-of-the-art methods with outstanding improved computational efficiency, which is 20 times faster than the existing best method.

It is worth to note that this method could be accelerated with GPU, which could further improve the computational efficiency.

4. “Real-Time Detection and Recognition of Road Traffic Signs” by Jack Greenhalgh and Majid Mirmehdi, Senior Member, 2012, IEEE.

This paper aims to deal with novel system for the automatic detection and recognition of traffic signs.

The training data were generated from synthetic template images that are freely available from an online database; thus, real footage of road signs were not required as training data. Therefore, a total of 1200 synthetic images were generated for each of 43 classes.

Candidate regions are detected as maximally stable extremal regions (MSERs), which offers ruggedness to variations in lighting conditions.

Recognition of traffic symbols is based on a cascade of support vector machine (SVM) classifiers that were trained using histogram of oriented gradient (HOG) features. The classifier that was trained on the synthetic data gave anaccuracy of 85.7%, and the classifier that was trained on real data gave an accuracy of 85.9%.

Based on these results, it was shown that the synthetic data set produced results comparable to a data set of hand-labeled real images.

This system is accurate at high vehicle speeds, operates under a range of weather conditions, runs at an average speed of 20 frames per second and recognizes all classes of idea-based (non-textual) traffic symbols from an online road sign database.

5. “Traffic indication symbols recognition with shape context” by Kai Li, Weiyao Lan, Department of Automation Xiamen University, China, 2011, IEEE.

In this paper to detect the traffic sign, HIS color model followed by circle detection is used. The regions detected by color detection cannot be determined to the exact sign region.

- In this method the edge of interested regions is traced to get their contours after morphologic operations.
- Then to find the target region, Hough circle transform is applied.

The object have been detected and extracted after the previous two steps. We next recognize the symbol in the destination area.

- The image is preprocessed to remove noise.
- To obtain a clear silhouette boundary of the traffic indication symbol Edge detection and segmentation are used specifically to the image.
- Shape context is based on the contour of the object.

6. “Traffic sign detection and recognition for intelligent vehicle” by Long Chen, Qingquan Li and Qingzhou Mao, Published in IEEE Intelligent Vehicles, 2011.

In this paper, a computer vision based system is proposed for real-time robust traffic sign detection and recognition, especially developed for intelligent vehicle.

In detection phase, a color-based segmentation method is used to scan the scene in order to quickly establish regions of interest (ROI). Sign candidates within ROIs are detected by a set of Haar wavelet features obtained from AdaBoost training.

Then, the Speeded Up Robust Features (SURF) is applied for the sign recognition. SURF finds local invariant features in a candidate sign and matches these features to the features of template images that exist in data set.

The recognition is performed by finding out the template image that gives the maximum number of matches. Evaluation of the proposed system is done on an intelligent vehicle SmartVII. A recognition accuracy of over 90% in real-time had been achieved.

Chapter 3: Proposed Method

3.1 Design and Required Mathematical Equations

The proposed method for traffic sign recognition consists of two basic tasks: traffic sign detection and classification.

➢ **Traffic Sign Detection:**
Traffic signs have regular shapes and clearly visible colours that are different from natural objects and background. The goal of traffic sign detection is to find the locations and sizes of traffic signs in the given images. The well-defined colours and shapes are two main cues for traffic sign detection.

Firstly, we extract binary image with active blue regions (because we deal with traffic signs having blue coloured background) using HSV colour model.

Then, we employ ***Binary Morphological Segmentation*** aiming to find traffic sign board in the given image. Morphological segmentation in binary images aims to find regions that corresponds to individual overlapping particles. Each particle is marked first, ultimate erosion maybe used for this purpose or markers maybe placed manually.

The next task is to grow objects from the markers provided they are kept within the limits of the original set and parts of objects are not joined when they come close to each other. The technique used for this purpose is called dilation. Dilation is used for growing, and the result is constrained by the two conditions: remain in the original set, and do not join particles.

Binary erosion and dilation are explained as follows:
The sets of black and white pixels constitute a description of a binary image. Assume that only black pixels are considered, and the others are treated as a background. The primary morphological operations are dilation and erosion, and from these two, more complex morphological operations such as opening, closing, and shape decomposition can be constituted.

Erosion:
The morphological transformation Erosion $\ominus$ combines two sets using vector subtraction (or Minkowski set subtraction, e.g., (a, b) - (e, d) = (a - c, b - d)) of set elements. In simple terms, it erodes away the boundaries of the foreground object.
Let *E* be a Euclidean space or an integer grid, and *A* a binary image in *E*. The erosion of the binary image *A* by the structuring element *B* is defined by:

$$A \ominus B = \{z \in E | B_z \subseteq A\}$$

where B_Z is the translation of *B* by the vector z, i.e., $B_z = \{b + z | b \in B\}$

Dilation:
The morphological transformation Dilation $\oplus$ combines two sets using vector addition (or Minkowski set addition, e.g., (a, b) + (e, d) = (a + c, b + d)) of set elements. It is the opposite of erosion. Let *E* be a Euclidean space or an integer grid, *A* a binary image in *E*, and *B* a structuring element regarded as a subset of R^d.

The dilation of *A* by *B* is defined by $A \oplus B = \bigcup_{b \in B} A_b,$

where A_b is the translation of A by b.

➢ **Traffic Sign Classification:**

In detection of traffic signs, we have detected signs in a given raw image. However, we still do not know which classes they belong to. Therefore, we further classify the detected signs into their classes in this section.

Lloyd's K-Means Clustering technique along with ***Template Matching*** is employed to classify/identify the detected traffic sign from the image.

K-means clustering is a method of vector quantization, that aims to partition N observations into K clusters in which each observation belongs to the cluster with the nearest mean (cluster centers or cluster centroid), serving as a prototype of the cluster.

Mathematical background of K-means:
The k-means algorithm takes a dataset X of N points as input, together with a parameter K specifying how many clusters to create. The output is a set of K cluster centroids and a labeling of X that assigns each of the points in X to a unique cluster. All points within a cluster are closer in distance to their centroid than they are to any other centroid.
The mathematical condition for the K clusters C_k and the K centroids μ_k can be expressed as:

$$\sum_{k=1}^{K} \sum_{\mathrm{x}_n \in C_k} ||\mathrm{x}_n - \mu_k||^2 \quad \text{with respect to } C_k, \mu_k.$$

Lloyd's algorithm:
Finding the solution is unfortunately NP hard. Nevertheless, an iterative method known as Lloyd's algorithm exists that converges in few steps. The procedure alternates between two operations. (1) Once a set of centroids μ_k is available, the clusters are updated to contain the points closest in distance to each centroid. (2) Given a set of clusters, the centroids are recalculated as the means of all points belonging to a cluster.

$$C_k = \{\mathrm{x}_n : ||\mathrm{x}_n - \mu_k|| \leq \text{ all } ||\mathrm{x}_n - \mu_l||\} \qquad (1)$$

$$\mu_k = \frac{1}{C_k} \sum_{\mathrm{x}_n \in C_k} \mathrm{x}_n \qquad (2)$$

The two-step procedure continues until the assignments of clusters and centroids no longer change. The detected traffic sign is then identified by comparing (template/pattern matching) the obtained prototype of the ROI with the keys of pre-determined clusters in the dictionary.

3.2 Algorithm:

Steps Description

1. Image acquisition - read an image from the dataset.

2. Perform image preprocessing -
   - Find blobs with blue colour on the image.
   - After blobs were found detect the largest square blob, that must be the sign, as follows.
   - Define range HSV for blue colour of the traffic sign.
   - Compute frame area i.e. image size.
   - Convert colour image to HSV colour scheme.
   - Extract binary image with active blue regions.

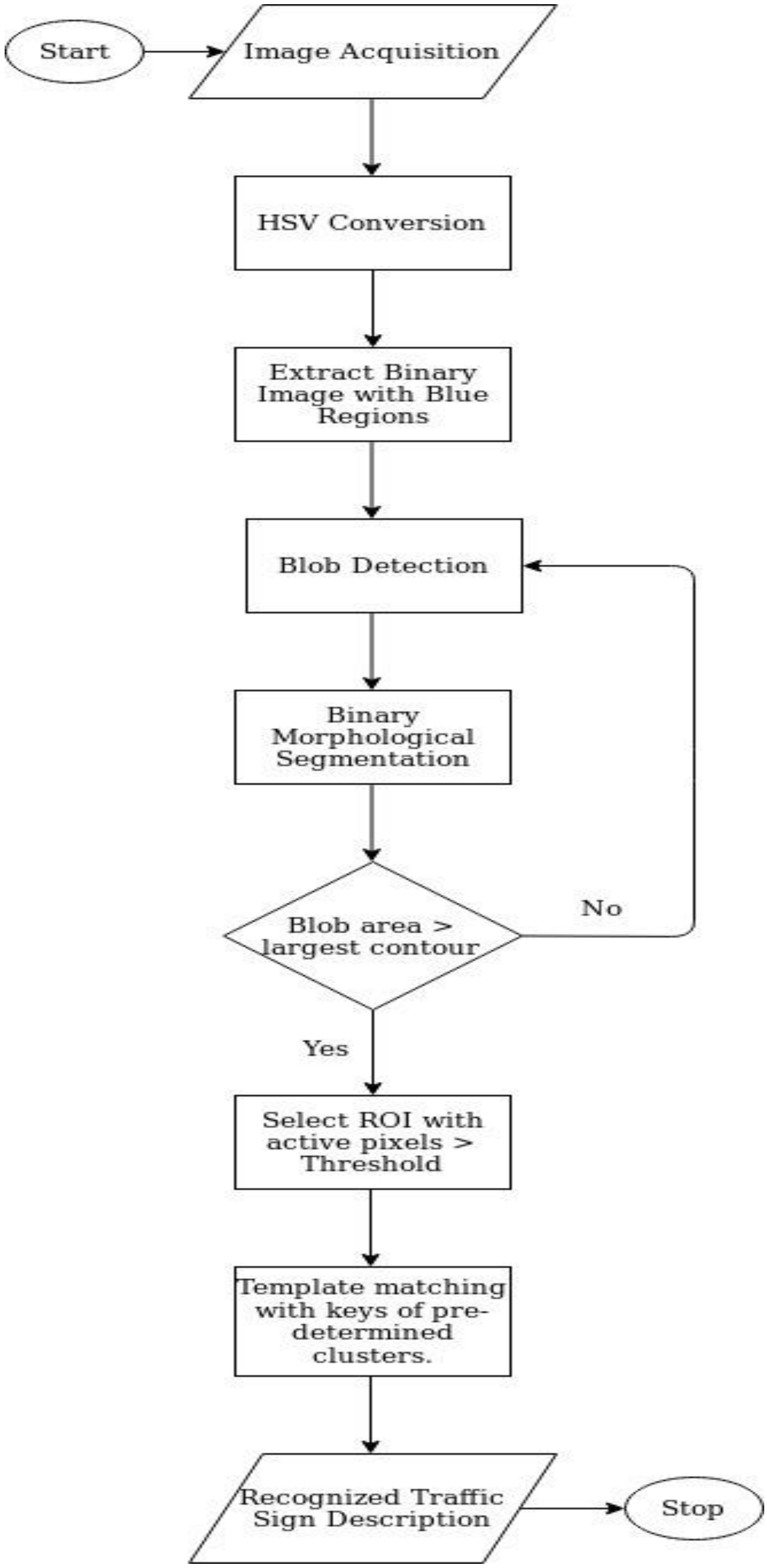
3. Perform Binary Morphological Segmentation - to find regions corresponding to individual overlapping objects, based on image shape. It needs two inputs- the original image, and structuring element or kernel which decides the nature of operation. Two basic morphological operators used are Erosion (it erodes away the boundaries of foreground object) and Dilation (opposite of erosion).
   - Define kernel for smoothing.
   - Perform morphological operations:
     - Opening (erosion followed by dilation)
     - Closing (dilation followed by erosion)
   - Find contours in the mask and proceed if at least one contour was found:
     - Draw a bounding rectangle with minimum area, so it considers the rotation also.
     - Compute area of rectangle from Euclidian distance of each side of the rectangle to find the largest rectangle within all contours.
   - Draw contour of the found rectangle on the original image.
   - If largest rectangle is not none: cut and warp interesting area.

4. Perform template matching with K-mean clusters - to identify the detected traffic sign.
   - Select some ROI in the warped image in which we expect to have the sign parts.
   - If the ROI has more active pixels than threshold we mark it as 1, else 0 (defining clusters).
   - After path through all four regions and tracking the fraction of each ROI, we compare the tuple of ones and zeros with keys in dictionary (centroids of pre-determined clusters) to identify the sign.

5. Output the recognized traffic sign description.

3.3 Flow Chart

Visual representation of the above algorithm **:**



Chapter 4: Experimental Analysis

4.1 <u>Information on the Dataset used</u>

The project requires images of traffic signs captured under various circumstances and background. I have dealt with four sub-classes of traffic signs under the super-class Direction-signs. The four sub-classes are: Move Forward, Turn Around, Move Left and Move Right. Hence, given an image, we identify to which sub-class it belongs to.

The image formats used are: .png and .jpeg. A total of 12 images: 4, 2, 4, and 2 for Move Forward, Turn Around, Move Left and Move Right have been considered respectively.

4.2 <u>Experimental Settings</u>

I have used basic computer vision-based python library modules namely, imutils - version 0.5.3, OpenCV – version 4.2.0.34, and numpy – version 1.19.dev0 under python 3.8.2. This project does not necessarily require an IDE to run. An Ubuntu 18.04.2 LTS terminal with the above modules installed is fairly sufficient.

These modules enable us to exploit certain series of convenience functions to make basic image processing functions such as translation, rotation, resizing, skeletonization, displaying Matplotlib images, sorting contours, detecting edges, and much more easier with OpenCV and both Python 2.7 and Python 3.

In brief, Experimental environment:
OS: Ubuntu 18.04.2 LTS
CPU: Intel Core i5-4200U CPU @ 1.60GHz × 4
OS type: 64-bit
Memory: 4 GB
Language: Python 3.8.2

4.3 <u>Result Table</u>

Sl. No.	Type of Traffic Sign	No. of Input images	Sign Detection	Sign Recognition	Success rate	Recognition Time (T) *
1.	Move Forward	4	4	4	4/4	0.56s
2.	Turn Around	4	4	4	4/4	0.66s
3.	Move Left	4	4	3	3/4	0.48s
4.	Move Right	4	4	4	4/4	0.59s

* Ignoring interactive response-wait time.

4.4 <u>Discussion on Results</u>

Several experiments have been conducted to verify the efficiency and the accuracy of the proposed method for the detection and recognition of road traffic signs.

From the above result table:
Success rate in % = (4/4 + 4/4 + 3/4 + 4/4) x 100 / 4 = 93.75%
Average recognition time (T) = (0.56 + 0.66 + 0.48 + 0.59) / 4 = 0.5725s

Therefore, the accuracy of the recognized traffic signs is above 90%. This means that almost all the images in our data-set are recognized correctly. The average recognition time (T) is fairly acceptable w.r.t images in our data-set.

4.5 <u>Complexity of the Algorithm</u>

Let the total number of pixels in an image be N, and let there be M types of traffic signs in the data-set.

- For converting a colour image to HSV colour scheme and to a binary image -
  In each case, the algorithm takes N steps to iterate through N individual pixels.
  Hence, the time complexity in this case is **O(N)**.

- For performing morphological segmentation -
  Two basic morphological operators are used: Dilation and Erosion.
  The 2D Dilation and 2D Erosion algorithms (for 2D data i.e. an image) iterates over all co-ordinates of the input image. Hence, the time complexity of the 2D algorithm is **O(N)** i.e. linear w.r.t the size of the image (N pixels).
  At every coordinate, the 2D Dilation calls twice the 1D Dilation function: once for the vertical dilation and once for the horizontal dilation part. Similarly, at every coordinate, the 2D Erosion calls twice the 1D Erosion function: once for the vertical erosion and once for the horizontal erosion.

- For template matching with pre-computed keys of clusters in the dictionary-
  Since there are M types/sub-classes of traffic signs in the data-set, there will be M number of keys in the dictionary. To make a comparison with keys and find the correct associated label, M iterartions have to be made. Therefore, at the worst case time complexity will be **O(M)** which is also the average case. In best case i.e. when the match is found in the first comparison, complexity will be O(1). (Note: The k-means algorithm is known to have a quadratic time complexity which is ignored, because we only deal with pre-computed values of cluster centroids of standard shaped traffic signs).

Therefore, the algorithm takes O(N), O(N), and O(M) steps at each stage of pre-processing, detection and recognition respectively. The resulting equation would be,

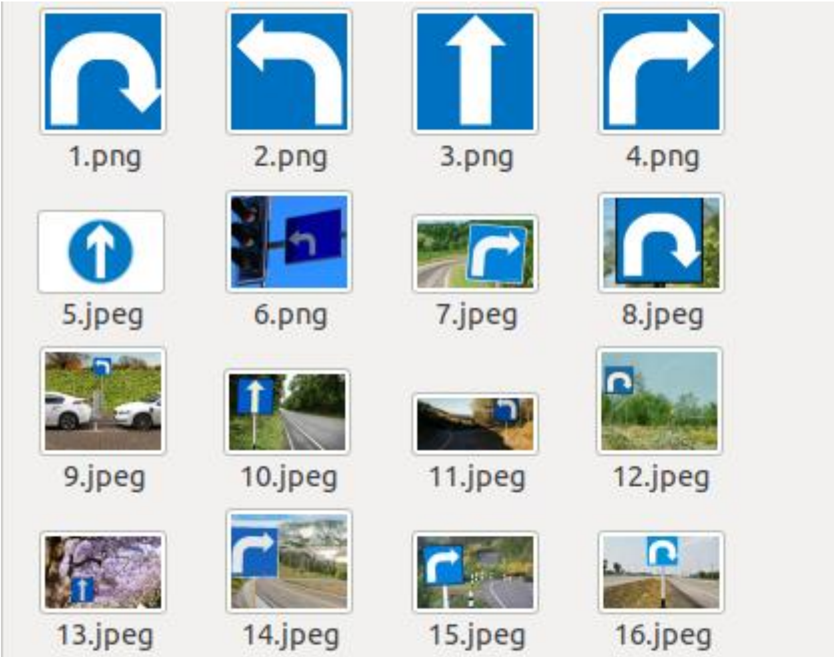
Time Complexity = O(N) + O(N) + O(M)

Since, N>>M i.e. the number of pixels in an image is much greater than the different types of traffic signs that exist, we ignore O(M) as it has negligible impact on the running time of the algorithm.

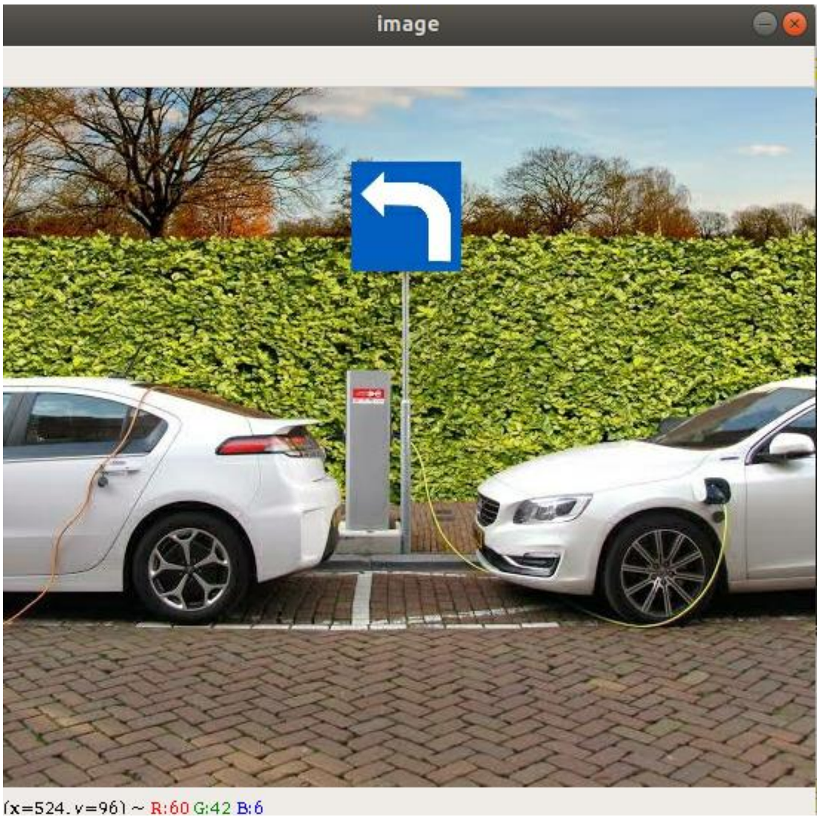
Therefore, Time Complexity = 2 x O(N) ≈ **O(N)**.

4.6 Snapshots of the Results:

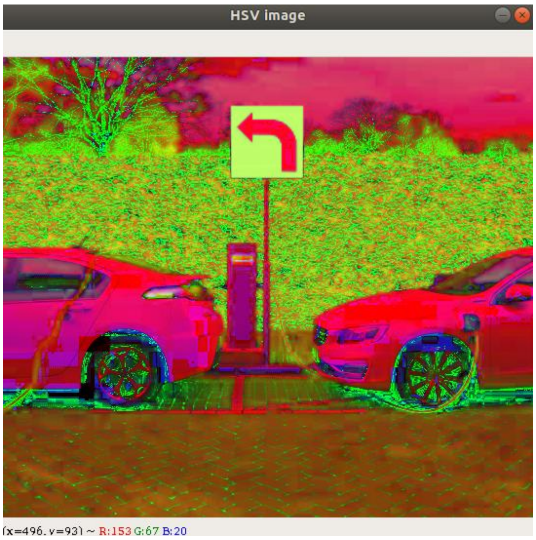
Dataset:

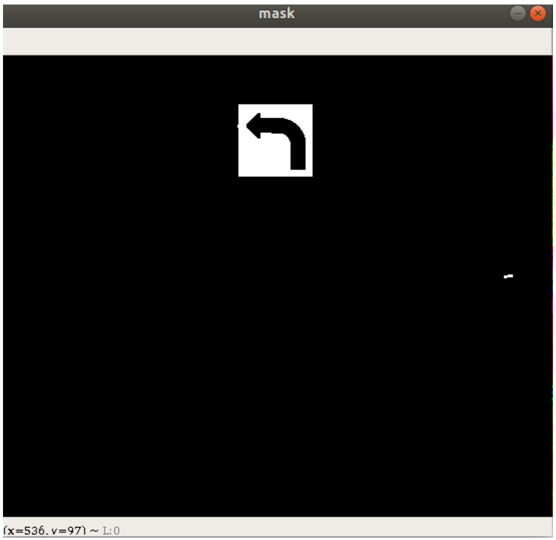


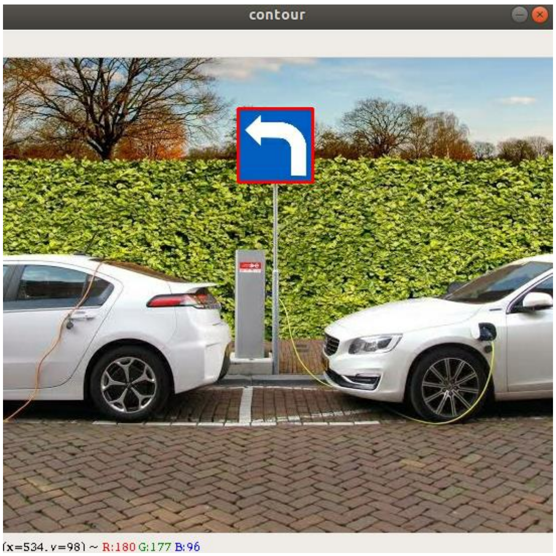
1. Input:

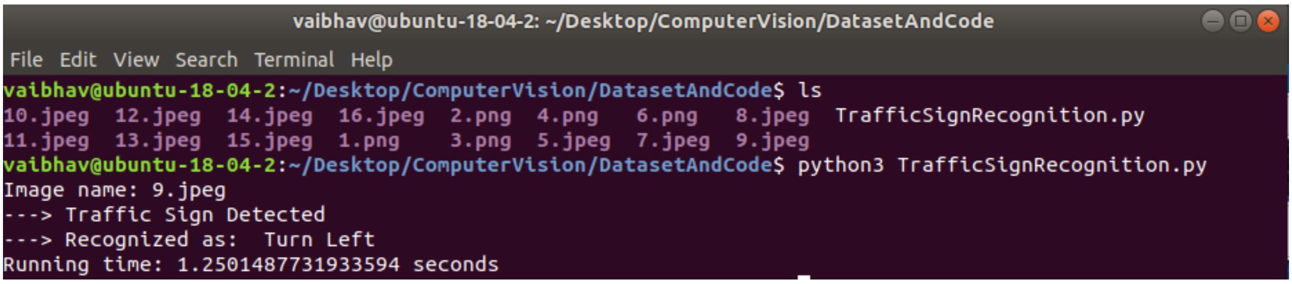


Output:





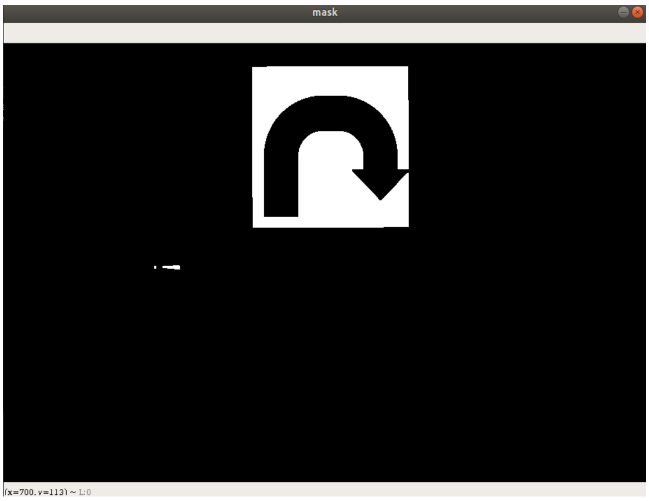


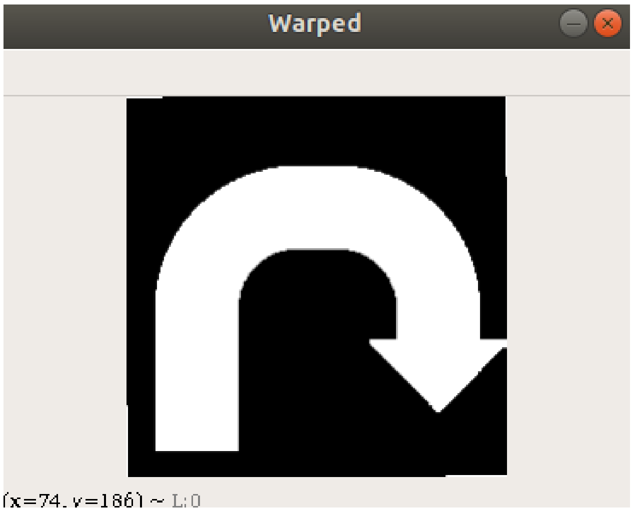


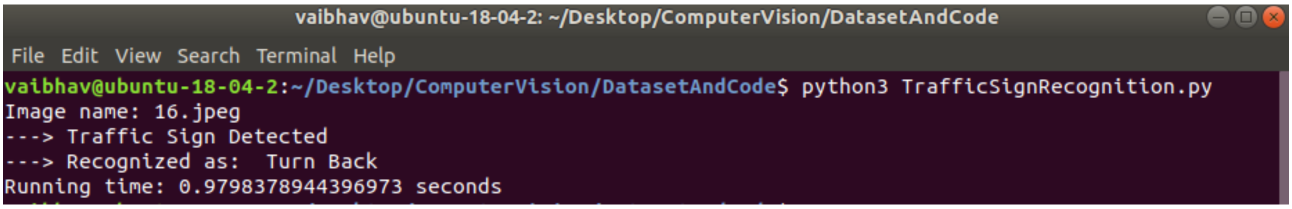

```
vaibhav@ubuntu-18-04-2:~/Desktop/ComputerVision/DatasetAndCode$ ls
10.jpeg  12.jpeg  14.jpeg  16.jpeg  2.png  4.png   6.png   8.jpeg  TrafficSignRecognition.py
11.jpeg  13.jpeg  15.jpeg  1.png    3.png  5.jpeg  7.jpeg  9.jpeg
vaibhav@ubuntu-18-04-2:~/Desktop/ComputerVision/DatasetAndCode$ python3 TrafficSignRecognition.py
Image name: 9.jpeg
---> Traffic Sign Detected
---> Recognized as:  Turn Left
Running time: 1.2501487731933594 seconds
```

2. Input:

Output:




Warped
(x=74, y=186) ~ L:0


vaibhav@ubuntu-18-04-2: ~/Desktop/ComputerVision/DatasetAndCode
File Edit View Search Terminal Help
vaibhav@ubuntu-18-04-2:~/Desktop/ComputerVision/DatasetAndCode$ python3 TrafficSignRecognition.py
Image name: 16.jpeg
---> Traffic Sign Detected
---> Recognized as: Turn Back
Running time: 0.9798378944396973 seconds

Chapter 5

Conclusion:

In this project, we aimed to detect and recognize traffic signs based on morphological segmentation and pattern matching with centroids of k-mean clusters is proposed. The obtained traffic sign images for the data-set are from different orientations. The HSV color space is used to segment the blue background of the traffic sign from the image as it decouples the color and intensity information. The different shapes of the sign are detected using binary morphological segmentation and finally recognized using pattern/template matching.

The experimental analysis showed better performance under various conditions such as, different backgrounds, lighting conditions, orientations and distances. We could achieve an efficiency of 93.75% with an average recognition time of 0.5725s. The algorithm can be further improved to perform better in all kinds of atmospheric and luminance conditions and can be made dynamic to work in real time.

Future scope:

This topic, Recognition of Traffic Signs has got immense future scope with an open challenge at research and industrial levels. At research level, the algorithms that drive this project can be constantly improved and efforts to emerge with new ways of solving challenges can be focussed.

At industrial level, manufacturers of on-road automatic vehicles need to incorporate this feature. For example, advanced driver-assistance systems and automatic self-driving cars. A completely independent automatic vehicle demands high-level of accuracy in decision making. Henceforth, making it completely reliable and safe.

6. References

- Miss. Priyanka A.Nikam, Prof.Nitin B.Dhaigude, “A Survey on Road sign Detection and Classification”, Volume: 04 Issue: 04, IRJET, 2017.

- S. Houben, J. Stallkamp, J. Salmen, M. Schlipsing, and C. Igel, “Detection of traffic signs in real-world images: The German traffic sign detection benchmark”, in Proc. IEEE IJCNN, pp. 1–8, 2013.

- Petr Dokládal, Eva Dokladalova, “Computationally efficient, one-pass algorithm for morphologicalfilters.” Journal of Visual Communication and Image Representation, Elsevier, 22 (5), pp.411–420. ?10.1016/j.jvcir.2011.03.005. hal-00692897, 2011.